



User Manual

Saola Animate

version 2.0

Last updated 05/15/2018

Table of Contents

Welcome to Saola Animate

Chapter 1: Overview

User Interface	8
Welcom Screen	8
Workspace	9
System Requirements	11
Common Editing Commands	11
Cut	11
Copy	11
Paste	11
Delete	12
Undo	12
Redo	12

Chapter 2: Creating Projects

Creating, Opening and Saving Projects	13
Creating Projects	13
Opening Projects	14
Saving Projects	15
Working with Scenes	15
Managing Scenes	15
Editing Scene Properties	16
Project Backup and Recovery	17

Chapter 3: Responsive Design

Responsive Scaling	18
Flexible Layout	18
Relative Positioning and Sizing	19
Maintaining Aspect Ratio	19
Pinning Element Position	19
Responsive Layout	20
Overview	20
Managing Breakpoints and Layouts	21
Managing Property and Animation Overrides	26
Resizing the Viewport	28
Responsive Symbols	28

Chapter 4: Inserting and Editing Elements

Element Overview	29
DIVs	29
Text	29
HTML Widgets	30

Images	31
Audio and Videos	32
Symbols	35
Shapes	35
Groups	39
Inserting Elements	40
Inserting Elements	40
Copying Elements	41
Removing Elements	41
Selecting Elements	41
Selecting Elements on the Canvas	41
Selecting Elements in the Timeline Pane	42
Selecting Elements in the Elements Pane	42
Arranging Elements	42
Aligning Elements	42
Ordering Elements	44
Grouping Elements	46
Editing Element Properties	47
Editing Properties on the Canvas	47
Editing Properties in the Properties Pane	47
Adding Text to Elements	48
Chapter 5: Creating Animations	
Keyframes	50
Adding Keyframes	50
Editing Keyframe Values	53
Moving, Copying and Removing Keyframes	54
Auto-Keyframe Mode	54
Animations	55
Animation Segments	56
Transitions	57
Easing	57
Motion Paths	72
Creating Motion Paths	73
Editing Motion Paths	74
Orienting Elements to Paths	76
Sprite Sheet Animations	76
Default Sprite	77
Custom Sprite	79
Chapter 6: Adding Interactivity	
Events, Actions and Triggers	81
Adding Events - Actions	81
Adding Actions	82

Editing Actions	83
Replacing Actions	83
Ordering Actions	83
Copying Actions	83
Removing Actions	83
Inserting Timeline Triggers	83
Inserting Triggers	84
Editing Triggers	85
Moving Triggers	85
Copying Triggers	85
Removing Triggers	85
Inserting Timeline Labels	85
Inserting Labels	85
Moving Labels	86
Copying Labels	86
Removing Labels	86
Chapter 7: Using Symbols	
Converting Elements to Symbols	87
Editing Symbols	88
Exporting and Importing Symbols	89
Exporting Symbols	89
Importing Symbols	89
Chapter 8: Using Fonts	
Choosing Fonts	90
Adding Google Fonts	90
Removing Google Fonts	93
Chapter 9: Previewing and Exporting Projects	
Updating Document Properties	94
Previewing Scenes and Projects in Browser	96
Exporting Projects to HTML5	96
Browser Compatibility for HTML5	97
Chapter 10: Using Canvas	
Rulers	98
Snapping on Canvas	98
Zooming Canvas In and Out	99
Chapter 11: Using Timeline Pane	
Managing Multiple Timelines	100
Creating Timelines	100
Removing Timelines	101
Renaming Timelines	101
Setting Timelines as Main Timeline	101
Navigating in Timeline	102

Snapping in Timeline	102
Zooming Timeline In and Out	103
Managing Timeline View	103
Showing and Hiding Element Properties	104
Filtering Elements	104
Showing, Hiding and Locking Elements	104
Showing and Hiding Elements	104
Locking and Unlocking Elements	105
Chapter 12: Using Properties Pane	
General Section	106
Position & Size Section	108
Transform Section	110
Fill Section	111
Border Section	115
Corner Section	116
Text Section	116
Image Section	118
Media Section	119
Embedded Section	120
Scene Section	120
Transition Effects Section	122
Chapter 13: Using Resources Pane	
Managing Resources	123
Importing Resources	123
Removing Resources	124
Exporting Resources	124
Editing Symbol Resources	124
Script Resources	124
Editing Script Resources	124
Including JavaScript and CSS in Document <head>	125
Ordering JavaScript and CSS	125
Resource Groups for Audio and Videos	126
Adding Formats to Groups	127
Removing Formats from Groups	127
Exporting Formats	127
Previewing Formats	127
Chapter 14: JavaScript	
Creating Functions	128
Removing Functions	128
Renaming Functions	129
Exporting Functions to JavaScript Files	129
Editing Functions	129

JavaScript APIs	130
Document	131
Timeline	133
Element	134
Event	137
Preloader	137
Chapter 15: Using Text Editor	
Chapter 16: Using Color Picker	
Chapter 17: Event - Action References	
Events	140
Actions	143
Chapter 18: Customizing Preloader	
Default Preloader	147
Custom Preloader	148
Chapter 19: Menus, Toolbars and Shortcuts	
File Menu	150
Edit Menu	151
Scene Menu	153
Element Menu	154
Timeline Menu	155
View Menu	157
Help Menu	158
Floating Text Format Toolbar	159
Shortcuts For Script Editor	161
Chapter 20: Customizing Saola Animate	
Changing Language	163
Customizing User Interface	163
Showing and Hiding Panes and Toolbars	163
Changing Pane Location	163
Changing Pane Size	164
Customizing Preferences	164
General Tab	165
Shortcuts Tab	166
Code Editor Tab	167

Welcome to Saola Animate

Have Fun Creating Interactive Web Content

Saola Animate is the complete solution for creating HTML5 animation and interactive content for web, digital publishing, advertising, and more. The application's focus is twofold:

- Make animation design as efficient and enjoyable as possible.
- Enable users to produce the best possible animated content.

About This Documentation

This documentation is designed both as a course in using Saola Animate and as an ongoing reference while you are working with the application. You can skim it quickly for reference and refer to it for additional information whenever you need.

Get Started - New Users

- Study the **Overview** to familiarize yourself with the basics of the application.
- Visit the **User Community** or **email us** to get help and more information about the application.

Chapter 1: Overview

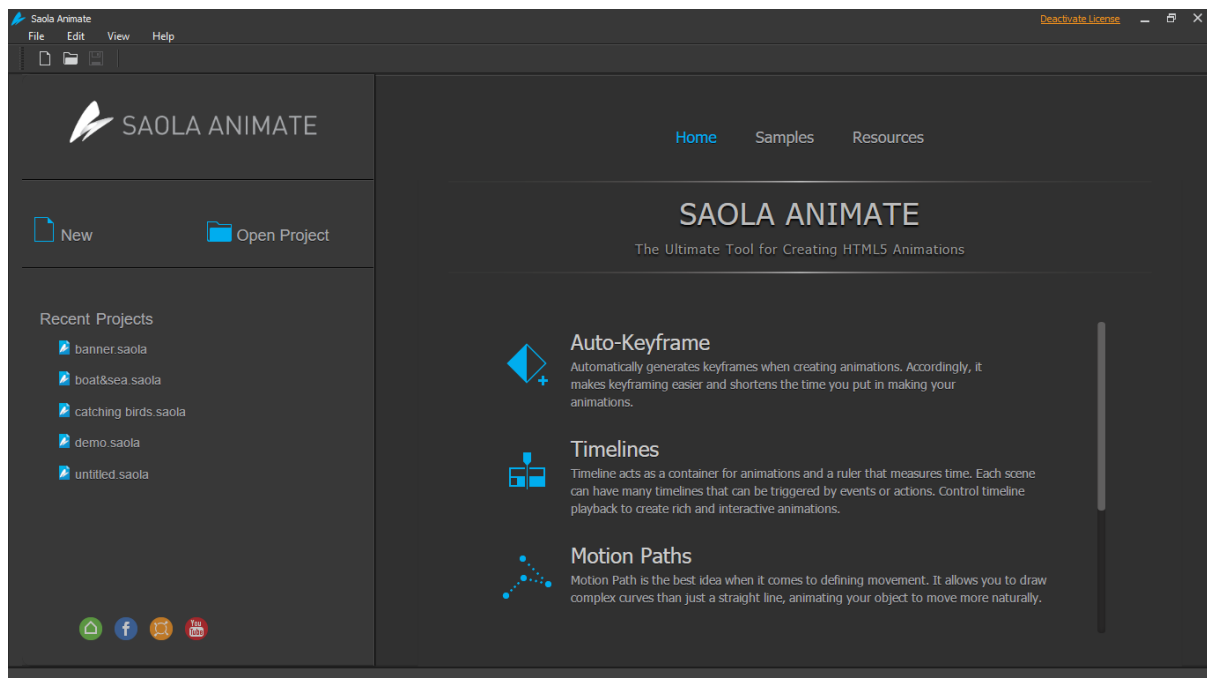
Saola Animate is an advanced application for designing and building HTML5 web content using an integrated visual. You can do almost everything in an intuitive editor with no coding required. Saola Animate enables you to create smooth animations and impressive visual effects using many types of elements and interactions.

Once you are done creating your content, export your final document with clean human-readable HTML5, CSS3, and JavaScript. The outputs have responsive layout that works well on any platform or device.

User Interface

Welcom Screen

The Welcome Screen is the first thing you see when launching Saola Animate. This screen is also displayed when no project opens. It allows quick access to a number of common tasks and links to many useful samples, demos, and other resources.



At the top is the menu bar and the toolbar where you can:

- Click **New** or **File > New** to **create a new project**.
- Click **Open Project** or **File > Open Project** to **open an existing project**.

- Click **Edit > Preferences** to **customize the application settings**.
- Click **View > Language** to **change the display language** in Saola Animate.
- Click **Help** to access information about the current version, open the documentation, check for updates, and open your account webpage where you can update your profile and manage your subscriptions.

On the left are two sections:

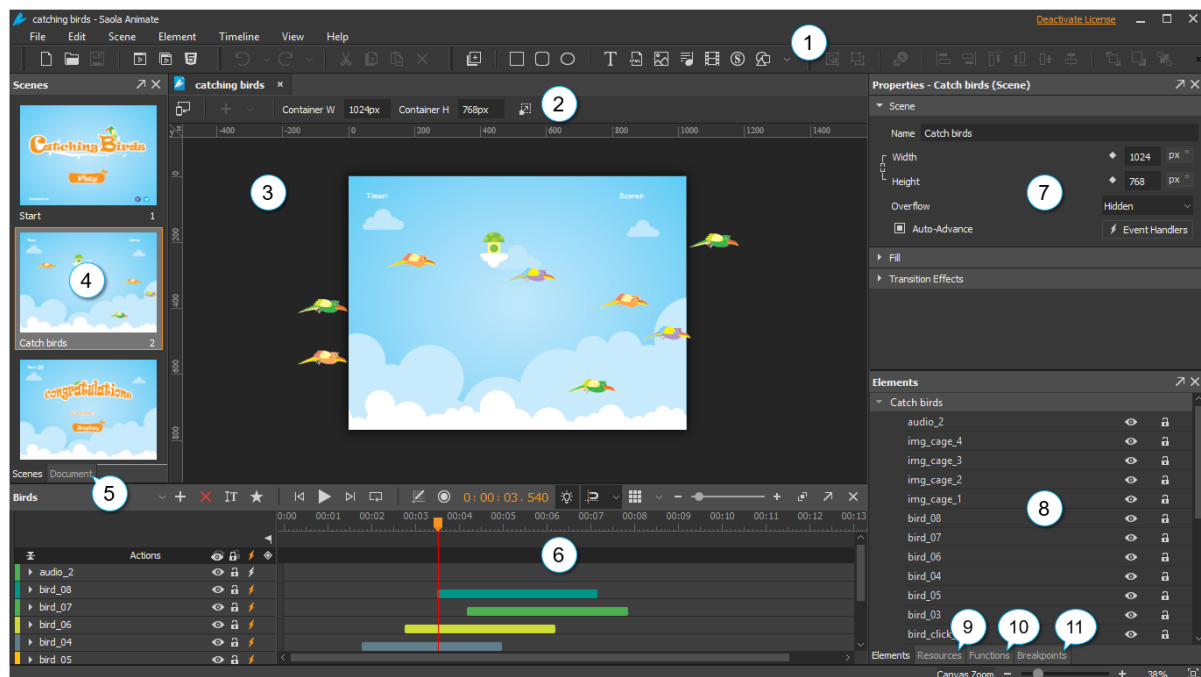
- The **Quick Start** provides two buttons to create new projects and open existing ones.
- The **Recent Projects** shows a list of projects you have recently been working with (if any). You can do one of the following:
 - Click a project or right-click > **Open** to open a project in the list.
 - Right-click a project > **Remove from List** to remove a project from the list.
 - Right-click a project > **Show in Folder** to open the containing folder of a project.

In the center are three tabs:

- The **Home** tab shows outstanding features which boost your creativity.
- The **Samples** tab shows a list of Saola Animate samples.
- The **Resources** tab provides links to **Saola Animate Homepage**, **User Community**, **YouTube Channel**, and **Online Demos**.

Workspace

Saola Animate provides a customizable workspace with a large work area for editing content. The central area is surrounded by a menu bar, a toolbar, a timeline, and a set of panes. You can resize the panes and toggle them on and off. You can also group them in tabs or make them free-floating on the screen, even outside the application window. See **Customizing Saola Animate** for more details.



1. The **Main Toolbar** contains buttons that act as shortcuts to the most frequently used commands from the menu bar.
2. The **Responsive** toolbar contains commands for creating and managing breakpoints and layouts in responsive documents.
3. The **Canvas** is the main work area in the center of the application window. It allows viewing, adding, modifying, and arranging elements.
4. The **Scenes** pane displays thumbnail views of scenes in a project.
5. The **Document** pane displays document properties and export options.
6. The **Timeline** pane contains animations, as represented in a time-based view. It allows organizing and controlling animations.
7. The **Properties** pane displays element properties and provides lists, text fields, and other controls to adjust and fine-tune element properties.
8. The **Elements** pane displays all elements included in a project. It allows viewing, changing the selection, locking, and toggling the visibility state of elements.
9. The **Resources** pane displays all resources in a project. It also holds any symbols you have created or imported, allowing you to edit symbols.
10. The **Functions** pane displays all JavaScript functions in a project. Create and manage functions to control various aspects of a document.
11. The **Breakpoints** pane displays breakpoints and layouts in responsive documents.

System Requirements

For Saola Animate, the following considerations apply:

- Operating system: Windows 7/8/8.1/10 (64-bit).
- Processor: Intel Pentium 4 or later.
- Memory: 4 GB or larger.
- Internet connection: Required.

Common Editing Commands

You are probably familiar with several common editing commands (cut, copy, paste, delete, undo, and redo) in Microsoft Office or other applications. In Saola Animate, these editing commands are also available to edit elements, scenes, keyframes, animation segments, etc. Find these commands on the main toolbar, in the **Edit** menu, or in the context menu of selected items.



Cut

The Cut command removes the selected items and places them on the Clipboard.

Toolbar shortcut: . Keyboard shortcut: **CTRL+X**.

Copy

The Copy command copies the selected items onto the Clipboard. While cutting a selection removes it from a scene, copying a selection leaves the original in place.

Toolbar shortcut: . Keyboard shortcut: **CTRL+C**.

Paste

The Paste command inserts the contents of the Clipboard at the current location. You can paste the Clipboard's contents back to any opening Saola Animate projects.

Toolbar shortcut: . Keyboard shortcut: **CTRL+V**.

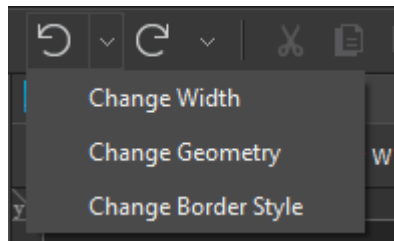
Delete

The Delete command deletes the currently selected items.

Toolbar shortcut: . Keyboard shortcut: **DELETE**.

Undo

The Undo command reverts the last change made to the document. Press the **Undo** button or **CTRL+Z** repeatedly if you want to undo multiple actions. If you want to revert several actions at once, click the down arrow next to the button and select the action in the list that you want to undo.



Toolbar shortcut: . Keyboard shortcut: **CTRL+Z**.

Redo

The Redo command redoes the most recent action that you have already undone. If you want to redo several actions, click the down arrow next to the button and select the action in the list that you want to redo. Note that the **Redo** button is only available after you have undone an action.

Toolbar shortcut: . Keyboard shortcut: **CTRL+Y**.


Chapter 2: Creating Projects

Creating a project is the first and most basic thing you need to do when working in Saola Animate.

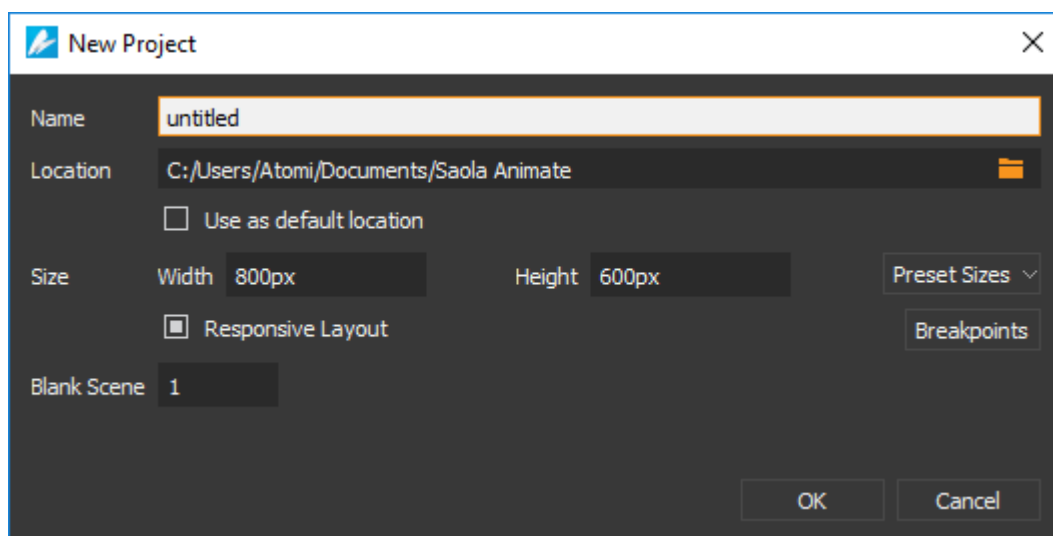
Creating, Opening and Saving Projects


Saola Animate stores a project in a folder whose name is based on the project title or project name. It also allows packaging project files into a package file (*.saolapack) for faster and easier sharing.

Creating Projects

On the Welcome Screen, click **New** to create a new project. Alternatively, in an opening project, click **New**  on the toolbar. You can also press **CTRL+N** to get the same result.

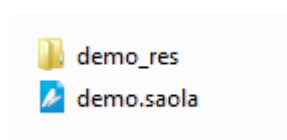
Fill in required information in the **New Project** dialog and click **OK** to submit.



- **Name:** Name of the project.
- **Location:** Location of the project. Once the project is successfully created, the project folder will be named based on the project title or project name. Enter a location or click **Browse**  to navigate to a location where you want to store your project.
- **Use as default location:** Make a location to be default every time you create a new project.

- **Width, Height:** Size of the container in which the document is embedded. The container size is editable afterward in the **Responsive** toolbar > **Container W/H**.
- **Preset Sizes:** Use a preset size to save time and ensure consistency when creating similar documents.
- **Responsive Layout:** Selected by default to create a **responsive document** in which the scene size starts at 100% of the container and the viewport width is set to the device width. Disabling this will create a non-responsive document, but later while editing you can still change the settings to create a responsive document.
- **Breakpoints:** Add and manage breakpoints before creating the document. Later while editing, you can still **manage breakpoints** as you wants.
- **Blank Scene:** Number of blank scenes you would like to initially create. Later you can insert new scenes at any time. The scene size, by default, is the size of the above-mentioned container.


Once the project is created, it will be stored in a folder with structure as shown below. The files are named based on the project title or project name.



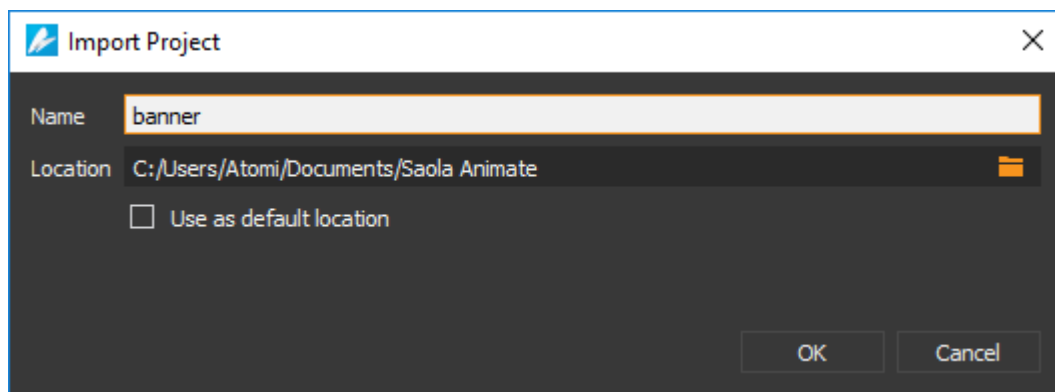
- **demo_res:** Folder that contains all resource files (images, audio, videos, etc.) that will be inserted into project during editing.
- **demo.saola:** Manifest file that stores information and data of the project.

Opening Projects


You can open existing projects from the Welcome Screen or while you're working on another project.

- Press **CTRL+O**, then browse and open a project.
- On the Welcome Screen, click the **Open Project** button. If the project you want is listed in the Recent Projects section, just click to open it.
- In an opening project, click **Open Project**  on the toolbar or click **File > Open Project** from the menu bar.

You can also import a project from a package file. Click **Open Project** and select a package file (*.saolapack). The **Import Project** dialog appears as below. Fill in the project name and click **OK** to import.



Saving Projects

After creating and editing a project, save it by pressing **CTRL+S** or clicking **Save**  on the toolbar.

You can also save a project as a file with a different name to make a backup of your file, for example. To do that, click **File > Save As > Other Project**. Fill in required information in the **Save As** dialog and click **OK** to submit.

To save a project as a package, click **File > Save As > Package**. Provide the file name for the package and click **Save** to submit.


Working with Scenes

Saola Animate documents can have one or multiple scenes which are managed in the **Scenes** pane. You can customize the initial number of blank scenes a project have when you **create a new project**.

Managing Scenes

Inserting Scenes

Do one of the following to insert a new scene:

- Click **Blank Scene**  on the toolbar.
- Click **Scene > Blank Scene** from the menu bar.
- Right-click a scene and select **Blank Scene**.

The new scene will be inserted below the last selected scene from which its size and background property will be inherited.

Duplicating Scenes

Duplicate a scene to create a copy which copies all aspects of the current one. To duplicate a scene, right-click it and select **Duplicate Scene**, or click **Scenes > Duplicate Scene** from the menu bar. The new scene will be inserted below the current one.

Copying Scenes

Scenes can be copied and pasted inside a current project or to another project. If you copy scenes across projects, all resources and user-defined functions used in the scenes will also be copied.

Removing Scenes

Select one or more scenes and press **DELETE** to delete them.

Displaying Scene Thumbnails

The **Scenes** pane displays scene thumbnails in two modes. Right-click the pane and select either of the modes:

- **Show in Column:** Scene thumbnails expand their size to fit into the available width.
- **Show in Grid:** Scene thumbnails are uniformly sized and displayed in grid.

Editing Scene Properties

Edit scene properties in the **Scene** and **Transition Effects** sections of the **Properties** pane. You can edit properties of multiple scenes simultaneously except the name of scenes.

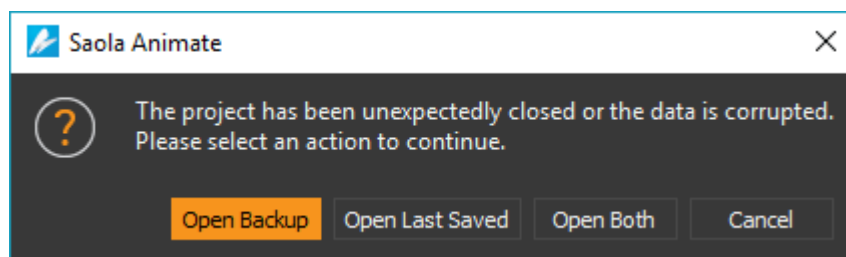
- **Name:** Unique name of the scene.
- **Width, Height:** Set the exact width and height of the scene. The size can be a fixed value (in pixels) or relative to the container size (in percentages).
- **Overflow:** Specify how scene content should be displayed if it extends beyond the bounds of the scene.
 - **Visible:** The overflow isn't clipped; it renders outside the scene's box. This is default.
 - **Hidden:** The overflow is clipped; any content that extends beyond the box will be hidden.
 - **Scroll:** The overflow is clipped, but a scroll bar is added to view the rest of the content.
 - **Auto:** If the overflow is clipped, a scroll bar should be added to view the rest of the content.

- **Auto-Advance:** Jump to the next scene automatically when the current scene ends. If you disable this feature, use **actions** to play scenes.
- **Transition Effects:** Movements from one scene to another.
 - **Type:** Type of the transition effect.
 - **Duration:** Duration of the transition in milliseconds.
 - **Easing:** The timing function that updates the transition by time.
 - **Function:** Direction of the Push transition effect.

Project Backup and Recovery

Auto-Backup enables you to recover your work if Saola Animate closes unexpectedly and your work has not been saved. By default, Saola Animate automatically creates a backup for a project every five minutes. You can customize the interval or turn off this feature using the **General** tab of the **Preferences** dialog.

When you open a project after an accidental closure or crash, a dialog appears as below:



- **Open Backup:** Open the backup version of the project. When editing, if you save your changes, this version will overwrite the last saved version.
- **Open Last Saved:** Open the last saved version of the project. The backup version will be deleted immediately.
- **Open Both:** Open both backup and last saved version to help you decide which version will be used. The backup version will be opened as a temporary project, meaning that it will be deleted if not saved.
- **Cancel:** Cancel the pending action.

Chapter 3: Responsive Design

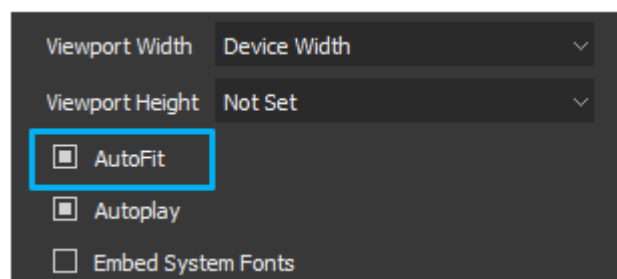
This chapter introduces some approaches that Saola Animate supports to make your document look good on any devices or screen sizes, from desktop monitors to mobile phones. You are not limited to just one method, but can combine any of them in one document to create your desired output content. Besides, these methods can be applied to the main document as well as any symbol documents.

Responsive Scaling

Responsive scaling means that the document is scaled automatically without changing its proportion to fit into the browser window. With responsive scaling, the document layout will be kept intact on all devices because the entire document will be scaled, not just some elements. This works like you're zooming in or out an image or video.

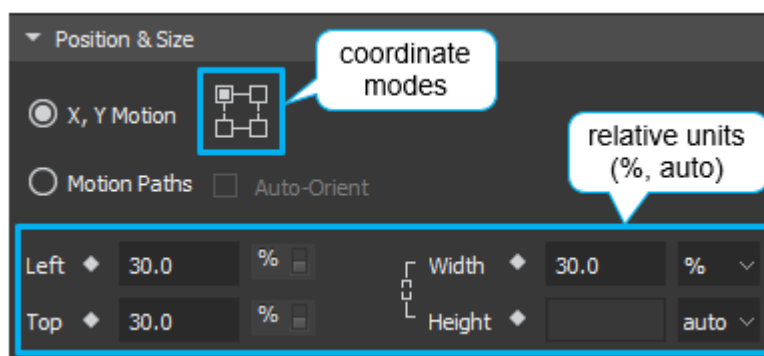
To make a document scale responsively:

1. Make sure the scene width and height is not set to 100%. 100%-size scenes cannot be scaled because they already fit into the browser window. Of course, if the document contains some 100%-size scenes, the responsive scaling still works well with the other scenes (which are not 100%-size).
2. In the **Document** pane, select the **AutoFit** check box.



Flexible Layout

Flexible layout uses relative units (percentage, auto) and different types of positioning (top-left, top-right, bottom-left or bottom-right) to adapt the document content to the browser window size.



Relative Positioning and Sizing

The position and size of scenes and elements can have absolute pixel unit (px) or relative percentage unit (%). Using percentage unit keeps the position and size of elements relative to their parents. For example, a top level element whose width is 30% will resize to ensure that its width is always 30% of the scene width. A scene whose width is 100% will expand to take up the entire viewport width. As a result, to make the position and/or size of a top level element respond to the viewport size changes, you must first set the scene size to % unit, then set the element position/size to % unit, too.

Maintaining Aspect Ratio

Regarding the element size in responsive documents, you can maintain the aspect ratio of elements by setting either the width or height unit to percentages and the other unit to **auto**. For example, set the width unit of an image to percentages and the height unit to auto. That way, the height of the image will change proportionally with the width while the width responds to the viewport size changes.

Pinning Element Position

Flexible layout allows pinning elements to the corners of the scene thanks to the coordinate modes. Elements can be set to relative to the top left, top right, bottom left, or bottom right corner of the scene, affecting how they adapt to the viewport size changes. When being positioned using percentage values, the relative distance from an element to the pinned corner of the scene will remain unchanged. For example, an element is positioned 30% away from both the top and the left edges of the scene will always keep its relative distance to these edges regardless of the viewport sizes.

The coordinate modes also affect how elements are resized. When you adjust the width or height of an element, it will shrink or grow to the opposing corner but not the pinned corner. For example, an element is positioned relatively to the bottom right corner of the scene.

When being enlarged, its bottom right corner stays where it is while the rest of the element grows.

Responsive Layout

Overview

Responsive layout is a truly responsive design method which allows you to create multiple different layouts for a single document. Each layout in turn will support responsive scaling and flexible layout methods. When viewing in a browser, the HTML5 player will load the suitable layout depending on the viewport size, and that layout will also be responsive to the viewport size if you've already made it scaled or flexible.

Breakpoints and Layouts

When a responsive site reaches a certain viewport width or height, it calls new element properties and animations for that width/height. **Breakpoints** mark the transition to a new layout where the element properties and animations change. You can create both horizontal and vertical breakpoints.

A range is the space between two breakpoints. If a scene has only one set of either horizontal or vertical breakpoints, each horizontal/vertical range will be equal to a **layout**. If a scene has two sets of horizontal and vertical breakpoints, a pair of horizontal and vertical ranges will define a layout.

For example, adding only one horizontal breakpoint at 500px creates two horizontal layouts: one is from 0px to 500px and the other extends right from 500px. Meanwhile, adding one horizontal and one vertical breakpoint creates two horizontal and two vertical ranges that intersect with each other. The four grids where these ranges intersect are equal to four layouts.

When the output document is displayed in a browser, the HTML5 player will choose the layout based on the current device's viewport size. Take the 500px horizontal breakpoint above as an example. Your output document has two layouts: the small layout below 500px and the large layout above 500px. If the current viewport width is smaller than 500px, the player will choose the small layout, otherwise it will choose the large layout. As the viewport width changes and passes 500px, the player will alter the layout, calling the new element properties and animations which fit the new width range.

Base Layout

Each scene has a base layout whose element properties and animations will be applied to all responsive layouts. In other words, element properties and animations in responsive layouts are inherited from the base layout. In a newly created responsive document, the base layout is activated by default so that you can work on it first. Once you get things done in the base

layout, you create the same set of element properties and animations in all responsive layouts. Now switch to the responsive editing mode to customize responsive layouts, creating major or minor changes compared to those of the base layout.

Setting the Viewport

Saola Animate allows you to set the viewport size for mobile browsers to give instructions on how the page should be rendered. For responsive projects, Saola Animate sets the viewport size to the size of the device by default to match the virtual viewport with the device dimensions. You shouldn't use this settings in non-responsive projects, unless you know exactly what you want.

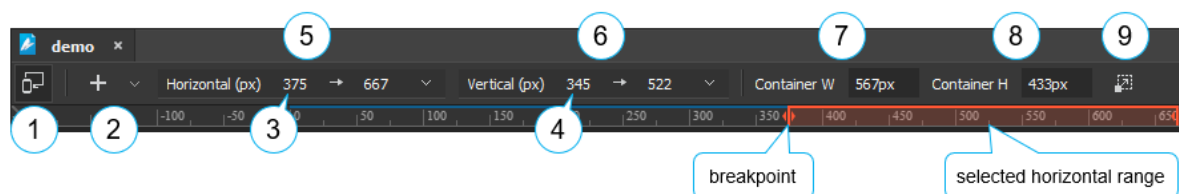
For details, see [Updating Document Properties](#).

Managing Breakpoints and Layouts

Responsive Toolbar and Breakpoints Pane

Work with the **Responsive** toolbar and the **Breakpoints** pane to manage breakpoints and layouts and navigate between layouts. If a scene has only one set of either horizontal or vertical breakpoints, each horizontal/vertical range will be equal to a layout. If a scene has both horizontal and vertical breakpoints, a pair of horizontal and vertical ranges will define a layout.

The **Responsive** toolbar is located above the Canvas and below the document tab.



- Click **Responsive** (1) to activate or exit the responsive editing mode.
- Click **Add Breakpoint** (2) > **Horizontal/Vertical** to add new horizontal/vertical breakpoints.
- Click **Add Breakpoint** (2) > **Horizontal/Vertical Preset** to add preset layouts.
- Click the horizontal/vertical breakpoint value (3, 4) to change the value (by scrolling the mouse wheel or entering a new value).
- Click the **Horizontal/Vertical Range** drop-down list (5, 6) to view the list of horizontal/vertical ranges. Click a range to select the corresponding layout and highlight it along the Canvas rulers. If you no longer want a range/layout, select it and click **Remove**.
- Click **Container W/H** (7, 8) to change the container width/height.

- Click **Adjust Container Size** (9) to adjust the container to see how the scene will look with different container sizes.

The **Breakpoints** pane displays breakpoints and ranges. Click a range to select it. You can click the **Base Layout** button to select the base layout.

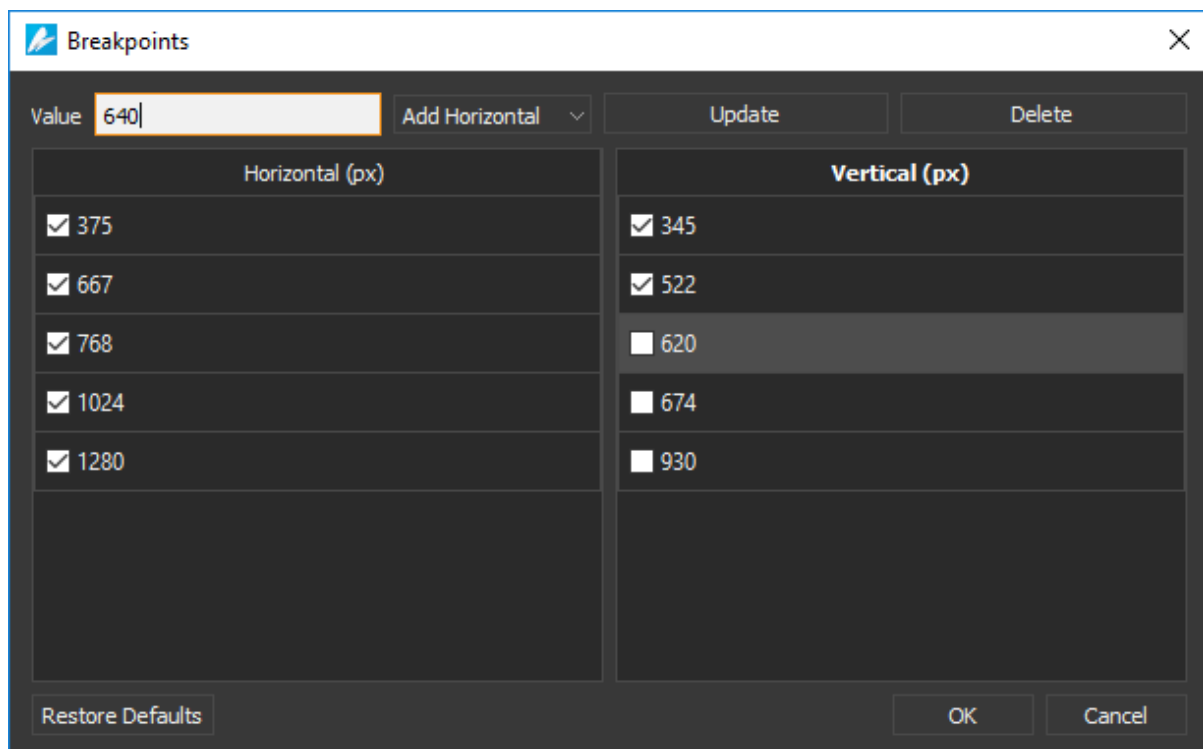
Breakpoints			↗ ×
Base Layout			
Horizontal (px)			
0	→	375	
375	→	667	
667	→	768	
768	→	1024	
1024	→	1280	
1280	→	...	
Vertical (px)			
0	→	345	
345	→	522	
522	→	620	
620	→	...	

Adding Breakpoints

You can add and manage breakpoints right from the start when you create a new responsive document, or later when you are editing the document.


a. Adding Breakpoints When Creating New Projects

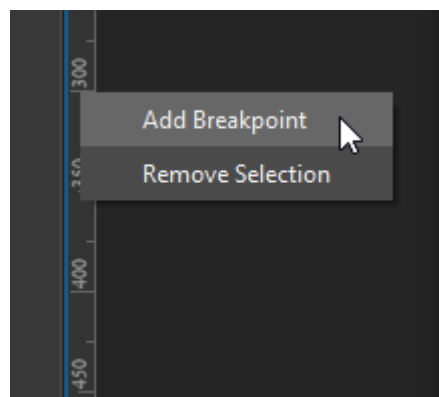
In the **New Project** dialog, click **Breakpoints** to open the **Breakpoints** dialog.



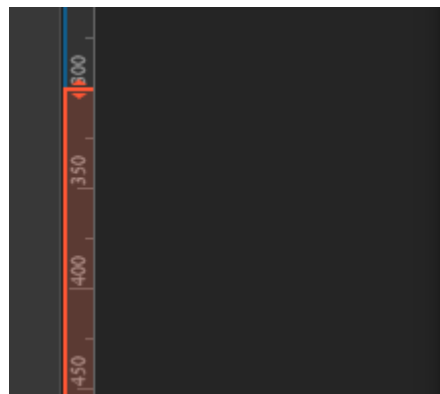
- To add a new breakpoint, enter a numeric value in the **Value** box. Then, click **Add Horizontal** to add a new horizontal breakpoint. Otherwise, click **Add Vertical** to add a new vertical breakpoint.
- To change an existing breakpoint, select an existing breakpoint and enter a new value in the **Value** box. Then, click **Update**.
- To delete a breakpoint, select it and click **Delete**.
- You can choose to not include a breakpoint in the document by clearing the check box next to it.
- If you want to go back to the original settings, click **Restore Defaults**.

b. Adding Breakpoints on the Canvas Rulers


1. Click **Responsive**  on the **Responsive** toolbar to switch to the responsive editing mode.
2. Right-click on either the horizontal or vertical ruler where you want to add a breakpoint.

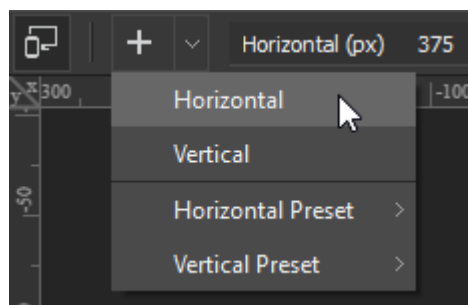


3. Select **Add Breakpoint** to add a new breakpoint. The range that extends right or bottom from the breakpoint until the next breakpoint is automatically selected for editing.

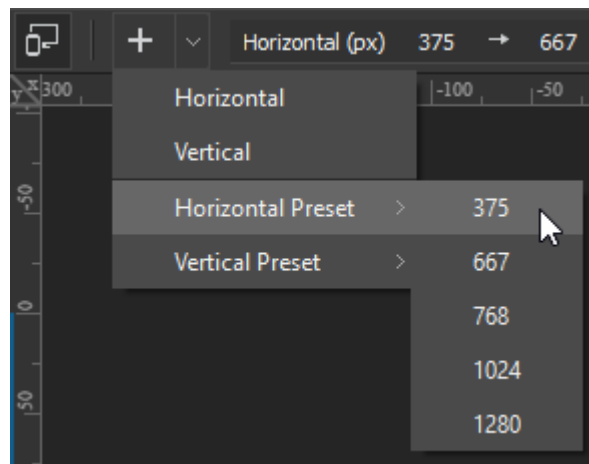


c. Adding Breakpoints with the Responsive Toolbar

1. Click **Responsive**  on the **Responsive** toolbar to switch to the responsive editing mode.
2. Click **Add Breakpoint** > **Horizontal** or **Vertical** to add a horizontal/vertical breakpoint. The range that extends right or bottom from the breakpoint until the next breakpoint is automatically selected for editing.



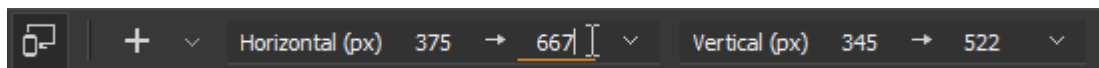
You can also select **Horizontal/Vertical Preset** to add a horizontal/vertical preset breakpoint.



Changing Breakpoints

Make sure the responsive editing mode is enabled.

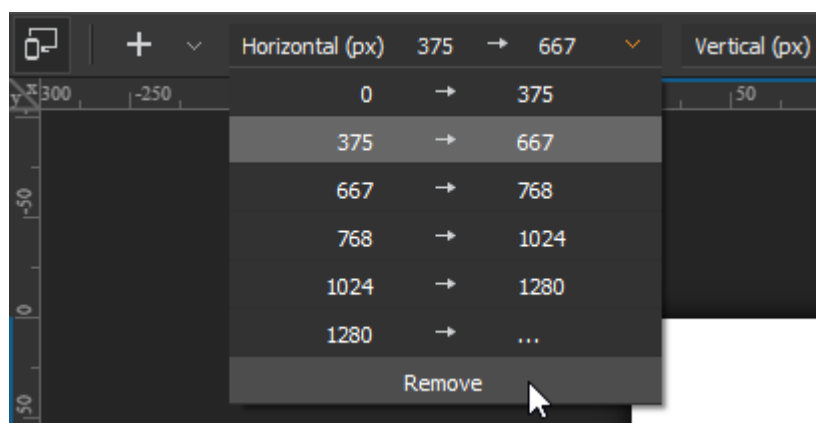
- Drag a breakpoint on the rulers to a new location.
- Click the breakpoint value on the **Responsive** toolbar, then scroll the mouse wheel to increase or decrease the value. You can also enter a new value.



Removing Breakpoints


Make sure the responsive editing mode is enabled.

- Right-click a range on the rulers > **Remove Selection** to remove the separate breakpoint that makes up the range, thus removing the selected range (and the corresponding layout). The other breakpoint shared between the recently removed range and another range will remain.
- Click the **Horizontal/Vertical Range** drop-down list on the **Responsive** toolbar, then click **Remove** to remove the current range (and the corresponding layout). This will remove the separate breakpoint that makes up the range while keeping the other breakpoint shared with another range.



Choosing a Layout to Edit

In the base layout, you can enter the responsive editing mode by one of the following:

- Click **Responsive**  on the **Responsive** toolbar.
- Click the range you want in the **Breakpoints** pane to both enter the responsive editing mode and select that range to edit. As mentioned earlier, a layout can be equal to a range or be defined by a pair of horizontal and vertical ranges.

In the responsive editing mode, to switch between layouts:

- Click the range/layout in the **Breakpoints** pane.
- Click the **Horizontal/Vertical Range** drop-down list on the **Responsive** toolbar and choose the range/layout you want.

To exit the responsive editing mode and back to the base layout, you can click **Responsive** on the **Responsive** toolbar or click the **Base Layout** button in the **Breakpoints** pane.

Managing Property and Animation Overrides

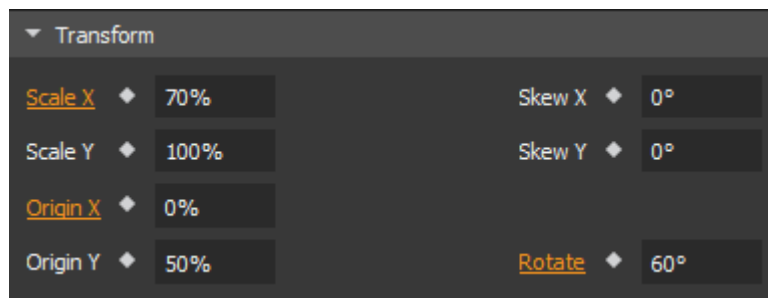
Creating Overrides

As said earlier, you should work in the base layout first and later customize the element properties and animations in any layouts if needed.

1. Make sure the responsive editing mode is enabled.
2. Select a layout and make changes in properties and animations.

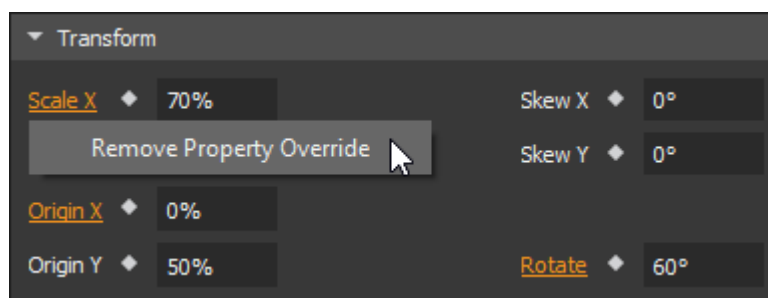
Overridden properties, animations, and the elements contain them are marked orange in the layout they appear. They are also marked in the base layout to indicate that there is at least one layout which isn't inherited from the base layout any more. Be aware that overridden properties and animations are no longer inherited from the base layout. Uniform properties,

animations, and elements across all the layouts (including the base layout) are not highlighted.



Removing Overrides

When you make a change in element properties and animations in responsive layouts, it overrides those in the base layout. You can remove overrides to revert the element properties and animations back to the default in the base layout. Just click any highlighted properties, animations, or elements and select **Remove Property/Animation Override**. If you want to remove all property and animation overrides in a layout, right-click the Canvas > **Remove All Overrides**.




- If the current layout is not the base layout, removing overrides will revert the selected property/animation in that layout only to the default in the base layout.
- If the current layout is the base layout, removing overrides will revert the selected property/animation in all layouts to the default in the base layout.

For example, a scene has a base layout and two responsive layouts (layout_1 and layout_2). The Left position of an element is 200px in the base layout and is overridden in layout_1 (changed to 50px) and layout_2 (changed to 100px). If you remove override of the element's Left property in layout_1, it will revert to 100px, meaning it is inherited from the base layout. Layout_2 won't be affected at all. If you remove override in the base layout, the element's Left property will be reverted to 100px in both the layouts.

Resizing the Viewport

Resizing the viewport while creating responsive content allows you to see how the output document will look in different container sizes. You can preview your responsive content right within the program.

In the base layout, do the following:

1. Click **Adjust Container Size**  on the **Responsive** toolbar. The light gray control handles appear on the edges and corners of the container.
2. Drag one of the control handles to change the container size.

Switching to the responsive editing mode, you can still use the **Adjust Container Size** button to adjust the container size. Alternatively, click anywhere on the Canvas rulers to snap the container to that pixel size.

Responsive Symbols

In responsive documents, **symbols** can respond to the container size changes. A responsive symbol has all breakpoints and layouts cloned from the parent symbol or the main document at the time it is created. After that, the breakpoints and layouts of the child and parent symbols no longer affect each other. Adding, changing, or removing breakpoints and layouts in the child symbol don't change those in the parent symbol, and vice versa. You can edit a responsive symbol as a normal responsive document. Just double-click the symbol element on the Canvas or the symbol in the **Resources** pane.

The size of the symbol element is the container size of the symbol embedded within it. For this reason, the symbol element size decides which layout will be used when the symbol runs in the output document. If the symbol element size is in the range of a layout, that layout will be selected. Thus, you can create a responsive symbol and reuse it in different responsive documents. Just resize the size of the symbol element to determine the layout in which the symbol will run.

Chapter 4: Inserting and Editing Elements

Elements are objects in scenes, including DIVs, text, HTML widgets, images, audio, videos, symbols, and shapes. Saola Animate offers various ways to insert and edit elements.

Element Overview

There are many types of elements that can be inserted into a scene. All element types have the same basic properties such as Name, Cursor, Overflow, Position, Transformation, etc. Meanwhile, some specific element types have their unique properties. See **Using Properties Pane** for details about element properties and how to animate them.

DIVs


Properties: Fill, Border, Corner, Text

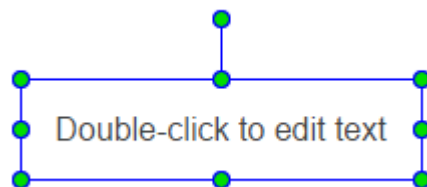
DIV element is displayed as an HTML <div> element. They come in three predefined shapes, including Rectangle, Rounded Rectangle, and Ellipse. These shapes are flexible and can be edited by adjusting the **Corner** property.

Insert DIV elements by clicking the **Rectangle** , **Rounded Rectangle** , and **Ellipse**  buttons on the toolbar.

Text

Properties: Fill, Border, Text

Text element displays text in a scene. Insert text elements by clicking **Text**  on the toolbar. Double-click the element to edit text inline.




Format or style text using the **Floating Text Format** toolbar or the **Properties** pane > **Text** section. You can change the font, size, color, styling, alignment, spacing, etc. For more styling control, edit the HTML for the text element.

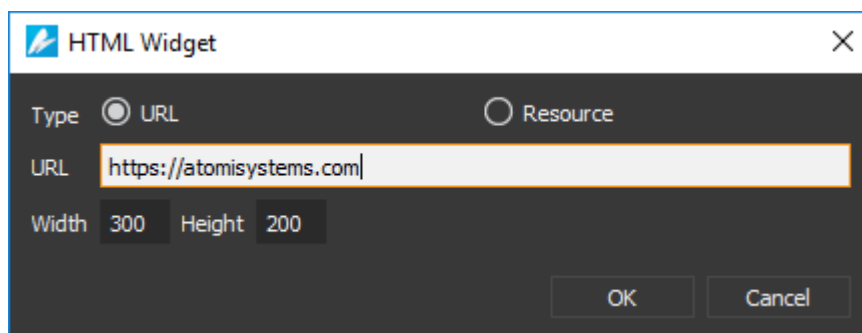
Saola Animate makes web safe fonts and system fonts available. When using system fonts, you can **embed fonts** in the HTML5 output using CSS3 Web Fonts Specification. Text will maintain appearance no matter the font installation on viewers' computer. Saola Animate also includes support for **adding Google fonts**. Added Google fonts will appear as font resources in the **Resources** pane.

HTML Widgets

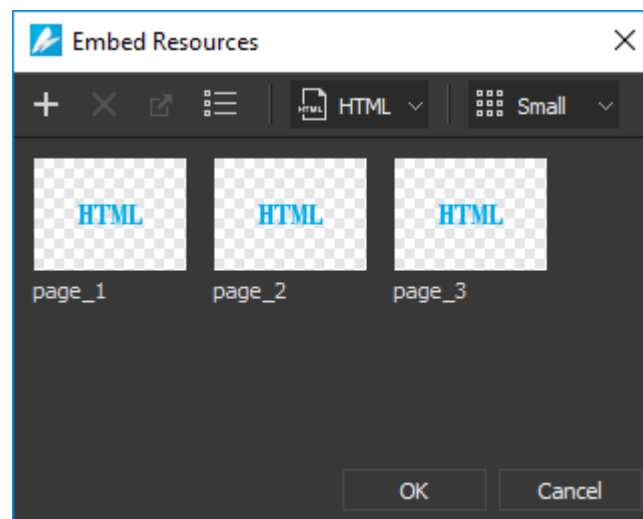
Properties: Border, Embedded

HTML widgets can be used to display webpage, embedded HTML, or SVG images in a scene. To insert an HTML widget, click **HTML Widget**  on the toolbar. Then, choose either of the two types to embed content.

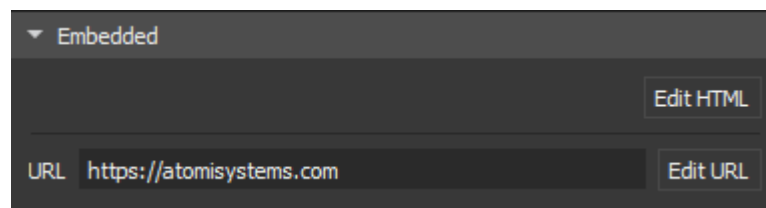
To display a webpage, select **URL** and enter the URL. You can change the size of the widget now or later.



To embed HTML, select **Resource** > **From Resources**. The **Embed Resources** dialog appears, allowing you to import or select an HTML resource. If you want to display an SVG image, import or select an SVG image resource from the dialog.



After inserting an HTML widget, you can edit the HTML or change the content by clicking the **Properties** pane > **Embedded** section > **Edit HTML** or **Edit URL**. Note that the HTML is always given priority. So, if you specify both the HTML and the URL of a webpage, the widget will show the embedded HTML. In that case, clear the HTML to display the webpage.



Images

Properties: Fill, Border, Image

Saola Animate supports six image formats, including JPG, JPEG, PNG, GIF, BMP, and SVG.


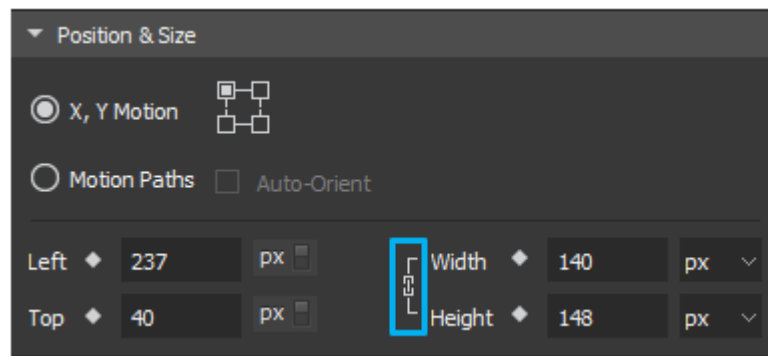
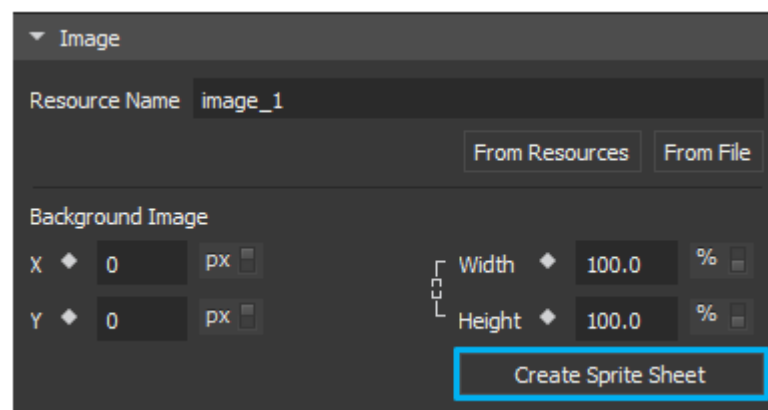
It allows importing images in more than one way. You can click **Image**  on the toolbar, drag and drop images onto the Canvas, or copy them from your computer and paste them there. You can also import images into the **Resources** pane to prepare and later add them to a scene.

Image elements maintain their aspect ratio by default, which keeps them proportional while being resized. To change the ratio, clear the **Link width to height** check box (the chain icon).



While editing images, chances are you want to revert them to their original size. Just right-click the images on the Canvas and select **Restore Original Size**. The option works at anytime without affecting images's other properties except the size property.

You can replace the image embedded within an image element while maintaining all existing animations. In doing so, make sure the new image is the same size as the original one. Otherwise, it may be resized or look messy. You can also animate the position and size of the embedded image or use it to **create sprite sheet animation**.





Audio and Videos

Properties: Media

Saola Animate supports the HTML5 <audio> and <video> elements, enabling native audio and video playback within the browser. You can import a wide variety of audio and video formats, including:

- **Audio:** MP3, WAV, M4A, AAC, OGA, OGG
- **Video:** MP4, M4V, WEBM, OGV

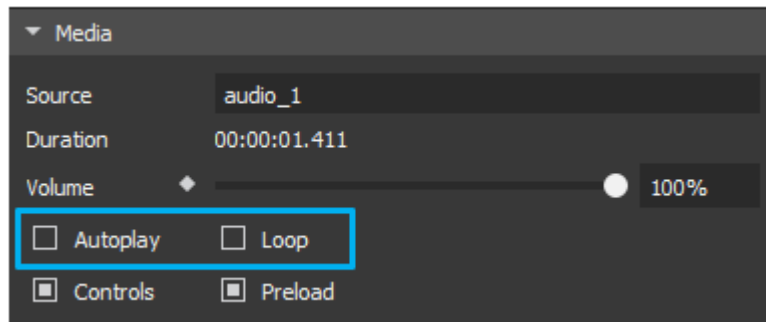
The ways to import audio and videos are similar to those used to import images. You can click **Audio**  and **Video**  on the toolbar, drag and drop files onto the Canvas, or

paste them there. By default, video elements also preserve their proportion while being resized. After resizing videos, if you want to revert them to their original size, right-click the videos on the Canvas and select **Restore Original Size**.

Controlling Playback

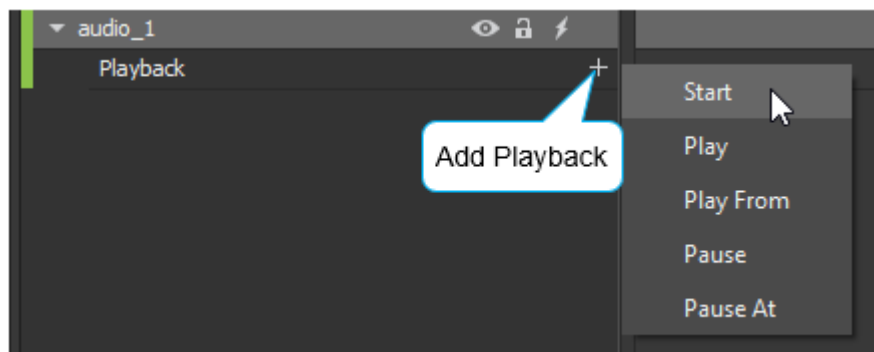
You can control audio/video playback using the following options.

- **Properties**



- **Autoplay:** Allow the audio/video to automatically play on the timeline and desktop browsers.
- **Loop:** Make the audio/video replay automatically from the beginning after it finishes.

- **Playback animations**



- **Start:** Play the audio/video from the beginning.
- **Play:** Resume the audio/video from the point where it has been paused.
- **Play From:** Resume the audio/video from a specific time.
- **Pause:** Pause the audio/video at the current time.
- **Pause At:** Jump to and pause at a specific time of the audio/video.
- **Media actions:** Use **actions** to control media playback through events.

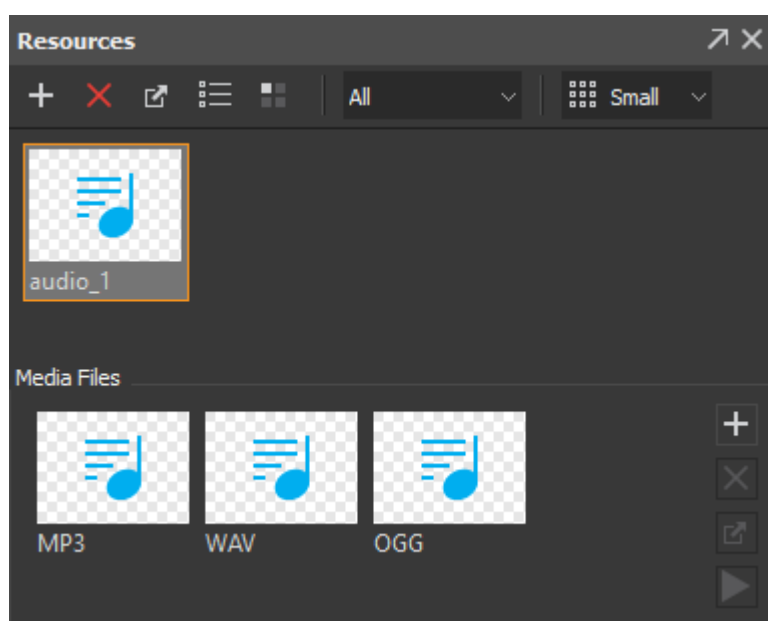
Besides, the **Media** section in the **Properties** pane lets you:

- **Controls**: Enable the default player controls.
- **Preload**: Preload audio/video before the document loads.

Resource Groups

HTML5 `<audio>` and `<video>` elements support multiple sources for one element. In other words, a single `<audio>` or `<video>` element contains more than one format, allowing a browser to pick the most compatible it supports. In Saola Animate, each audio/video comes with an audio/video group to which you can add other formats. When you run the HTML5 output in a browser, the browser will download and play the single most suitable format while ignoring the rest.


To add additional audio/video formats, first open the **Resources** pane. From here, select an audio/video and add formats to the resource group. That way, each audio/video resource has a resource group which contains the resource file and the fallback versions.



For modern browsers support HTML5 `<audio>` and `<video>` elements, including just MP3 for audio and MP4 for videos is enough. But for the broadest browser compatibility, you should provide MP3 and OGG files for each audio group and MP4 and WEBM for each video group.

For information about browser support for different audio formats, please read **Media formats for HTML audio and video** on MDN. You can also get the latest information about browser compatibility on caniuse.com.

Symbols


Symbol element displays a symbol in a scene. In other words, it is a copy of the symbol located on the Canvas. Insert symbol elements by clicking **Symbol**  on the toolbar.

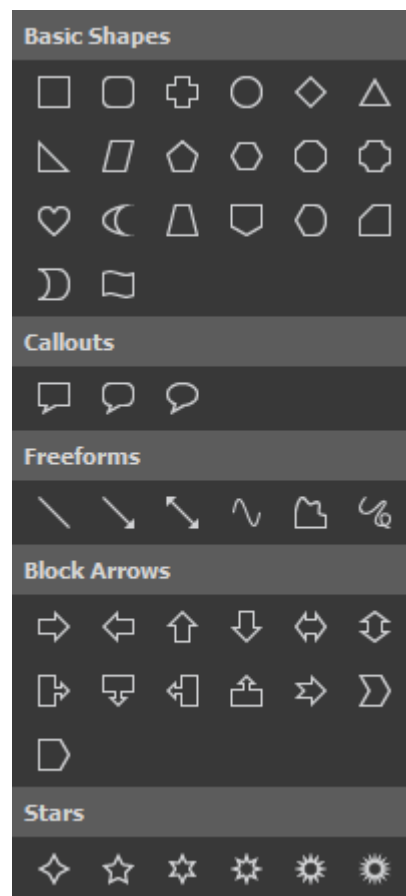
The size of the symbol element is the container size of the embedded symbol. A single symbol can be displayed by more than one symbol element. When you modify a symbol, all related symbol elements update automatically. That way, you can share identical elements across scenes or at different positions in the same scene. On the contrary, applying effects to a symbol element updates that element only. So, a symbol element can be different from the embedded symbol and other elements using the same symbol.

For details about symbols, see **Using Symbols**.

Shapes

Properties: Fill, Border, Text


Shape elements are the same as DIVs except that they have border and background using SVG images. Insert shapes using the **Insert Shape**  button on the toolbar. Shapes have many types as shown below:



Apart from predefined types, you can draw six types of custom shapes, including line, arrow, double arrow, curve, freeform shape, and freeform scribble.

- **Line, Arrow, Double Arrow:** Draw simple straight, point to point lines (with or without arrows).
- **Curve:** Draw curved lines or shapes.
- **Freeform Shape:** Draw lines/shapes that have both straight and curved segments (dual capability).
- **Freeform Scribble:** Draw lines/shapes as if you were using a pencil on paper.

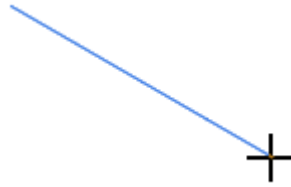
Drawing Freeforms

On the toolbar, click **Insert Shape** and select the drawing tool you want. As soon as you select the drawing tool, the cursor changes into a cross hair .

a. Drawing Lines

Click anywhere on the Canvas to set the start point. Move the mouse without releasing the mouse button to draw a line. While moving the mouse, hold down **SHIFT** to constrain the line

to 0, 30, 45, 60, 90, 120, 135, 150, and 180 degrees. Once you are satisfied, release the mouse button to set the end point.



b. Drawing Curves

Click anywhere on the Canvas to set the start point. Move the mouse to a different position to form a line. Click the mouse button to curve the line. As you move the mouse, the curve changes correspondingly. Now move the mouse and click the mouse button again to add turns to the curve. Each click equals a new curve.

A curve can be either open or closed. If you want an open curve, double-click to set the end point. Otherwise, move the mouse near the start point and left-click to get a closed curve filled with a solid color.



c. Drawing Freeform Shapes

The Freeform Shape tool allows you to draw both straight lines and curves. First, click anywhere on the Canvas to set the start point. To draw a straight line, release the mouse button and place the cursor elsewhere, then click. If you keep holding down the mouse button, you'll draw a freeform line. When drawing, you can alternate between straight and freeform lines within the same line shape.

Similar to drawing a curve, you can double-click at any time to set the end point, making an open freeform. Otherwise, move the mouse near the start point and left-click to get a closed freeform filled with a solid color.



d. Drawing Freeform Scribbles

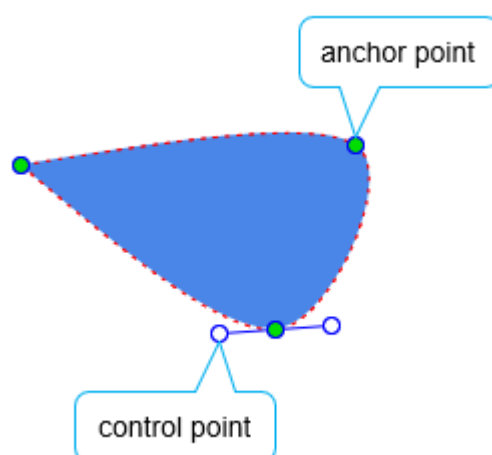
The Freeform Scribble tool is almost similar to the Freeform Shape tool except one essential difference. Once you start drawing with the Freeform Scribble tool, you have to keep holding down the mouse button until you finish. The point where you release the mouse button will be the end point of your drawing.



Editing Freeforms

There are many ways to change a shape's appearance after you draw it. First, right-click the shape and select **Edit Point** to open the editing mode. Now the shape is outlined by a red dashed line.

In the editing mode, you'll work with the anchor point and control point to alter the shape.



- **Anchor points:** Green circles within the outline which indicate a curve or a corner.

- **Control points:** White circles which appear at the end of a blue handle when you select an anchor point. When dragging a single control point, the remaining control point will move to maintain the angle between them. If you want to change the selected point only, hold down **CTRL** or **ALT** while dragging it.

If the anchor point and the control point have the same position, clicking will select the anchor point. If you want to select the control point, hold down **ALT** and click the mouse.

a. Adding and Deleting Anchor Points

- Click and drag the outline to add an anchor point.
- Hold down **SHIFT** and click an anchor point to delete it, making the shape change accordingly.

b. Opening and Closing Paths

- To open a closed path at an anchor point, right-click the anchor point and select **Open Path**. Otherwise, right-click anywhere within the path to open it at the start point.
- To close an open path, right-click it and select **Close Path**. A linear segment is added from the previous end point to the start point.

c. Making Straight Segments

Right-click a curved segment and select **Straight Segment** to straighten it.

d. Exiting the Editing Mode

Once you're done editing, click any elsewhere outside to exit the editing mode. Alternatively, right-click a shape, including its outline, anchor points, or control points, and select **End Edit Point**.

Groups

Properties: Fill, Border

Group elements are those containing other elements as children. You create group elements by **grouping multiple elements** in a scene. By doing so, you can easily move and animate multiple elements as if they were a single one.

To create a group, select more than one element and press **CTRL+G** or right-click and select **Group**.

Inserting Elements

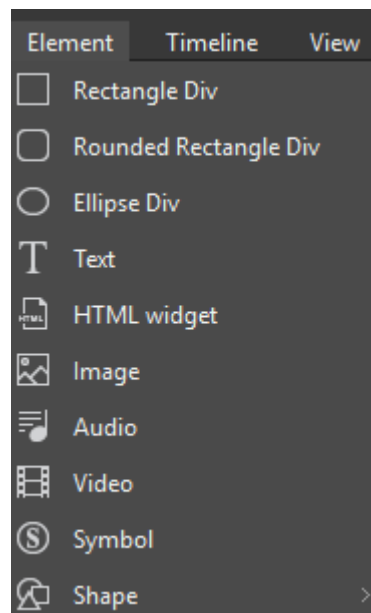
Inserting Elements

You have more than one way to insert an element into a scene. You can insert multiple elements at the same time.

- Directly drag or copy images, audio, videos, or symbols from your computer onto the Canvas.
- Click the Element buttons on the toolbar.



- Click the **Element** menu and select the element you want, or right-click the Canvas > **Element** > select the element you want.



- Drag resources from the **Resources** pane onto the Canvas, or right-click and select **Add to Scene**.

Note:

- If you want to insert a DIV or shape, click the corresponding buttons on the toolbar. Then, click and drag the mouse on the Canvas to define the position and size of the new element. The new element will be inserted when you release the mouse button.
- Text elements is inserted in the center of a scene with a placeholder text. Double-click the text element to enter text. The position and side of the element can be modified.
- For HTML widgets, specify required information in the **HTML Widget** dialog to insert them. For details, see **HTML Widget**.

After inserting an element, you may want to rename it. Double-click the element name in the **Timeline** or the **Elements** pane to enter a new name. You can also go to the **Properties** pane > **General** > **Name** and rename the element.

Copying Elements

You can copy multiple elements within a scene or across scenes and documents. Use shortcuts to quickly copy and paste elements. There are three options to paste elements:

- **Paste**: Insert the copied elements without animations. New elements will be placed near the position of the original elements or in the center of the viewport (if the original elements are placed outside the viewport).
- **Paste Here**: Insert the copied elements at the position of the cursor.
- **Paste with Animation**: Insert the copied elements with their animations.

If elements are copied and pasted across documents, all of their resources and JavaScript functions will also be copied and pasted to the target document. For example, if an image element is copied from a document to another document, its image resource will also be copied.

Removing Elements

Select one or more elements on the Canvas and press **DELETE** to remove them. You can also delete elements using the **Elements** pane.

Selecting Elements

Scene elements can be selected in many ways from many panes with the selection state synchronized among panes. Thus, if an element is selected on the Canvas, it'll also be selected in the **Timeline** and **Elements** panes.

Selecting Elements on the Canvas

On the Canvas, do one of the following:

- Select a single element by clicking it.
- Select multiple element: Use the rubber-band selector which appears when you click on a scene and drag the mouse. All elements that stay in or intersect with the selector will be selected.
- Select more elements: If you have already selected several elements and want to select more, hold down **CTRL** and click the additional elements. At the same time, if you want to cancel the selection of an already selected element, click it.

- Select child elements in **groups**: Select the group first, then click child elements to select them.

Selecting Elements in the Timeline Pane

Click the element name to select it. Alternatively, click the element's **keyframes** or **animation segments**. Note that in the **Timeline** pane you can only select and edit elements but not delete them.

Selecting Elements in the Elements Pane

Elements in the **Elements** pane are displayed in a tree view. Child elements are listed under and at the next indent level compared to their parent. You can click the triangle buttons to collapse or expand items.

Click an element to select it. If an element belongs to a group, when you select it, the group is also selected.

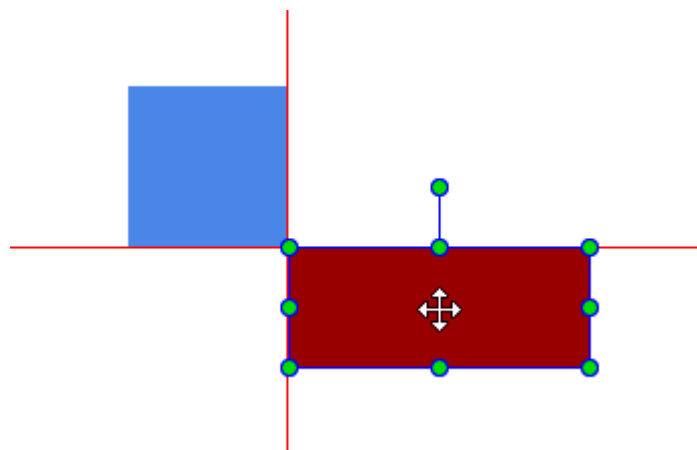
Arranging Elements

Aligning Elements

Saola Animate offers a variety of functions to align elements automatically and precisely.

Aligning Elements Using Canvas Snapping

Canvas Snapping snaps elements to the center or edges of the Canvas or other elements. This feature is indicated by red lines which appear when an element comes in the vicinity of the Canvas or another element.

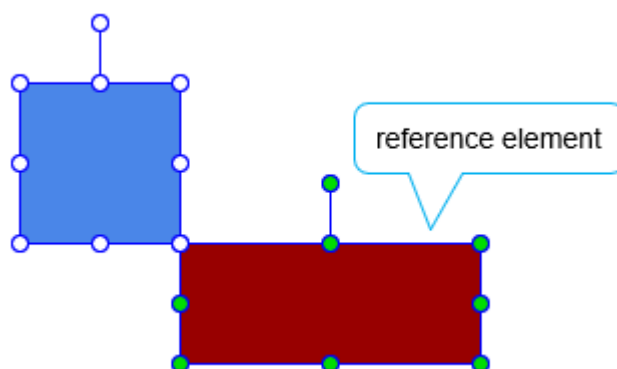


Canvas Snapping is turned on by default. To turn it off, click **View > Canvas Snapping**. You can temporarily disable this feature by holding down **CTRL** when moving or resizing elements.

Aligning Elements with Reference Element




You can align elements with a reference element. That way, the reference element stays fixed at the same position while all others move to align with it.

A reference element is the one with which all other elements align. To define a reference element, select all elements you want to align. The last selected one becomes the reference element with green handles rather than white handles like others. To define a different reference element, click any of the currently selected elements without holding **CTRL**.






Saola Animate provides six Alignment options, including both vertical and horizontal alignment, to align elements with a reference element. To set alignment, select at least two elements that have the same parent, then use one of the Alignment buttons on the toolbar or click **Edit > Alignment** from the menu bar.

Command	Icon	Remark
Align Left		Align the left side of all other elements with the left of the reference element. Their vertical position will not be affected.
Align Right		Align the right side of all other elements with the right of the reference element. Their vertical position will not be affected.
Align Top		Align the top of all other elements with the top of the reference element. Their horizontal position will not be affected.

Command	Icon	Remark
Align Bottom		Align the bottom of all other elements with the bottom of the reference element. Their horizontal position will not be affected.
Align Center		Align the center of all other elements with the vertical center of the reference element. Their horizontal position will not be affected.
Align Middle		Align the center of all other elements with the horizontal center of the reference element. Their vertical position will not be affected.

Resizing Elements to Reference Element

Besides six Alignment options, Saola Animate provides three Sizing options to quickly resize elements to a reference element. Select at least two elements that have the same parent, then click one of the Sizing buttons on the toolbar or click **Edit > Size** from the menu bar.

Command	Icon	Remark
Make Same Width		Match the width of all other elements with the width of the reference element.
Make Same Height		Match the height of all other elements with the height of the reference element.
Make Same Size		Match the width and height of all other elements with those of the reference element, respectively.

Ordering Elements

When elements overlap, z-order determines which one covers the other. You can reorder elements by clicking one of the four Display Order buttons on the toolbar or clicking **Edit > Display Order** from the menu bar. The options let you set the z-order of elements even when they do not overlap each other. So, you can bring an element to the front or back to easily identify and work with it.

For example, there are five shapes in the following order. Use Display Order options to change the z-order of the red shape.




Command	Icon	Remark
Bring to Front		Place the selected element at the very top of the stack.
Bring Forward		Send the selected element higher by one level in the stack.
Send to Back		Place the selected element at the very bottom of the stack.
Send Backward		Send the selected element lower by one level in the stack.

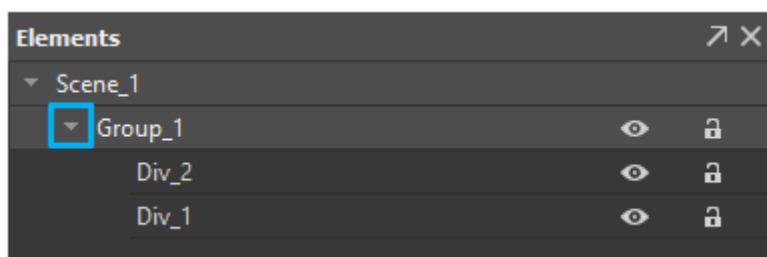
Grouping Elements

Grouping multiple elements as a single element lets you work faster and easier. For example, group several elements and they will stay together when being moved. Otherwise, you have to select all of the elements each time you want to change their position. Also, groups have properties like normal elements, meaning that you can animate their properties or add events and actions to them. You can group elements to animate them in the same way, instead of animating each with the same keyframes and properties.

Grouping Elements

1. Select two or more elements of the same parent.
2. Press **CTRL+G** or click **Group**  on the toolbar. Alternatively, right-click and select **Group**. The selected elements are grouped. A single box with sizing handles appears around the entire group. All moving animations of selected elements will be converted to a new coordinate system of a new parent.


The newly created group appears in the **Timeline** and **Elements** panes with a default name. Double-click its name to rename it. In the **Elements** pane, all child elements are listed under and at the next indent level compared to their group. Click the triangle buttons to collapse or expand these child elements.



After grouping elements, you can still edit child elements separately without breaking and affecting the group. On the Canvas, select the group first, then click the child element to start editing it. You can also click the child element name in the **Timeline** and **Elements** panes to select it.

Ungrouping Elements

Sometimes, you no longer need a group and want to break it apart.

1. Select a group.
2. Press **CTRL+SHIFT+G** or click **Ungroup**  on the toolbar. Alternatively, right-click and select **Ungroup**. The group will be broken apart into separate child elements.

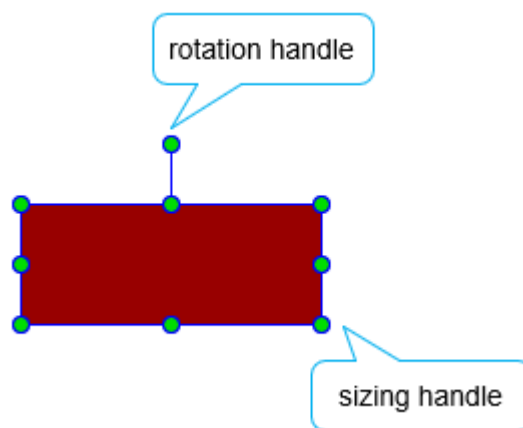
Note: If you want to add more elements to an existing group, ungroup that group and group all desired elements again.

Editing Element Properties

Element properties are displayed on the Canvas, but not all of them could be edited directly from here. Instead, go to the **Properties** pane to edit all properties of an element. You can select and edit multiple elements at the same time.

Editing Properties on the Canvas

On the Canvas, you can reposition, resize, and rotate an element using the sizing and rotation handles which appear when you select an element.



Editing Properties in the Properties Pane

All element properties are listed in sections in the **Properties** pane. Each section shows a group of related properties. For example, the **Fill** section provides four background fill styles for elements.

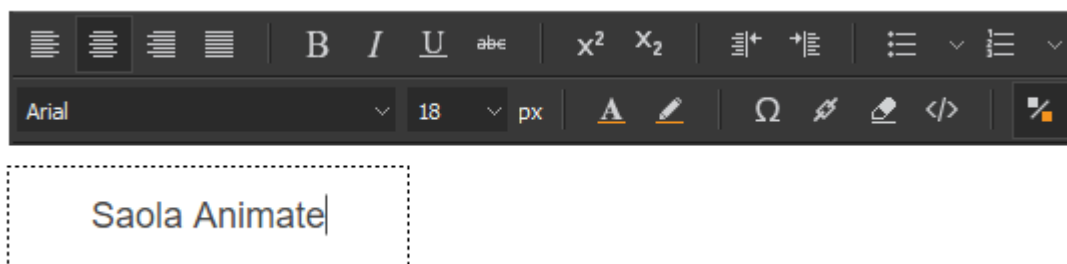
Property values are shown in some UI control types, such as spin boxes, sliders, text fields, etc., depending on the property types. When multiple elements are selected, there are two cases:

- If a property is unique, such as name, the control showing the property will be disabled. It is enabled only when you select one element.
- If a property isn't unique, the control will be enabled, but it only displays a value when the property values of all selected elements are the same. For example, when you select two elements with different Left values (in X, Y Motion relative to top and left),

the Left property shows no value. In that case, if you enter a value for the Left property, you'll set the Left value of both the elements.

Adding Text to Elements

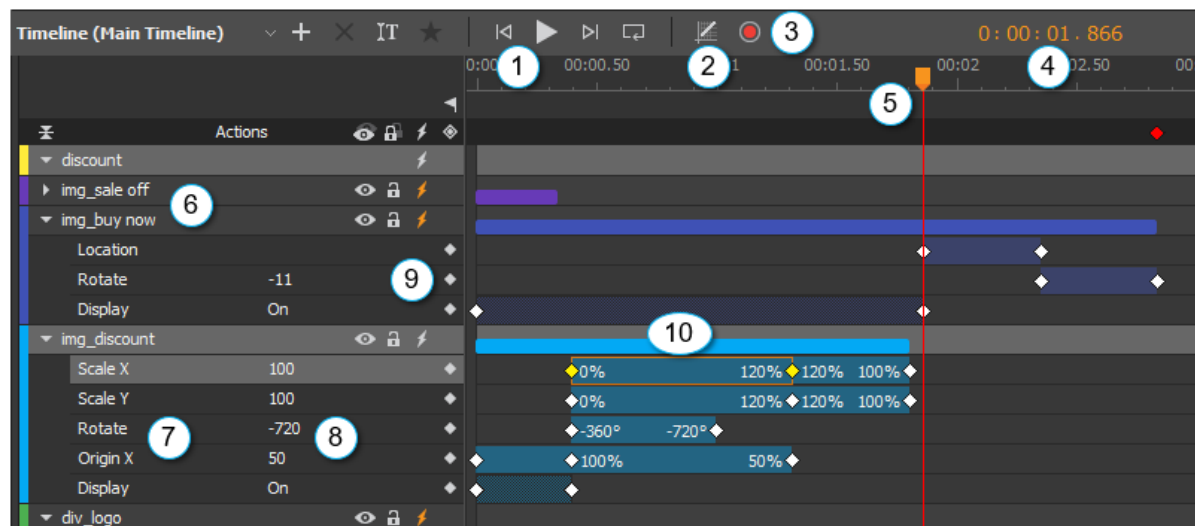
Text can be added to any text-based elements, including text, DIVs, and shapes. Insert these elements into a scene, then double-click them to open the inline text editor and type text. To stop editing text, click any elsewhere outside or select **Stop Editing** from the context menu.



Saola Animate provides numerous options for customizing text. You can change the font, size, color, styling, alignment, spacing, etc using the **Floating Text Format** toolbar or the **Properties** pane > **Text** section. You can also use fonts from your system or **Google fonts** to make your text stand out.

Chapter 5: Creating Animations

Creating animations is the most important and creative task when working in Saola Animate. Animations are organized and controlled in the **Timeline** pane. Think of this pane as a container for animations and a ruler that measures time in animations.



1. **Playback controls** (left to right): Go to Start, Play/Stop, Go to End, Loop Playback. The playback controls allow you to navigate in timeline and preview your animation within the editor.
2. **Easing button**: Customize the easing method which specifies the speed of the transition effect.
3. **Auto-Keyframe Mode button**: Toggle the Auto-Keyframe Mode to create keyframes automatically.
4. **Time control**: Display the current time, in other words, the current position of the Playhead (5). Click on the Time control to make it editable. Then, use the mouse wheel or enter a new value to adjust the current time.
5. **Playhead**: The orange Playhead and its red marker line moves as a document plays. It indicates both the current timestamp displayed on the Canvas and the current time in the Time control (4).
6. **Element name**: Elements on the Canvas are all displayed on the timeline.
7. **Property name**: Name of all properties that are animated.
8. **Property value**: Property value of properties at the Playhead position. Double-click it to make it editable, then you can change the property value.
9. **Add Keyframe button**: Add a new keyframe at the Playhead position.

10. **Animation segment:** An animation segment is composed of two keyframes defining its starting and ending values.
-

Keyframes

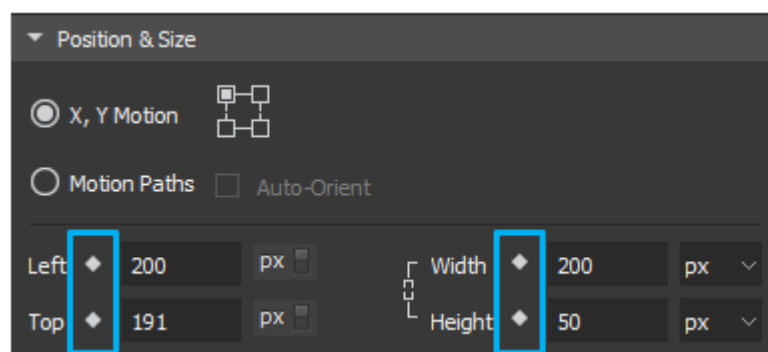
Keyframes specify the value of a property at a specific time. They are available throughout the **Properties** pane to set parameters of any properties that can be changed over time. When animating with keyframes, you add at least two keyframes with different values of the same property. The animation is formed based on the property change from the start to the end.

Adding Keyframes

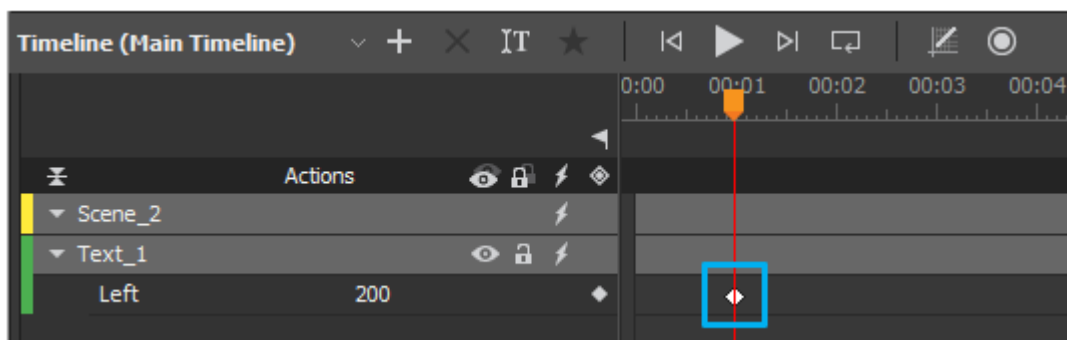
Keyframes can be added manually or automatically in several ways.

Using the Properties Pane

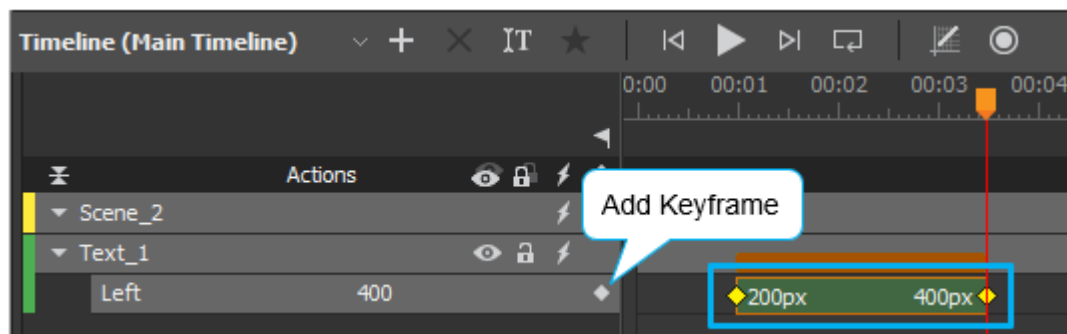
In the **Properties** pane, you can use small diamonds next to animatable properties to add keyframes. The keyframe will be visibly added at the Playhead position on the timeline.



To add the starting keyframe, first select an element. Then, position the Playhead to where you want the animation to begin, and click the keyframe diamond of a property. Now, set the property value of the starting keyframe. This will be the initial state of the element.



To add the ending keyframe, move the Playhead to where you want the animation to end. Add another keyframe for the same property and change the property value. A bar between the two keyframes appears, indicating that the property is animated.



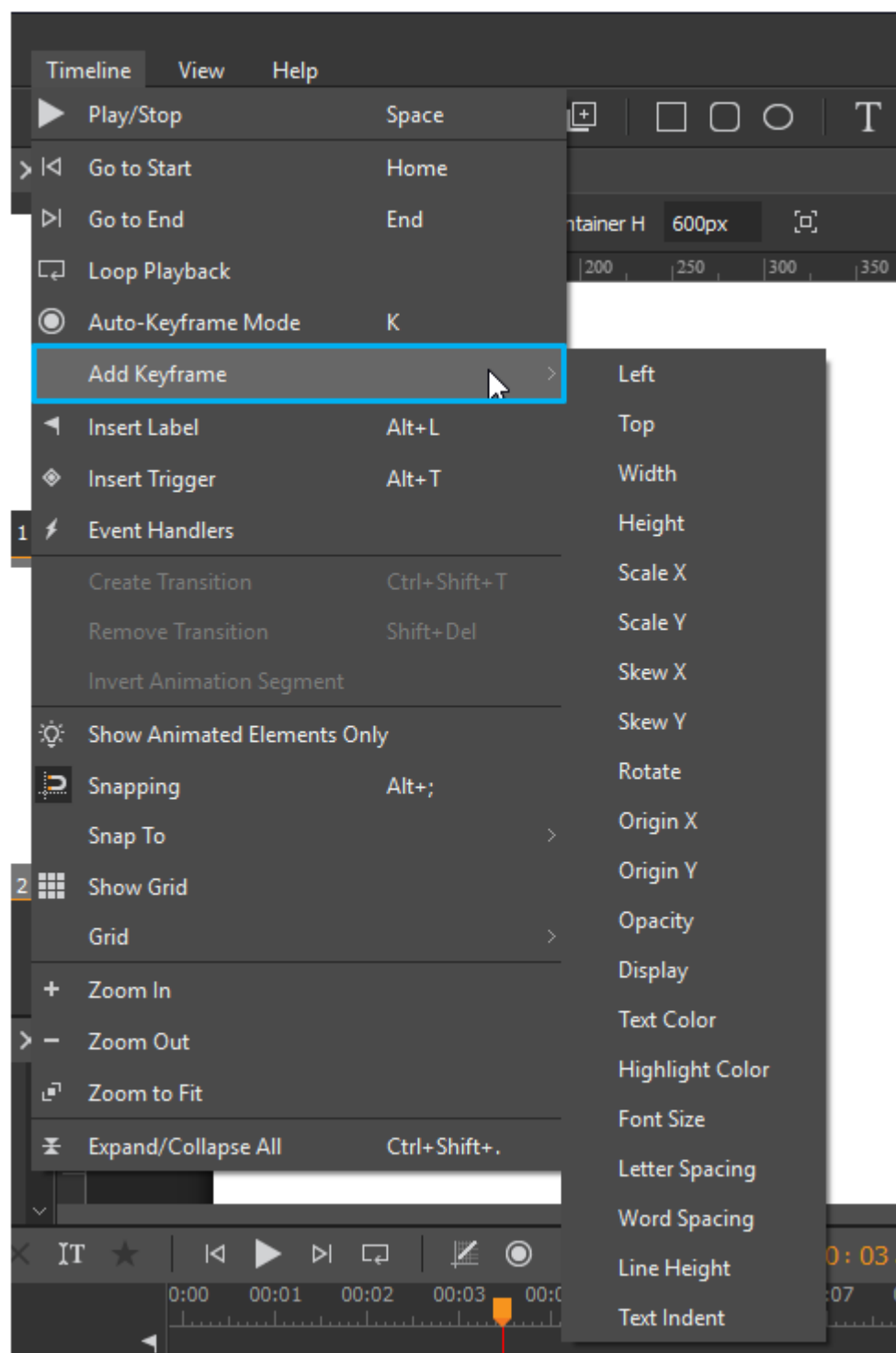
Keyframe diamonds also appear in the **Timeline** pane. Any property with at least one keyframe will be listed in this pane, allowing you to add more keyframes by clicking the **Add Keyframe** button. The best practice is use the keyframe diamonds the **Properties** pane to add the starting keyframes and use the diamonds in the **Timeline** pane for any other keyframes of the same property.

Using the Add Keyframe Menu

Another way to add keyframes is use the context menu. This option appears when you do either of the following:

- Select an element and go to the **Timeline** menu on the toolbar.
- Right-click an element on the Canvas, in the **Timeline** and **Elements** panes.

When you point to the **Add Keyframe** option, a list of available keyframes appears, allowing you to select the keyframe you want.



Creating Keyframes Automatically

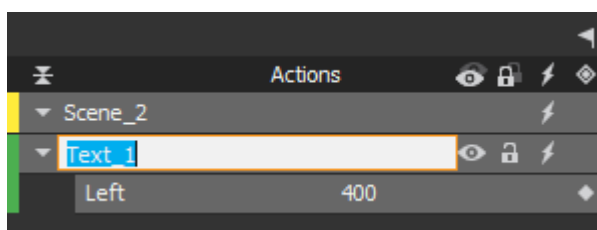
Auto-Keyframe Mode enables Saola Animate to generate keyframes automatically when you change the element properties. When being turned off, you have to add keyframes manually.

For more details, see **Auto-Keyframe Mode**.

Editing Keyframe Values

To edit a keyframe value, position the Playhead at the keyframe, then do one of the following:

- **In the Properties pane:** Adjust the property value using spin boxes, sliders, text fields, etc.
- **In the Timeline pane:** Double-click the value to make it editable and enter a new value. To finish editing, press **ENTER** or click any elsewhere outside.



- **On the Canvas:** For some certain properties such as Position, Size, and Rotate, you can change the value visually on the Canvas. For example, drag the rotation handle to adjust the Rotate value.


Note: If you place the Playhead at an existing keyframe and change the property value, the change is applied to that keyframe only. Meanwhile, if you place the Playhead at any other timestamps and change the property value, the change will be applied to all keyframes of the same property animation.

For example, the animation is [Left, duration: 2.5s, value: 200px - 400px].

- Place the Playhead at the ending keyframe and change the Left value from 400px to 500px. The element now moves 300px to the right instead of 200px as before. The ending point changes, leading to the change in the space the element moves.
- Place the Playhead at neither the starting nor the ending keyframes. Change the Left value to 500px. The element still moves 200px to the right as before, but it now moves from a different position to another different position. The space remains the same while the starting and ending points are shifted to the new positions.

Moving, Copying and Removing Keyframes

Moving Keyframes

You can move, copy, and delete multiple keyframes. When being dragged to move, keyframes keep their relative distance to each other. If you drag a keyframe to another, the animation segment has the same starting and ending keyframes. So, its duration becomes zero and the animation is displayed as a special keyframe  on the timeline.

By default, the **Timeline Snapping** mode is turned on, allowing keyframes to snap to particular tick marks or other timeline items. See **Snapping in Timeline** for more details.

Copying Keyframes

When copying keyframes, you can paste them onto the timeline of the same or of other elements as long as the target elements share the same properties with the source ones. For example, you can copy the Background Color keyframe between elements filled with solid color, not between an element filled with solid color and another element filled with gradient.

Removing Keyframes

To remove keyframes, click to select them so that they become highlighted, then press **DELETE**. If you delete adjacent keyframes, the animation segments between those keyframes will also be removed.

Auto-Keyframe Mode

Auto-Keyframe mode allows automatically recording keyframes when building animations. Simply enable the Auto-Keyframe mode, move the Playhead, and make an edit. Saola Animate itself will insert keyframes for any changed properties.

In the **Timeline** pane, click the **Auto-Keyframe Mode** button or press the **K** key to turn the mode on and record any changes as keyframes.



When you edit elements, the animation of the elements will be changed or remain the same depending on the two following cases:

Auto-Keyframe Mode On	Auto-Keyframe Mode Off
<p>If there is no keyframe at the current time, Saola Animate will create a new keyframe for the property that you change.</p> <p>If there exists a keyframe for a property at the current time and that property is modified, its value will be updated to the current value.</p>	<p>If there is no keyframe at the current time, all the keyframes will be changed as long as the corresponding property value is changed.</p> <p>If there exists a keyframe for a property at the current time and that property is modified, its value will be updated to the current value.</p>

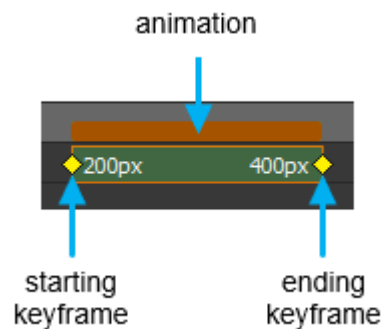
Be aware that automatic keyframing only creates keyframes when you make changes to properties. Let's consider the following scenario.

You insert an image onto the Canvas and want it to move across a scene. You enable the Auto-Keyframe mode and start animating. You move the Playhead to the third second, drag the image to another location, thinking this will create an animation segment. In fact, it won't because you make only one edit at the third second and create only one keyframe - not enough for an animation. No keyframe is added at the zero second to specify the starting position of the image.

So, always remember to create a keyframe at the beginning of the animation. As in the scenario above, the initial position of the image is perfect and you have nothing to change. But you must explicitly insert a keyframe for that position.

Animations


Animations are composed of two keyframes which define the starting and ending values of a property animation. An **animation segment** is the change between two keyframes - in other words, a segment defines the way a certain property changes. Meanwhile, a **transition** allows an element to gradually change from one state to another over the course of time. Hence, if there is no transition in an animation segment, there will be an instant change in the property value when the Playhead reaches the ending keyframe of the segment.



Animation Segments

Changing Timing and Duration

To change the timing of a segment, select the entire segment and drag it left or right. An orange border appears to indicate that that segment is selected.

To change the duration, select either the starting or ending keyframe and drag the keyframe. Longer segments make slower animations. If a segment has the same starting and ending keyframes, the duration is 0 and the animation is displayed as a special keyframe  on the timeline.

Inverting Segments

With any animation segment selected, right-click and select **Invert Animation Segment** to replace the starting keyframe with the ending keyframe and vice versa. This swaps the starting and ending values of the property animation.

Copying Segments

You can copy an animation segment and paste it onto the timeline of the same or other elements. That way, you copy both the starting and ending keyframes and the transition between them to the Clipboard.

1. Click a segment on the timeline. If you want to select multiple segments at the same time, hold down **CTRL** and click the segments.
2. Press **CTRL+C** or click **Copy** on the toolbar to copy the segments to the Clipboard. You can also cut the segments from the element by pressing **CTRL+X** or clicking **Cut** on the toolbar.
3. If you want to copy the segments to another element, select that element.
4. Move the Playhead to the position where you want the copied segments to start.
5. Press **CTRL+V** or click **Paste** on the toolbar to paste the segments onto the timeline.

Removing Segments

Select a segment and press **DELETE** to remove both the segment and its keyframes. If you select a keyframe and press **DELETE**, you will delete any segments composed of that keyframe.

Transitions

Suppose that the Left property of an element is animated with the starting keyframe [time: 1s, value: 50px] and the ending keyframe [time: 5s, value: 100px]. The transition between the two keyframes makes that element move smoothly from one position to another over five seconds.

Without the transition, the Left value will always be 50px at greater than or equal to 1s until the current time is greater than or equal to 5s. At that point, the Left value will immediately change to 100px, meaning that the element instantly jumps to a new position when the current time is equal to 5s. In other words, the Left property stays unchanged until the ending keyframe is encountered.

Creating Transitions

Right-click a segment and select **Create Transition** to make the element gradually change from one state to another over time.

Removing Transitions

Right-click a segment and select **Remove Transition** to create an instant change in the property value.

Customizing Easing Functions


Control the speed of a transition with CSS easing functions. For more details, see **Easing**.

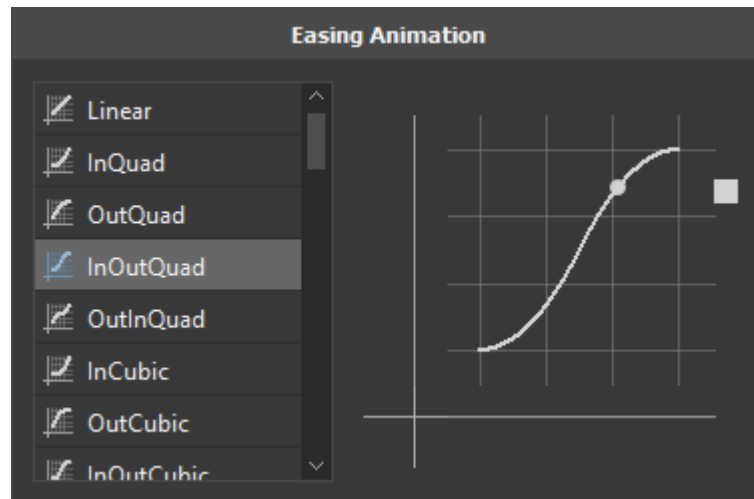
Easing

The motion in nature is rarely plain and linear from one point to another. In reality, when moving, things almost never move at an unchanging speed. Instead, they tend to accelerate or decelerate. For example, when a bus comes to a stop, it slows down before a full stop. When setting off again, it speeds up to a full speed. So, to mimic realistic dynamic movements, use CSS easing functions to make transitions change speed over time.

By default, Saola Animate uses Linear easing to define a **transition** with the same speed, in other words, no acceleration from start to end. For more dynamic movements, Saola

Animate offers up to 40 CSS easing functions. To customize easing, in the **Timeline** pane, do the following:

1. Select one or more animation segments.
2. Click the **Easing**  button or right-click > **Easing...** from the context menu. Both open the **Easing Animation** dialog which lists all easing functions.
3. Select an easing from the dialog.



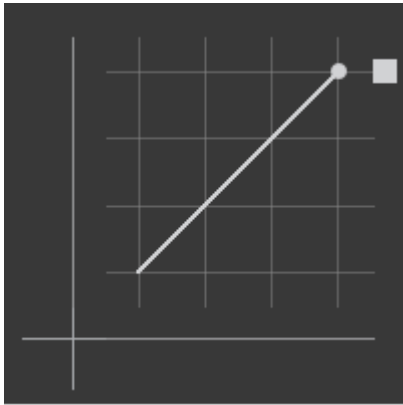
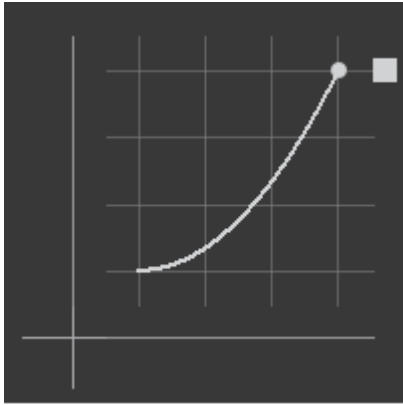
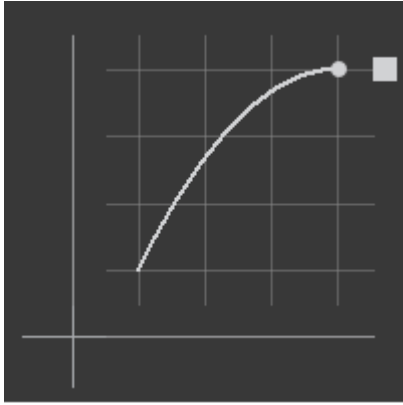
Easing functions in Saola Animate come in ten types:

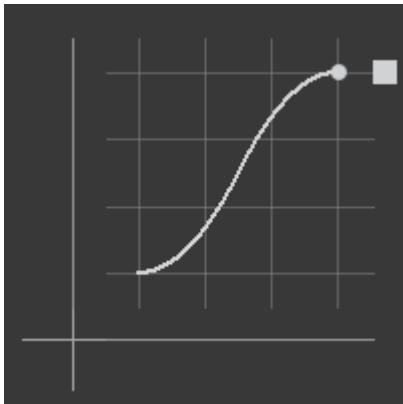
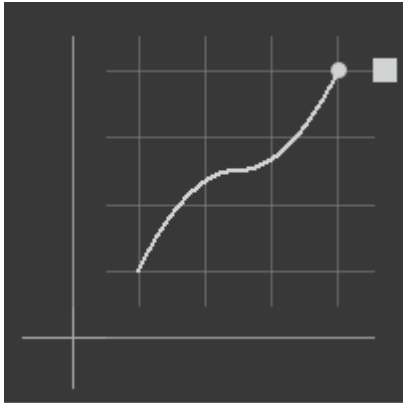
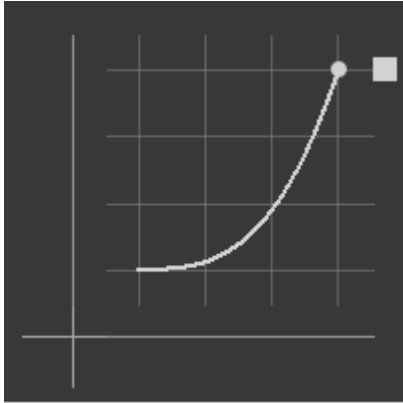
1. **Quad**: Display a quadratic transition.
2. **Cubic**: Display a cubicular transition.
3. **Quart**: Display a quartetic transition.
4. **Quint**: Display a quintic transition.
5. **Sine**: Display a sinusoidal transition.
6. **Expo**: Display an exponential transition.
7. **Circ**: Display a circular transition.
8. **Elastic**: Display an elastic transition that looks like a spring swinging back and forth until coming to rest.
9. **Back**: Display a cubicular transition with backward motion, overshoot, and settle.
10. **Bounce**: Display a parabolic bounce, simulating the motion of an object bouncing.

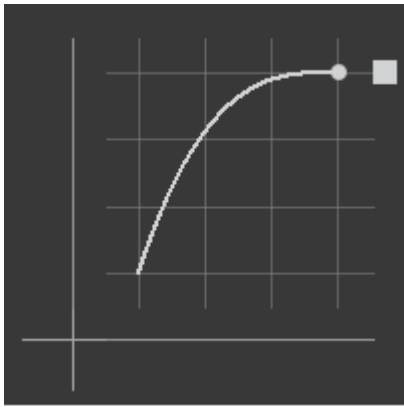
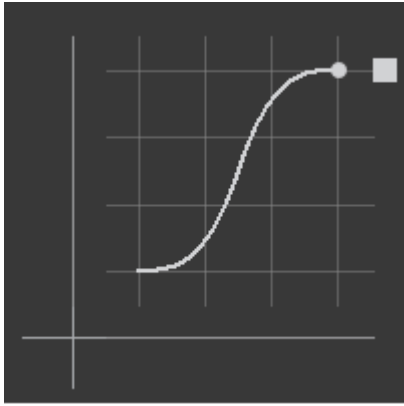
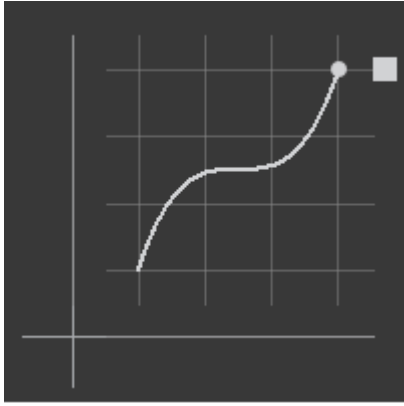
Each type has four values:

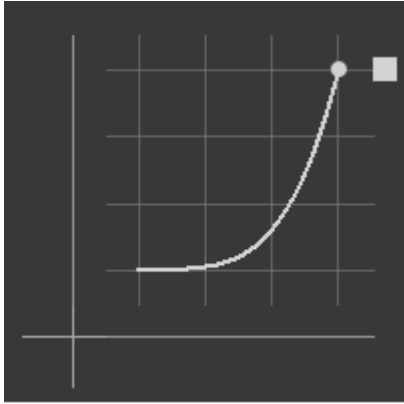
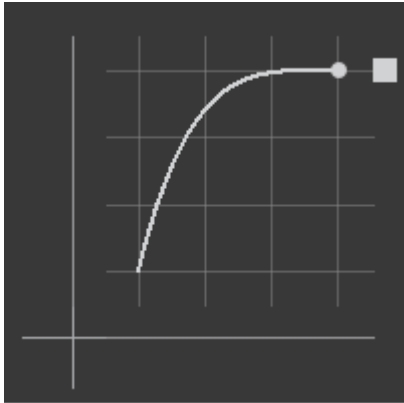
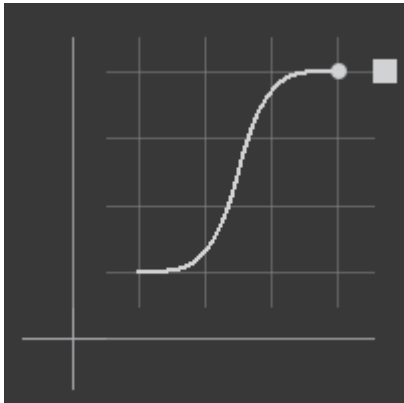
1. **Ease in**: Speed up from zero speed, then finish at full speed.
2. **Ease out**: Start at full speed, then slow down to zero speed.
3. **Ease in-out**: Start slowly, fastest in the middle, then finish slowly. In other words, the transition speeds up until halfway, then slows down.

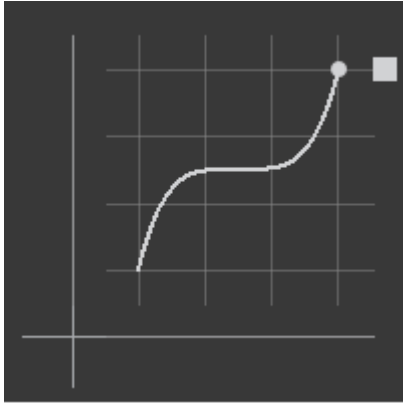
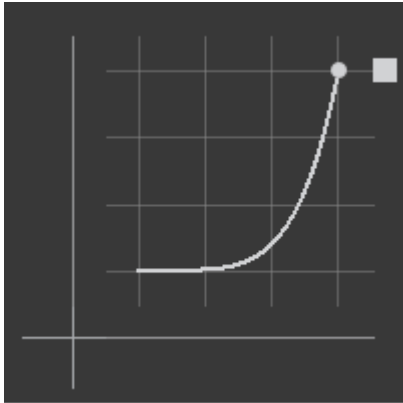
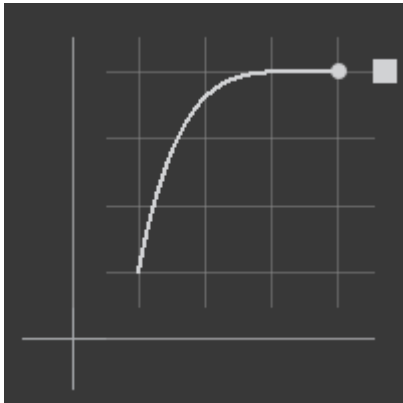
4. **Ease out-in:** Start and end fast, but slow in the middle. In other words, the transition slows down until halfway, then speeds up.

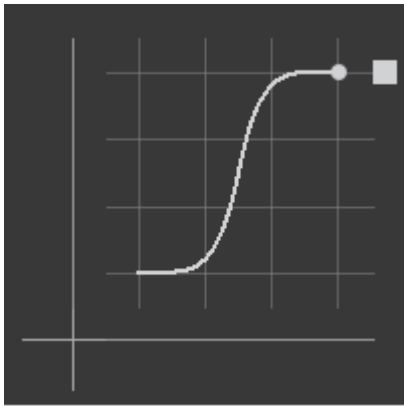
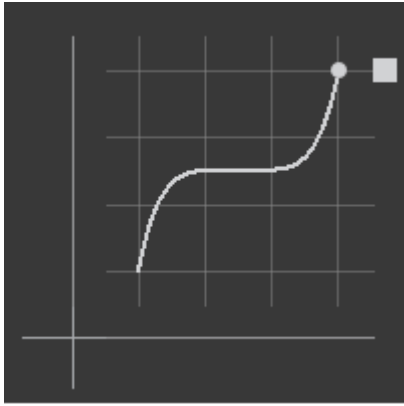
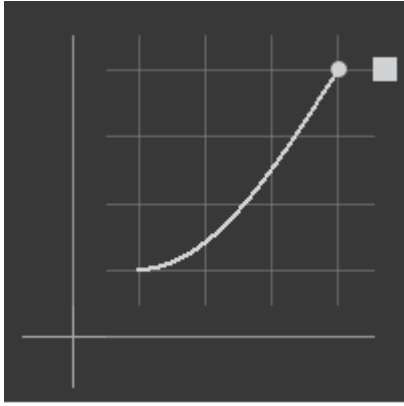
Constant	Illustration	Remark
Linear		Easing curve for a linear(t) function: velocity is constant.
InQuad		Easing curve for a quadratic (t^2) function: accelerating from zero velocity.
OutQuad		Easing curve for a quadratic (t^2) function: decelerating to zero velocity.

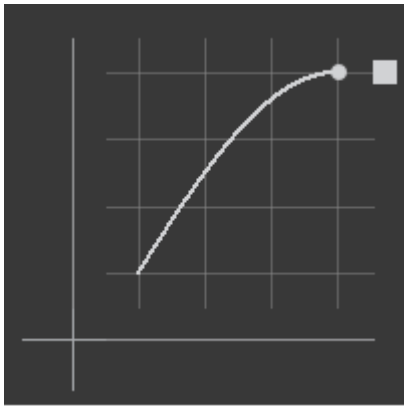
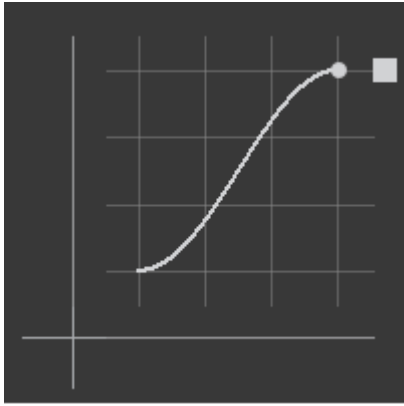
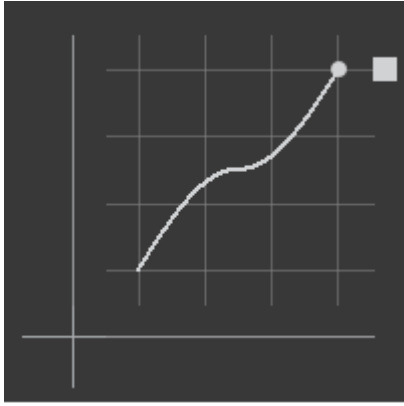
Constant	Illustration	Remark
InOutQuad		Easing curve for a quadratic (t^2) function: acceleration until halfway, and then deceleration.
OutInQuad		Easing curve for a quadratic (t^2) function: deceleration until halfway, and then acceleration.
InCubic		Easing curve for a cubic (t^3) function: accelerating from zero velocity.

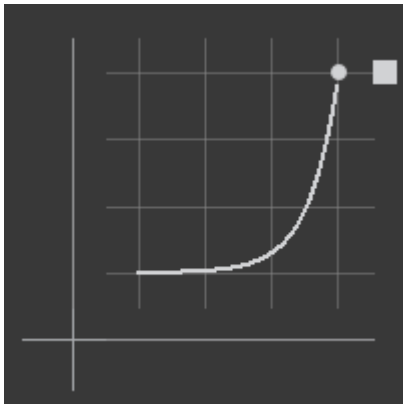
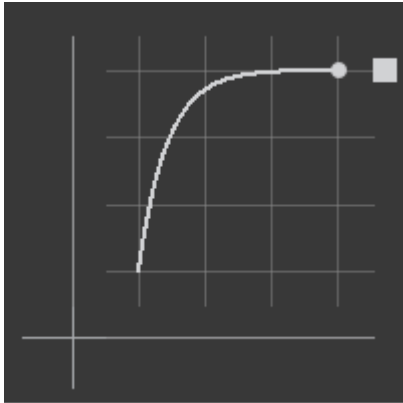
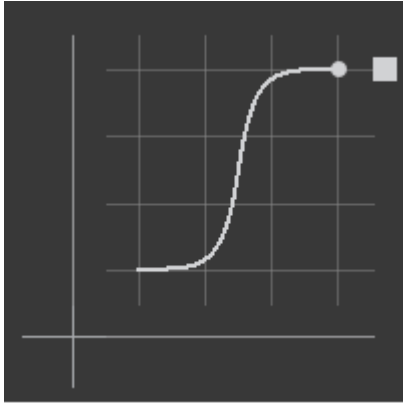
Constant	Illustration	Remark
OutCubic		Easing curve for a cubic (t^3) function: decelerating to zero velocity.
InOutCubic		Easing curve for a cubic (t^3) function: acceleration until halfway, and then deceleration.
OutInCubic		Easing curve for a cubic (t^3) function: deceleration until halfway, then acceleration.

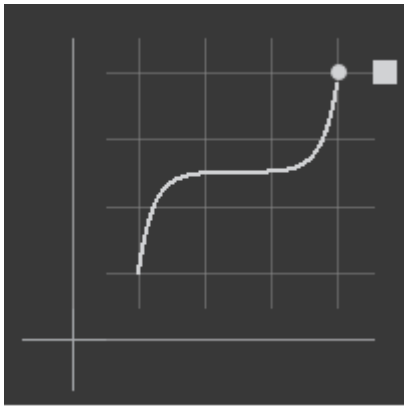
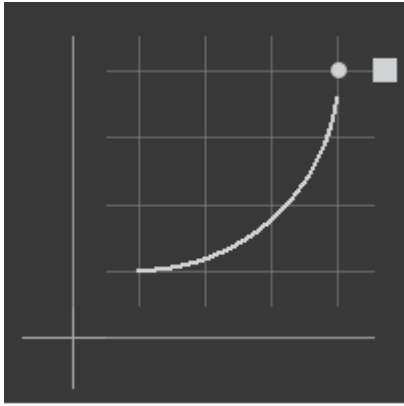
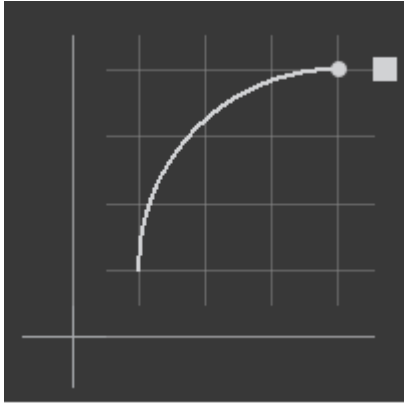
Constant	Illustration	Remark
InQuart		Easing curve for a quartic (t^4) function: accelerating from zero velocity.
OutQuart		Easing curve for a quartic (t^4) function: decelerating to zero velocity.
InOutQuart		Easing curve for a quartic (t^4) function: acceleration until halfway, and then deceleration.

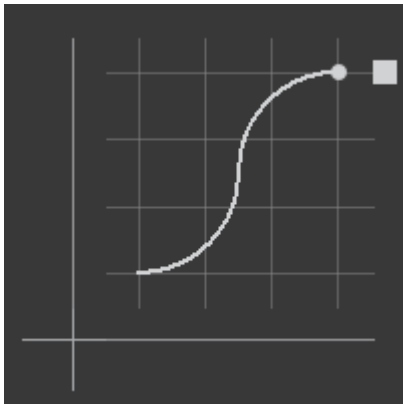
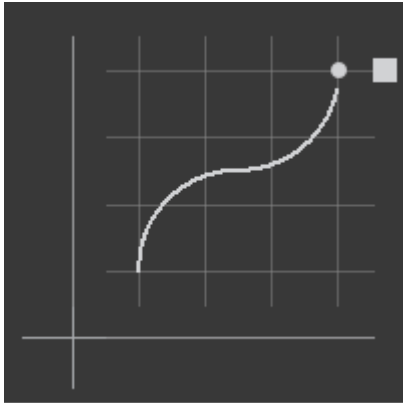
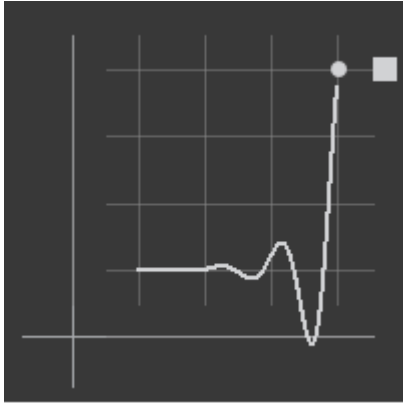
Constant	Illustration	Remark
OutInQuart		Easing curve for a quartic (t^4) function: deceleration until halfway, and then acceleration.
InQuint		Easing curve for a quintic (t^5) easing in: accelerating from zero velocity.
OutQuint		Easing curve for a quintic (t^5) function: decelerating to zero velocity.

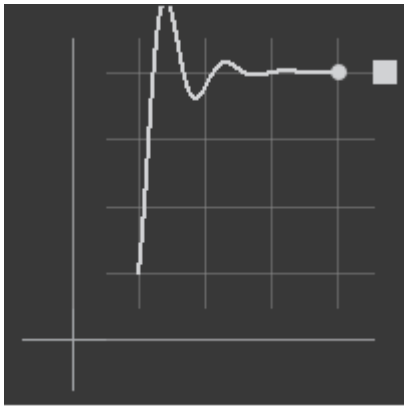
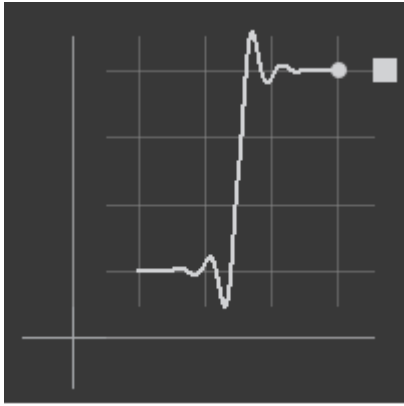
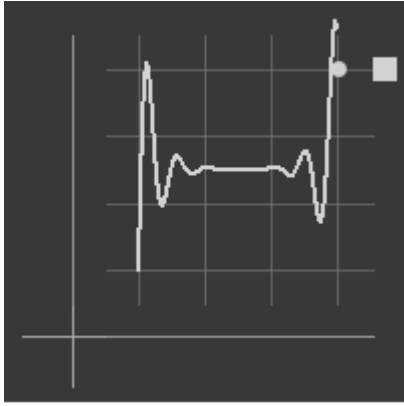
Constant	Illustration	Remark
InOutQuint		Easing curve for a quintic (t^5) function: acceleration until halfway, and then deceleration.
OutInQuint		Easing curve for a quintic (t^5) function: deceleration until halfway, and then acceleration
InSine		Easing curve for a sinusoidal ($\sin(t)$) function: accelerating from zero velocity.

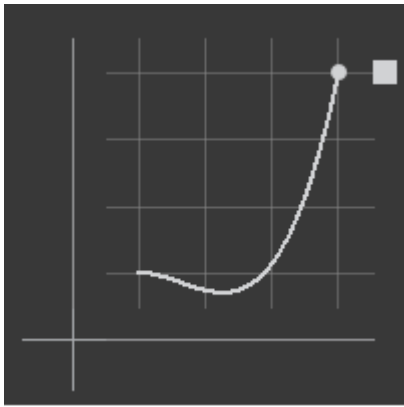
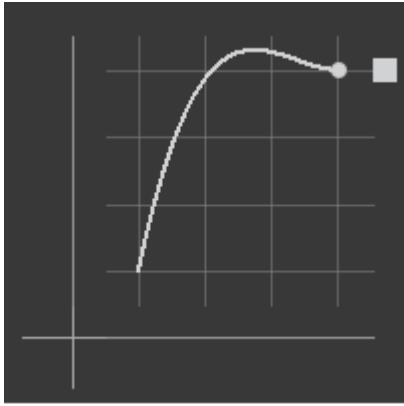
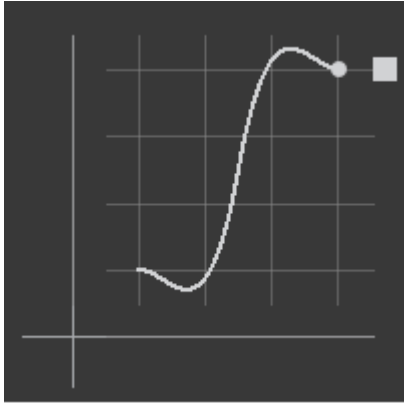
Constant	Illustration	Remark
OutSine		Easing curve for a sinusoidal ($\sin(t)$) function: decelerating from zero velocity.
InOutSine		Easing curve for a sinusoidal ($\sin(t)$) function: acceleration until halfway, and then deceleration.
OutInSine		Easing curve for a sinusoidal ($\sin(t)$) function: deceleration until halfway, and then acceleration.

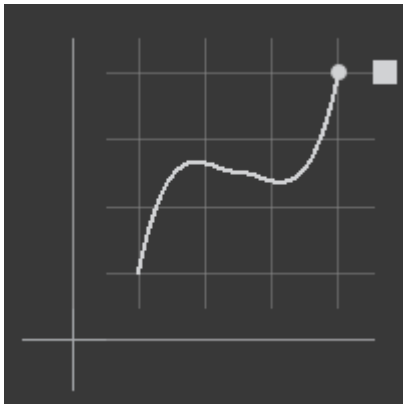
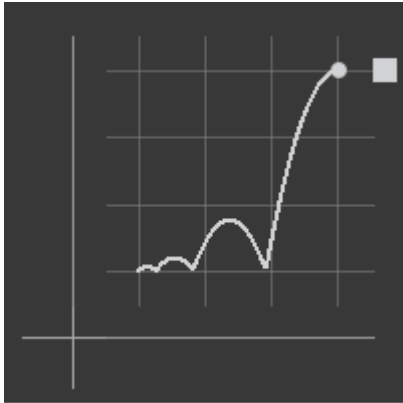
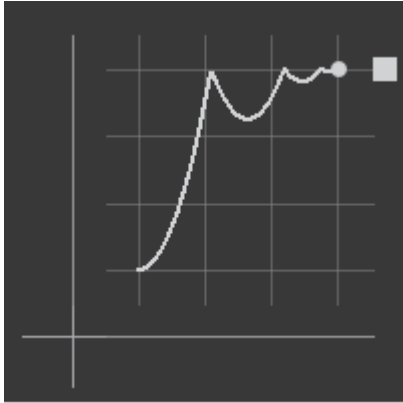
Constant	Illustration	Remark
InExpo		Easing curve for an exponential (2^t) function: accelerating from zero velocity.
OutExpo		Easing curve for an exponential (2^t) function: decelerating from zero velocity.
InOutExpo		Easing curve for an exponential (2^t) function: acceleration until halfway, and then deceleration.

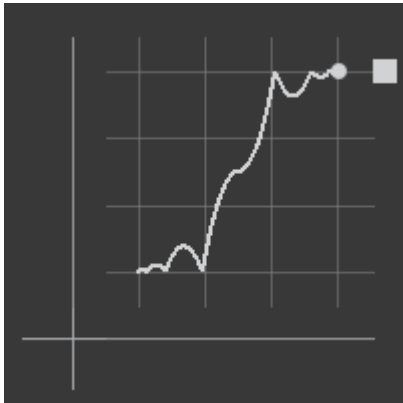
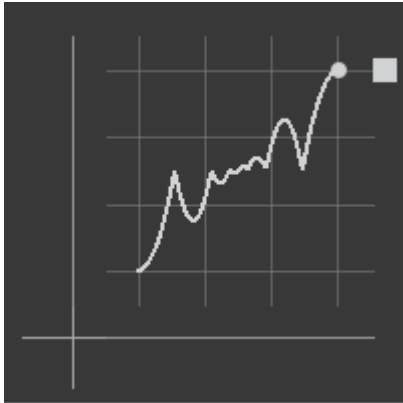
Constant	Illustration	Remark
OutInExpo		Easing curve for an exponential (2^t) function: deceleration until halfway, and then acceleration.
InCirc		Easing curve for a circular ($\sqrt{1-t^2}$) function: accelerating from zero velocity.
OutCirc		Easing curve for a circular ($\sqrt{1-t^2}$) function: decelerating from zero velocity.

Constant	Illustration	Remark
InOutCirc		Easing curve for a circular ($\sqrt{1-t^2}$) function: acceleration until halfway, and then deceleration.
OutInCirc		Easing curve for a circular ($\sqrt{1-t^2}$) function: deceleration until halfway, and then acceleration.
InElastic		Easing curve for an elastic (exponentially decaying sine wave) function: accelerating from zero velocity. The peak amplitude can be set with the amplitude parameter, and the period of decay by the period parameter.

Constant	Illustration	Remark
OutElastic	 A graph on a grid showing an easing curve that starts at the origin (0,0), rises steeply, and then oscillates with decreasing amplitude as it approaches the target value (1,1). The curve is smooth and has a single peak before settling.	Easing curve for an elastic (exponentially decaying sine wave) function: decelerating from zero velocity. The peak amplitude can be set with the amplitude parameter, and the period of decay by the period parameter.
InOutElastic	 A graph on a grid showing an easing curve that starts at the origin (0,0), rises to a peak, then falls to a trough, and finally rises to the target value (1,1). The curve is smooth and has two peaks before settling.	Easing curve for an elastic (exponentially decaying sine wave) function: acceleration until halfway, and then deceleration.
OutInElastic	 A graph on a grid showing an easing curve that starts at the origin (0,0), rises to a peak, then falls to a trough, and finally rises to the target value (1,1). The curve is smooth and has two peaks before settling.	Easing curve for an elastic (exponentially decaying sine wave) function: deceleration until halfway, and then acceleration.

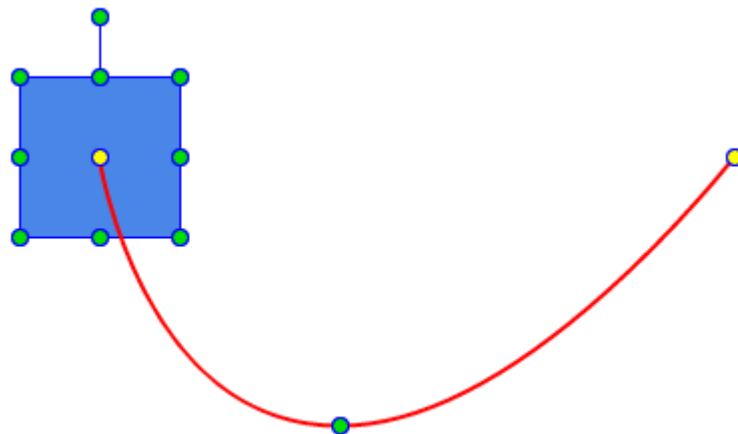
Constant	Illustration	Remark
InBack		Easing curve for a back (overshooting cubic function: $(s+1)^3 - s^3$) easing in: accelerating from zero velocity.
OutBack		Easing curve for a back (overshooting cubic function: $(s+1)^3 - s^3$) easing out: decelerating to zero velocity.
InOutBack		Easing curve for a back (overshooting cubic function: $(s+1)^3 - s^3$) easing in/out: acceleration until halfway, and then deceleration.

Constant	Illustration	Remark
OutInBack		Easing curve for a back (overshooting cubic easing: $(s+1)*t^3 - s*t^2$) easing out/in: deceleration until halfway, and then acceleration.
InBounce		Easing curve for a bounce (exponentially decaying parabolic bounce) function: accelerating from zero velocity.
OutBounce		Easing curve for a bounce (exponentially decaying parabolic bounce) function: decelerating from zero velocity.

Constant	Illustration	Remark
InOutBounce		Easing curve for a bounce (exponentially decaying parabolic bounce) function: decelerating from zero velocity.
OutInBounce		Easing curve for a bounce (exponentially decaying parabolic bounce) function easing out/in: deceleration until halfway, and then acceleration.

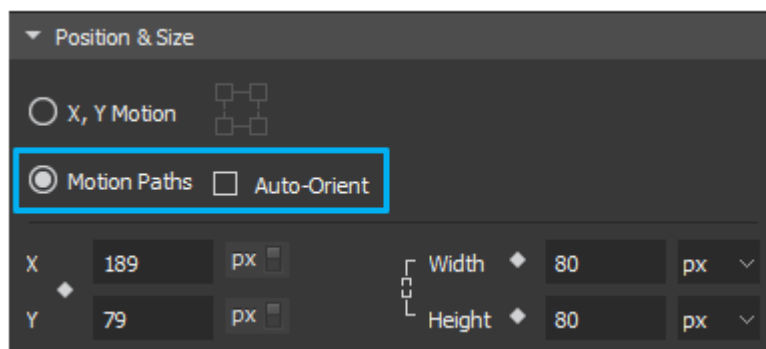
Motion Paths

A motion path is a curved animation between two or more points. Motion paths allow animating objects along complex curves to mimic real-life motion that cannot be achieved with linear motion.



Creating Motion Paths

To create a motion path, first create a basic path between two **keyframes**. Note that you can only switch to Motion Paths when the element's X, Y Motion is "relative to top and left". Otherwise, Motion Paths will be disabled.



1. Select an element.
2. In the **Properties** pane, switch from **X, Y Motion** to **Motion Paths**. This removes all animations of the Left/Top/Right/Bottom property of the element.
3. Add the starting Location keyframe at the point you want the animation to start.
4. Drag the Playhead to the point you want the animation to end.
5. Add the ending Location keyframe.
6. On the Canvas, drag the element to a new position. This creates a red line indicating the path of the motion. The Location property value changes, thus animating the element along a motion path.

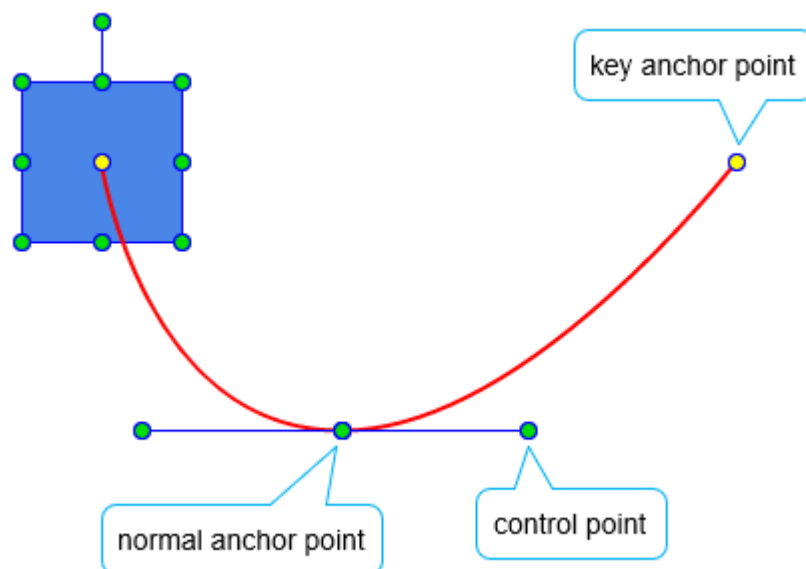
Repeat step 4, 5, and 6 many times to create a motion path between many points.

With these above steps, you've done creating a straight motion path between points. Edit the motion path by adding and modifying its anchor points.

Editing Motion Paths

Edit a straight path to make an element travel in a curve. Editing a motion path is actually working with its anchor points. Anchor points are circle points on a motion path that define the path. There are two types of anchor points:

- **Key anchor points:** Yellow points that have associated Location keyframes on the timeline.
- **Normal anchor points:** Green points with no associated keyframes on the timeline. They define the path an element moves between two key anchor points.



Each anchor point except the first/last key anchor one has two blue handles with a control point at their ends.

Creating Anchor Points

- **Key anchor points:** Add a Location keyframe to a motion path segment at a desired point of time. Location keyframes split a path into two. You can also delete a Location keyframe between two paths to create a contiguous path.
- **Normal anchor points:** Hover over a motion path until the mouse pointer turns into a pointing hand. Then, click and drag the path as desired.

Note: When the Playhead is on a Location keyframe, the element is on a corresponding key anchor point. Drag the element to change the shape of the path. If the Playhead isn't on a

Location keyframe, dragging the element is equal to dragging the entire path. The shape of the path remains the same.

Adjusting Control Points

With any anchor point selected, its control points appear to help adjust the path between two anchor points. Drag the control points to modify the path.

When dragging a single control point, the remaining control point will move to keep the angle between them unchanged. If you want to change the selected point only, hold down **ALT** and drag the point.

Note: If a control point and an anchor point have the same position, when you click the point, the anchor point will be selected. If you want to select the control point, hold down **ALT** and click the mouse button.

Removing Anchor Points

To remove an anchor point, click it while holding down **SHIFT**. If you are removing a key anchor point, one of the following cases happens:

- The key anchor point is the first/last point of the path AND there is no anchor point between this point and the next/previous key anchor point: The associated Location keyframe will be deleted.
- The key anchor point is the first/last point of the path AND there is an anchor point between this point and the next/previous key anchor point: The next/previous anchor point will become the first/last key anchor point of the motion path.
- The key anchor point is not the first/last point of the path: The associated Location keyframe will be deleted.

Note: If you click an anchor point and press **DELETE** without holding down **SHIFT**, you will delete the element together with the entire motion path.

Rotating Motion Paths

Hold down **ALT**, then click and drag the path. The path will be rotated around the element's transformation-origin point.

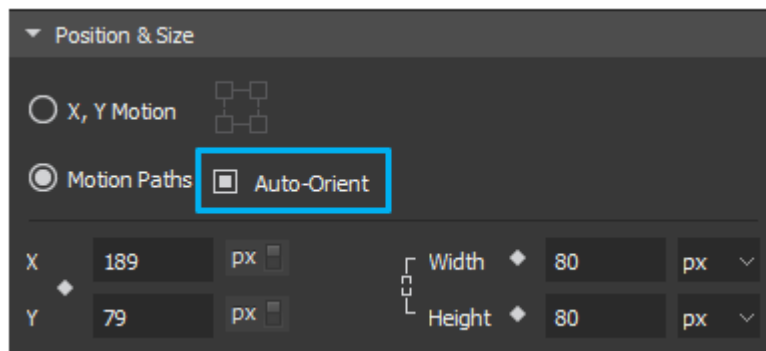
Moving Motion Paths

You can move the path without changing the relative movement of an element. Hold down **CTRL**, then click and drag the path to move the path.

Orienting Elements to Paths

Imagine you want to animate a car to drive up a slope. As the slope is not a straight line, you need to use a motion path to move the car. However, even with a motion path, the car's movement is still unnatural because the nose of the car doesn't orient along the direction which it is heading.

To solve this problem, use the Auto-Orient property to orient elements in relation to the direction of the motion path.



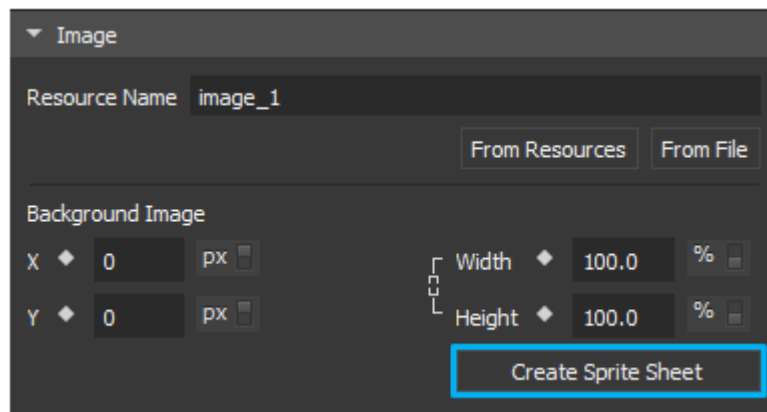
Note: If the right edge is not the “front” of an image, you may need to rotate the element so that the correct edge is on the right side. For example, if the car is moving to the left direction with Auto-Orient on, it will flip vertically. In that case, rotate the image by 180 degrees to avoid flipping.

Sprite Sheet Animations

A sprite sheet consists of multiple frames (also sprites) in one image. Using sprite sheets, you can create sprite sheet animation representing one or several animation sequences while only loading a single file.

Sprite sheet animation changes which frame/sprite is rendered in quick succession to give the illusion of movement.

1. Select an image element.
2. Go to the **Properties** pane > **Image** section > **Create Sprite Sheet**.



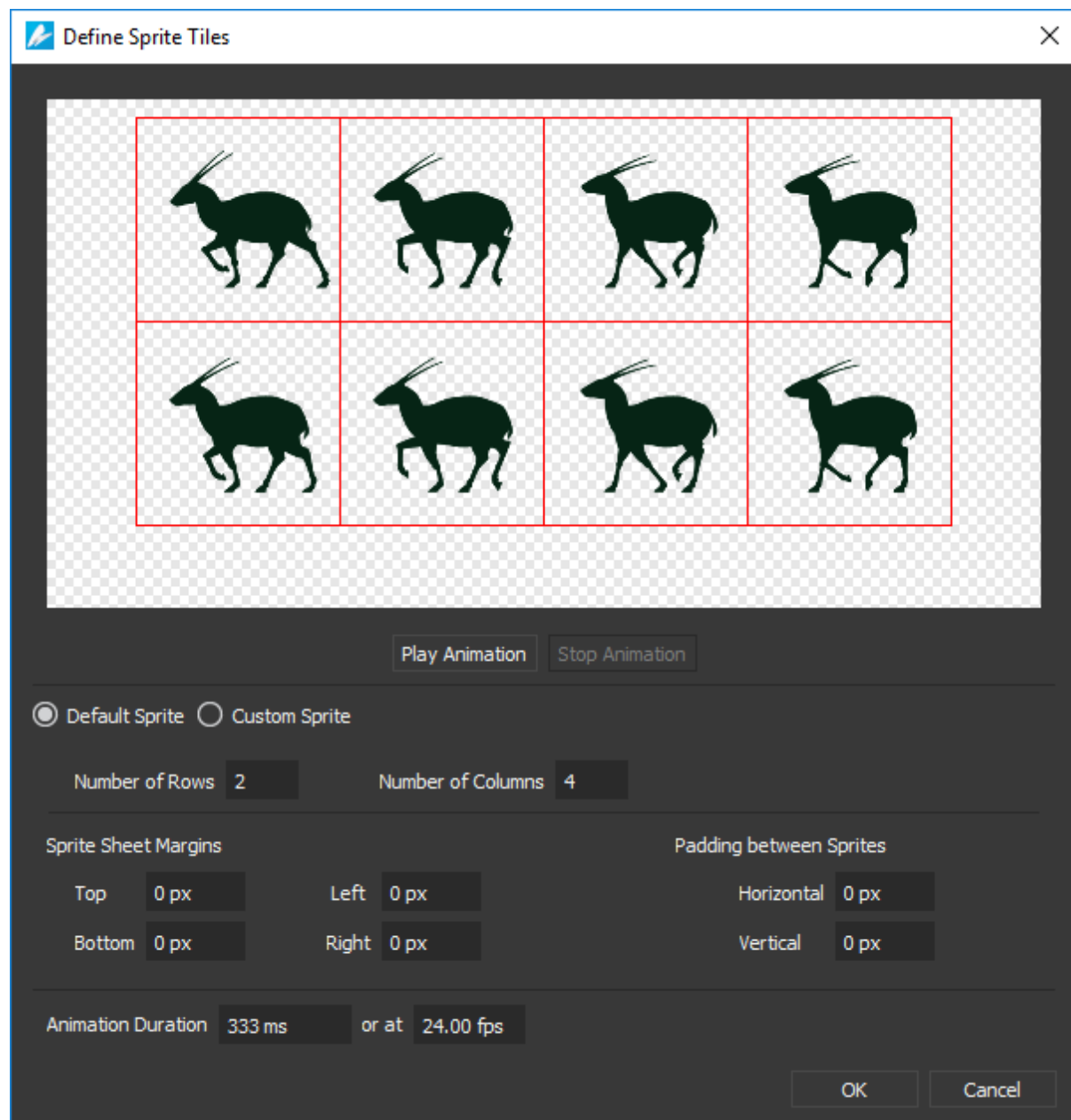
The **Define Sprite Tiles** dialog appears, providing two modes to define frames/sprites:

- **Default Sprite** generates frames automatically. Use this mode when your sprite sheet has uniform sized frames.
- **Custom Sprite** generates frames manually. Use this mode when your sprite sheet has non-uniform sized frames.

Default Sprite

To use the Default Sprite mode, ensure that:

- The frames in the sprite sheet are of equal dimensions.
- The padding between the frames is uniform.

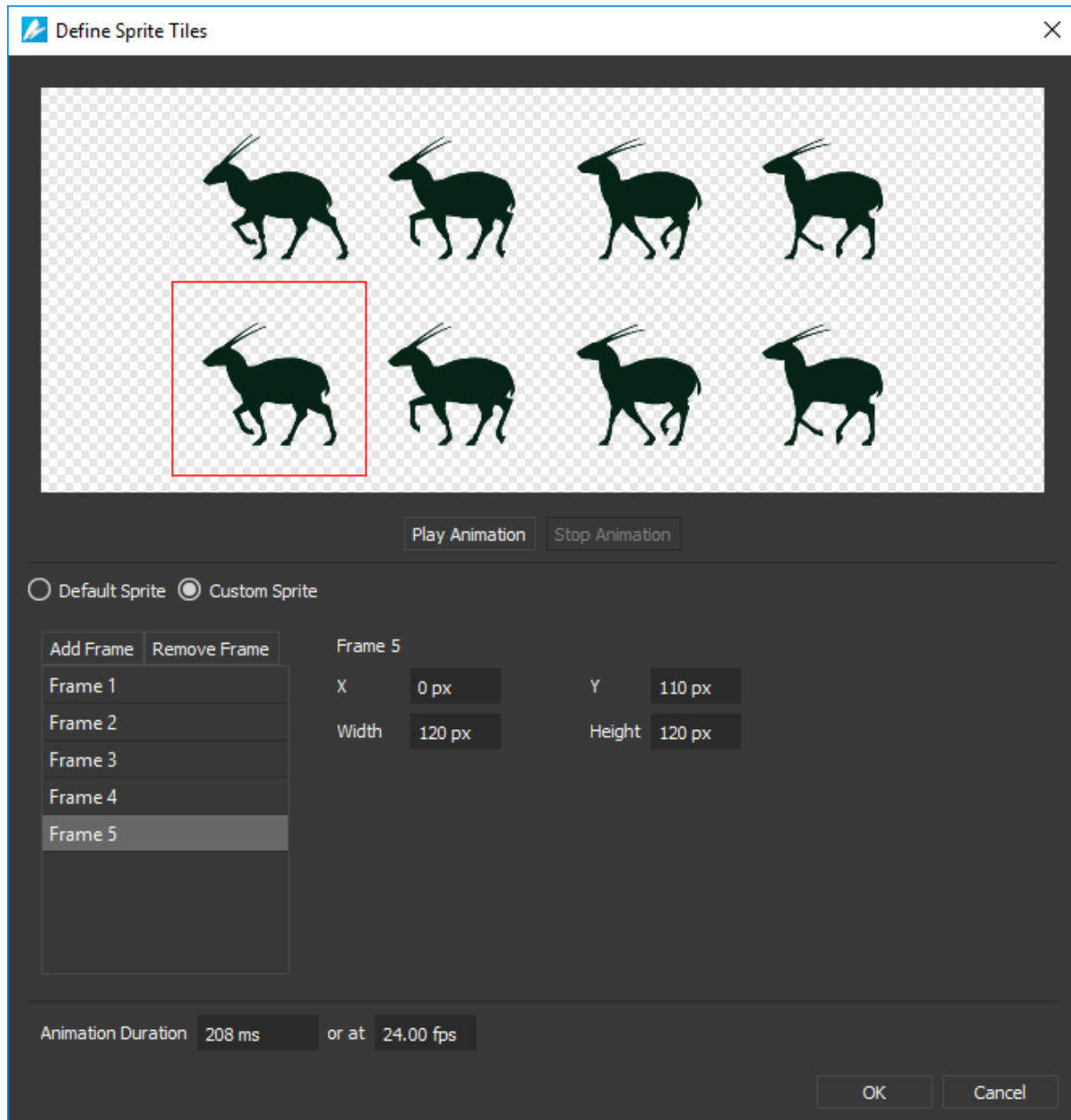


The dialog allows you to define both rows and columns for your sprite sheet animation.

1. Specify the **Number of Rows** and **Number of Columns**. Then, in the preview window, each frame is outlined by a red rectangle.
2. Specify the trimming options, including **Sprite Sheet Margins** and **Padding between Sprites**.
3. Specify the **Animation Duration** in milliseconds or at FPS.
4. Click **Play Animation** to preview animation in the preview window. Then, click **Stop Animation** to pause the preview.
5. Click **OK** to create frames.

Custom Sprite

If the images in the sprite sheet are of different sizes, use the Custom Sprite mode to manually define frames.



1. Click **Add Frame** to add a frame. Immediately, a red frame rectangle appears in the preview window.
2. To define the frame size, either drag and resize the frame rectangle in the preview window or specify the frame's position (**X**, **Y**) and size (**Width**, **Height**) in boxes.
3. Specify the **Animation Duration** in milliseconds or at FPS.

4. Click **Remove Frame** to remove a selected frame.
5. Click **Play Animation** to preview animation in the preview window. Then, click **Stop Animation** to pause the preview.
6. Click **OK** to create frames.

Chapter 6: Adding Interactivity

Interactivity is a great way to gain and keep audience's attention and interest as it gives them interactive control over their experience. Saola Animate provides great support for creating rich, interactive HTML5 content. You can use events, actions, and triggers to add interactivity to your project.

Events, Actions and Triggers

Events are occurrences that determine when to trigger an action. For example, the Mouse Up event occurs when users release a mouse button over an element. Events can be caused by user interactions such as a mouse click. Or, they can occur based on some conditions within the tool such as when the timeline pauses.

Saola Animate provides events for five item types:


1. **Element events:** Mouse and touch events on elements.
2. **Scene events:** Scene-specific events, keyboard events, and all element events.
3. **Document events:** Document-specific events and scene events.
4. **Timeline events:** Timeline-specific events related to timeline playback.
5. **Preloader events:** Preloader-specific events related to the loading progress.

Events trigger actions. That is, **actions** perform a function that responds when an event occurs. Saola Animate offers many different built-in actions:

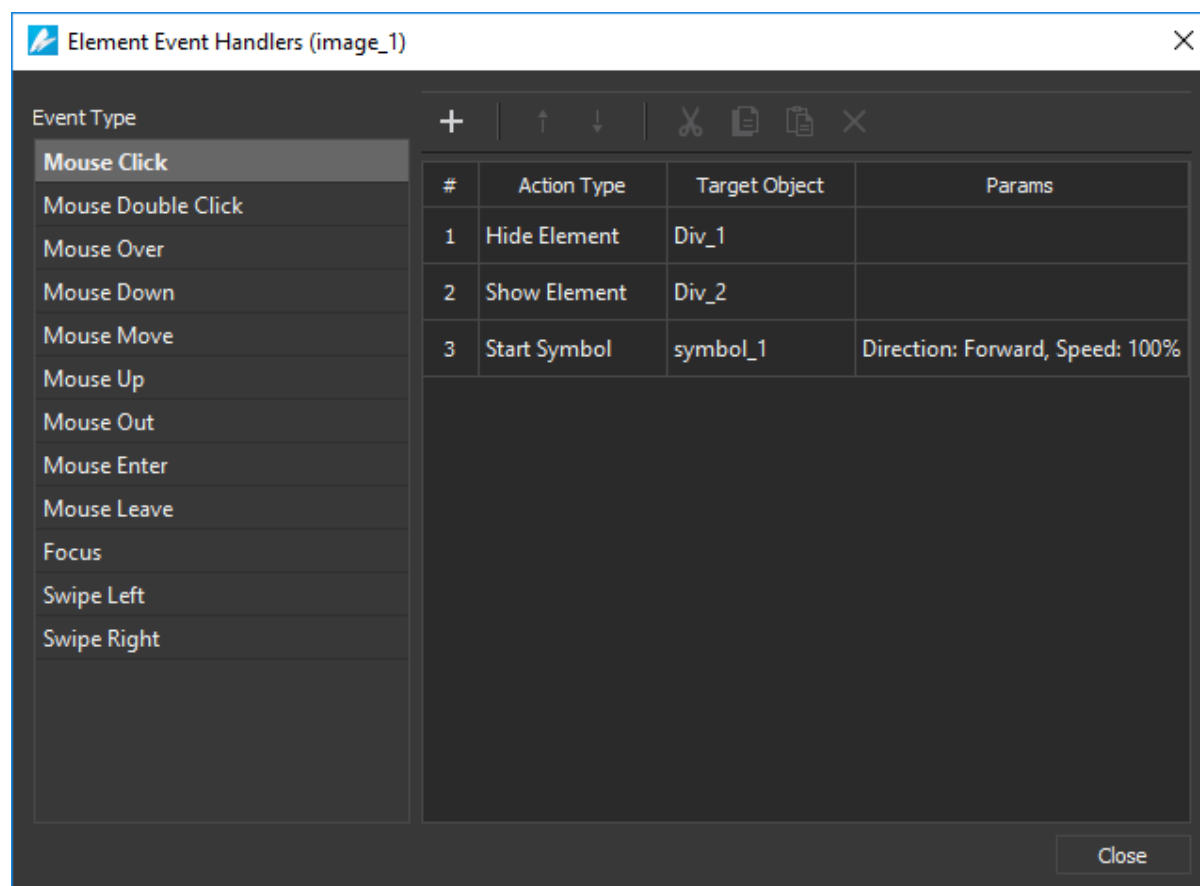
- **Timeline, Symbol, Media actions:** Control the playback of the timeline, symbol, audio, and video, such as playback state, speed, etc.
- **Visibility actions:** Control the visibility state of elements (show or hide elements).
- **Other actions:** Other actions such as loading a URL, invoking a JavaScript function, etc.

Triggers are actions placed on the timeline and executed without user control. You can insert many triggers on the timeline. Each trigger can hold different actions.

Adding Events - Actions


Add events and actions using the **Event Handlers** dialog. You can open this dialog by clicking the **Event Handlers**  buttons in the **Timeline**, **Properties**, and **Document** panes. You can also find the buttons in the context menu when you right-click an element on the Canvas or in the **Elements** pane.

In the **Event Handlers** dialog, the left side shows a list of available events that vary depending on the target, including element, scene, document, timeline, and preloader. The right side is for actions that have type, target object (if any), and parameters (if any) and are arranged in order in a table. You can double-click a table cell to edit the item. The dialog also has a toolbar on the top containing buttons for adding, ordering, copying, and removing actions.



Adding Actions

1. Choose an event to trigger the action.
2. Click **Add Action** to select the action and target object (if any).
3. Double-click the Params cell to edit the parameters. An action can have no or several parameters such as playback direction, URL, etc.

In the **Timeline** pane, an element with actions has a highlighted **Event Handlers**  button. So, it's easy to figure out which element holds interactivity.



Editing Actions

Double-click the Target Object and Params cells to change the target object and parameters (if any).

Replacing Actions

Double-click the Action Type cell to open the Action list. Then, select a new action to replace the existing one. Edit the new action's target object and parameters if needed.

Ordering Actions

The order of actions matters. Actions that are stated earlier are triggered before those stated later. You can reorder actions after adding them by clicking **Move Up**  and **Move Down**  on the toolbar.

Copying Actions

You can copy actions and paste them to the same or to another event. Just select the actions and use shortcuts or buttons on the toolbar to copy and paste them.

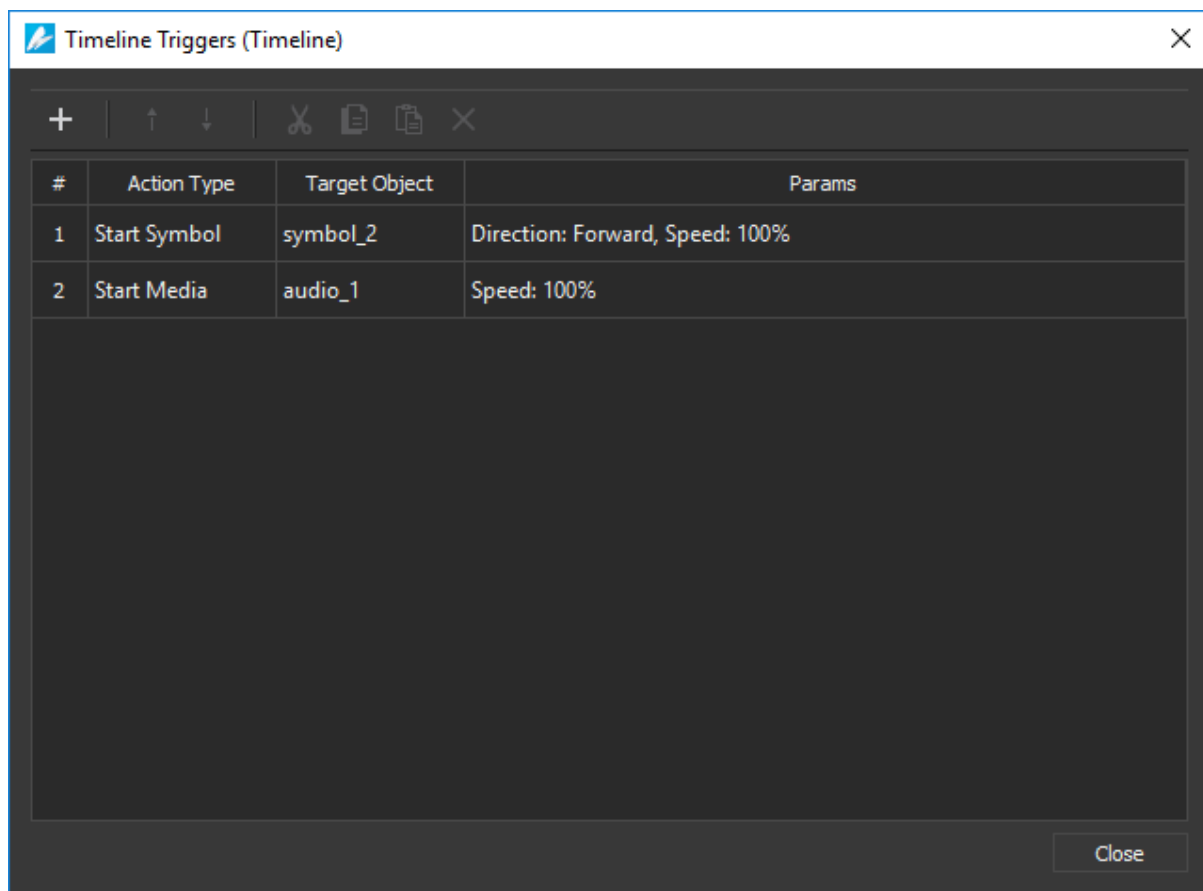
You can also copy all events and actions of an element and paste them to another element. On the Canvas or in the **Timeline** pane, right-click an element and select **Copy All Event Actions**. Then, select another element, right-click > **Paste Event Actions**.

Removing Actions


Select an action and press **DELETE** to remove the action. Alternatively, click **Remove Action** on the toolbar.

Inserting Timeline Triggers

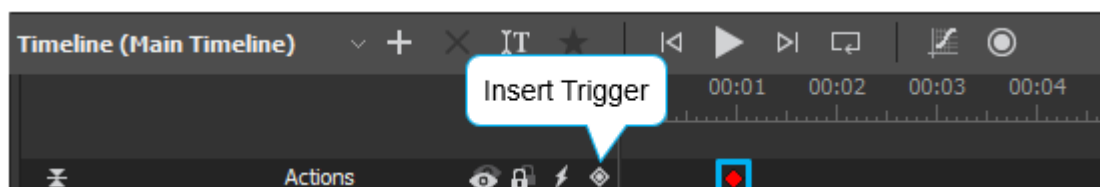
Timeline triggers fire at a certain point on the timeline, without user control. Manage timeline triggers in the **Timeline Triggers** dialog which looks similar to the **Event Handlers** one except that it doesn't have the Event Type column.



Inserting Triggers

1. Position the Playhead at the point you want to add a trigger.
2. Press **ALT+T** or click the **Insert Trigger**  button to open the **Timeline Triggers** dialog.
3. Click **Add Action** on the toolbar to add an action. Do the same steps as when you add actions in the **Event Handlers** dialog.

Click **Close** to apply changes and close the dialog. In the **Timeline** pane, the trigger is inserted at the Playhead position. It appears as a red diamond on the Actions bar.



Editing Triggers

Double-click a trigger to open the **Timeline Triggers** dialog. Then, edit the actions as desired.

Moving Triggers

In the **Timeline** pane, click and drag a trigger to a new position.

Copying Triggers

Click a trigger on the Actions bar. When selected, it turns yellow. Then, use shortcuts to copy (**CTRL+C**) and paste (**CTRL+V**) the trigger to the new Playhead position. You can also use commands in the trigger context menu to get the same result.


Removing Triggers

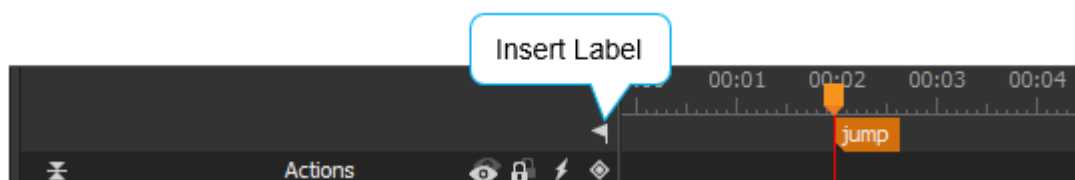
Select a trigger and press **DELETE** to remove it. Or, right-click and select **Delete** from the context menu. If you delete all actions bound to a trigger, the trigger is also deleted.

Inserting Timeline Labels

Timeline labels mark important points on the timeline. They serve as bookmarks to which you can refer later. Labels appear below the timeline ruler and right above the Actions bar, giving you the benefit of both a word and number. Give a label a descriptive name and they prove highly useful for triggers and actions. For example, you want to add a trigger that seeks a time. Instead of jumping to somewhere like 0:00:12.357, add a “jump” label and seek it.

Inserting Labels

1. Drag the Playhead to a special point on the timeline that you want to mark.
2. Press **ALT+L** or click the **Insert Label**  button (the flag icon). A label appears at the Playhead position with its default name such as Label_1, Label_2, etc.



Rename a label right at that time or afterward. If you created a label and later want to rename it, double-click it and enter a new name. Note that if you change the label's name, you have to go back to any triggers or actions using that label and update its name.

Moving Labels

Click and drag a label to a new spot. This comes in handy when you use labels to create actions. For example, you use a “jump” label in a trigger that seeks a time. When you're fine-tuning the animation, odds are you're constantly adjusting the timing to find a good time to “jump”. Without a label, you'd have to manually update the time value repeatedly. But using a label, you only need to move it to a new spot and preview your changes.

Copying Labels

Select a label and press **CTRL+C** to copy, then press **CTRL+V** to insert the copied labels into the current Playhead position. Alternatively, use the options in the label context menu.

Removing Labels

Select a label and press **DELETE** to remove it. Or, right-click and select **Delete** from the context menu.


Chapter 7: Using Symbols

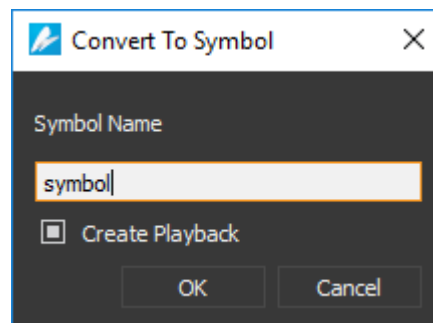
Symbols are reusable resources with independent timelines and interactive capabilities. Symbols enable you to easily reuse elements, scenes, timelines, and animations. For example, you can create a piece of animation, convert it to a symbol, and then use it many times in many larger animations.

Regarding symbols, you need to distinguish between symbols and symbol elements. Symbols are resources in the **Resources** pane of a document. Create symbols by converting elements on the Canvas. Meanwhile, symbol elements are the elements in which symbols are embedded. They are the copies of symbols located on the Canvas. Many symbol elements can use the same symbol.

Converting Elements to Symbols

Create animation symbols by converting one or multiple elements on the Canvas.

1. Select one or multiple elements of the same parent.
2. Click **Convert to Symbol**  on the toolbar or right-click > **Convert to Symbol**.
3. In the dialog, type a unique name for the new symbol.



4. The **Create Playback** check box is selected by default to create playback animation for the new symbol. If you don't want to create symbol playback at this time, clear the check box. Later, use **actions** to control symbol playback.
5. Click **OK** to convert selected elements into a symbol.

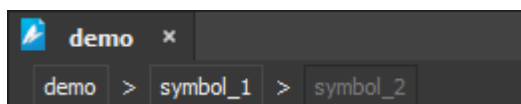
Saola Animate adds the new symbol to the **Resources** pane. The selection on the Canvas becomes a symbol element of that symbol. By default, the size of the symbol element is set to relative to the scene size and start at 100% of the scene size. If you want the size of the symbol element to be absolute, change the size unit to pixels (px).

Saola Animate also lets you create responsive symbols which have layouts and can respond to the container size changes. See **Responsive Symbols** for more details.

Editing Symbols

Double-click a symbol element on the Canvas to enter the symbol editing mode where you can edit the embedded symbol. Alternatively, right-click the symbol element on the Canvas and select **Edit Symbol**, or double-click a symbol in the **Resources** pane.

The editor then displays the content of the symbol which resembles a normal document with scenes and timelines. You can work with elements, animations, and interactivity as you do with a normal document. At the top of the Canvas, there is a stack of buttons displaying the symbol hierarchy. These buttons not only show which symbols you are editing but also help navigate the hierarchy. In the figure below, demo is the main document. The symbol_1 contains the symbol_2, in other words, the symbol_2 is nested within the symbol_1. You are working on the symbol_2.



To navigate the symbol hierarchy, do one of the following:

- Click the symbol/document name in the hierarchy to navigate to a desired symbol or the main document. For example, click the demo button to close all symbols and return to the main document.
- Double-click outside the content of the symbol on the Canvas to navigate to the closest parent symbol. For example, double-click outside the content of the symbol_2 to jump up to the symbol_1. Similarly, double-click outside the content of the symbol_1 to jump up to the main document.


Editing a symbol updates all related symbol elements in a document, which helps maintain consistency and save a lot of time. This is the reason why symbols are useful for sharing identical elements across scenes or at different positions in the same scene. Meanwhile, you can edit each symbol element of the same symbol differently. Editing a symbol element updates that element only.

Exporting and Importing Symbols

You can export symbols for reusing and sharing between Saola Animate documents.


Exporting Symbols

The **Resources** pane manages all resources of a document, including symbols. From here you can export a symbol to file (*.saolasym).

1. Select the symbol you want to export.
2. Click **Export** on the toolbar, or right-click > **Export** .
3. Provide the file name for the exported symbol.

Importing Symbols

When importing a symbol into a document, you also import all resources used in this symbol into the target document. These resources become resources of the document, not those of the symbol. Do one of the following to import a symbol:

- Drag and drop a symbol onto the Canvas.
- Copy and paste a symbol from your computer.
- Click **Symbol**  on the toolbar and select a symbol from your computer.
- Import a symbol into the **Resources** pane and later add the symbol to a scene.

Chapter 8: Using Fonts

This chapter provides information about selecting fonts and using Google fonts.

Choosing Fonts

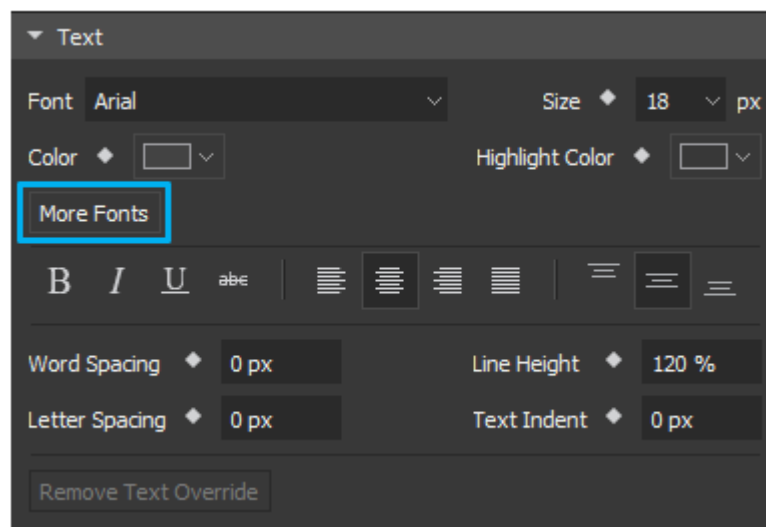
ActivePresenter provides three sets of fonts, including:

- **Web safe fonts:** Fonts are most likely to be available in a wide range of browsers and different computer systems. These fonts will show consistently across almost all browsers.
- **System fonts:** Fonts are already installed on your computer. You can use these fonts in your document and choose to export them with the project (**Document** pane > **Embed System Fonts**). That way, your text maintains consistent appearance despite the computer that plays the output.
- **Google fonts:** Free open source web fonts from Google Fonts. Using Google fonts brings great advantages:
 - **Open Source license:** They are freely accessible to everyone to use, redistribute, and customize.
 - **Cross-site caching:** Downloaded fonts are cached in the browser and reused by any other webpages using them, making the web faster.
 - **Cross-platform rendering:** They are supported in all major browsers and a wide range of mobile devices.

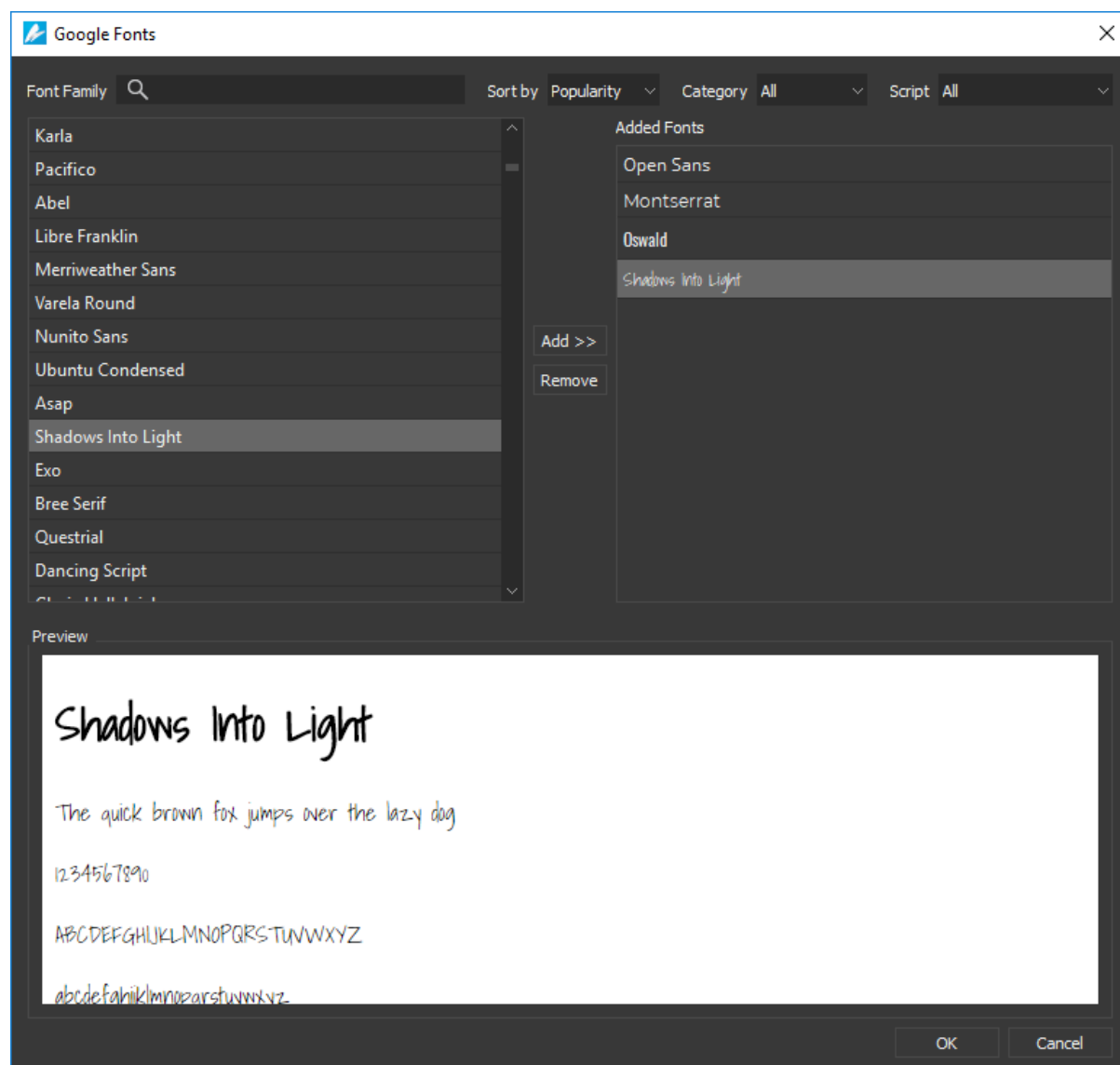
Adding Google Fonts

Add fonts from Google Fonts using the following steps:

1. In the **Properties** pane > **Text** section, click **More Fonts**.



Or, click **Add fonts...** from the **Font** combo box in the **Floating Text Format** toolbar and the **Text** section. This opens the **Google Fonts** dialog which lists all the available Google fonts.



You can search for specific fonts or filter fonts by font properties.

- **Search:** Search the font by family names.
 - **Sort by:** Sort the fonts based on popularity, alphabetical, date added, or trending.
 - **Category:** Filter fonts by typefaces including Serif, San serif, Display, Handwriting, and Monospace.
 - **Script:** Filter fonts by writing systems such as Arabic, Bengali, Greek, Khmer, etc.
 - **Preview:** Display a preview of the text with the selected font.
2. To add fonts to the document, select the fonts and click the **Add** button. The selected fonts now appear in the **Added Fonts** pane. You can select any added font to apply to the currently selected text or element, if any.

3. Click **OK** to apply changes and close the dialog.

Google fonts added to your document will appear in the **Font** combo box. They also become font resources in the **Resources** pane.

Note: Adding Google fonts requires a network connection.

Removing Google Fonts

To remove a Google font, do the following:

1. Open the **Google Fonts** dialog.
2. Select the fonts you don't want any more in the **Added Fonts** pane.
3. Click the **Remove** button to remove them from the selection list.
4. Click **OK** to apply changes and close the dialog. The fonts no longer appears in the **Font** combo box nor the **Resources** pane.

Besides, you can go to the **Resources** pane, select the Google font you don't want and click

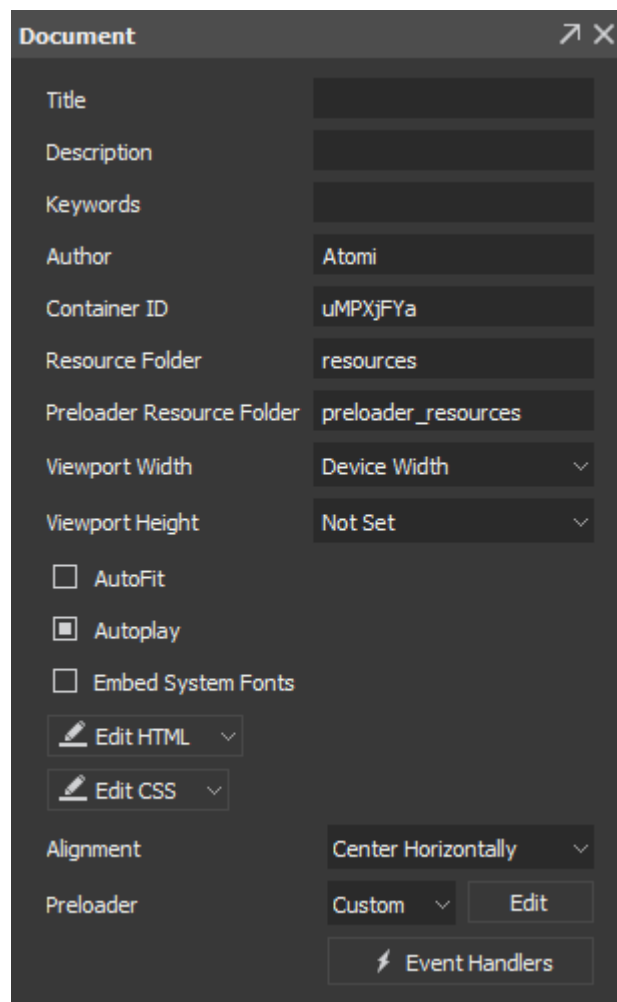
Delete  on the toolbar.

Chapter 9: Previewing and Exporting Projects

After editing a project, it's time to preview and export it to HTML5. This chapter covers the techniques for editing document properties, previewing, and exporting your project.

Updating Document Properties

Edit document properties in the **Document** pane to set up global parameters of a project and export options.



- **Title:** Export the content to the <title> tag to define the title of the HTML document, shown in a browser title bar or on the page tab. The <title> tag also provides a title for

the page when it is added to favorites and displays a title for the page in search-engine results.

- **Description, Keywords, Author:** Define the value of the <meta> tag to specify the description, keywords, and author of the HTML document. Metadata will not be displayed on the page but will be machine-parsable.
- **Container ID:** Define a unique identifier of the HTML tag in which the HTML5 output is embedded. The ID can contain only alphanumeric characters (A-Z, a-z, 0-9), underscore (_), and hyphen (-), and must start with an alphabetic character for CSS selector to work without escaping.
- **Resource Folder:** Define the folder path which contains the resources of the HTML5 output. This path is relative to the path of the HTML5 output. Make sure the path doesn't contain any of the following character: \ : * ? " < > | and uses / as a directory separator.
- **Preloader Resource Folder:** Define the folder path which contains the resources of the custom preloader of the HTML5 output. This option is only available when you use **custom preloader**.
- **Viewport Width, Height:** Define the value of the viewport <meta> tag to control layout on mobile browsers. Select a value from the combo box or input an integer value for the viewport width/height in pixels. Device Width/Height sets the width/height of the exported document viewport to follow the screen width/height of the device. Choosing Not Set excludes any viewport <meta> tag from the document content.
- **AutoFit:** Automatically scale the document without changing the proportion to fit it into the container. This option is also applied to symbol documents.
- **Autoplay:** Automatically play the document when a browser opens, otherwise the document must be called to start using **actions**.
- **Embed System Fonts:** Embed any used system fonts in the HTML5 output using CSS3 Web Fonts Specification. Text will maintain appearance no matter the font installation on end users' computer.
- **Edit HTML:** Open the **HTML Output** editor where you can edit the HTML of the document. Be careful when editing tags wrapped within {\$} as these tags will be replaced by program-generated contents. After editing, if you want to go back to the original content, click **Restore Defaults** (the down arrow).
- **Edit CSS:** Open the **CSS Output** editor where you can edit the CSS of the output. After editing, if you want to go back to the original content, click **Restore Defaults** (the down arrow).
- **Alignment:** Decide how the document should be displayed in the browser window. By default, the document is centered horizontally within the browser. This option is also applied to symbol documents.
- **Preloader:** Use either a default or a custom **preloader** as a visual indicator that is displayed when the document is loading.


- **Event Handlers:** Add interactivity to control the behavior of the document.

Previewing Scenes and Projects in Browser


Previewing during and after the editing process allows you to see how your project will look and behave to the end user. You can also review your work and notice the changes that are required. That way, you can be sure of the final output. Two preview modes are available: **preview scene** and **preview project**.

Previewing scene lets you view a single scene's animation and interactivity. In this mode, you cannot perform any actions navigating to other scenes. Meanwhile, previewing project lets you preview the whole project so that you can review all scenes' animation and interactivity, from the first to the last scene.

To preview the current scene in a browser, do one of the following:

- Press **CTRL+ALT+S**.
- Click **Preview Current Scene in Browser**  on the toolbar.
- Click **File > Preview Current Scene in Browser** from the menu bar.

To preview the whole project, do one of the following:

- Press **CTRL+ALT+P**.
- Click **Preview Project in Browser**  on the toolbar.
- Click **File > Preview Project in Browser** from the menu bar.

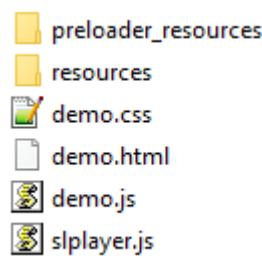
You can also preview your work using the Canvas. Click **Play/Stop** in the **Timeline** pane to play the current scene. Also, drag the Playhead along the timeline to get the same result. Remember that this only plays scene's animation, no interactivity involved.

Exporting Projects to HTML5

Saola Animate exports documents to HTML5. To export, do one of the following:

- Press **CTRL+ALT+E**.
- Click **Export to HTML5**  on the toolbar.
- Click **File > Export to HTML5** from the menu bar.

Once the project is exported to HTML5, it will be stored in a folder with structure as below. The files are named based on the project title or project name.



- **preloader_resources**: Optional folder which contains all resources inserted into the custom preloader. If the custom preloader doesn't use any external resources such as images, audio, or videos, this folder will not appear.
- **resources**: Folder which contains all resources inserted into the project.
- **demo.css**: Document-specific data which defines the CSS style of the document.
- **demo.html**: HTML document. Open this file to play the project in a browser.
- **demo.js**: Document-specific data which defines all scenes, timelines, elements, and animations of the document.
- **slplayer.js**: The main Saola Animate HTML5 player.

Browser Compatibility for HTML5

Saola Animate is compatible with:

- Chrome 4+
- Firefox 3.5+
- IE 9+
- Edge
- Safari 3.1+
- Opera 11.5+

Newer versions are required for some features including:

- **Gradient Fill**: Chrome 10+, Firefox 3.6+, IE 10+, Safari 5.1+, Opera 12.1+.
- **Audio/Video**: Firefox 11+, Safari 4+. Users still have to use compatible formats that a browser supports.

Saola Animate HTML5 output checks if a browser supports a specific feature or not. If the browser doesn't support the feature, Saola Animate will fall back on less advanced options to keep the output compatible. For example, if a browser doesn't support Gradient Fill, it will automatically use Solid Fill. If Saola Animate cannot find a fallback, the feature will not work in that browser.

Chapter 10: Using Canvas

The Canvas is the main work area in the center of the application window. It has horizontal and vertical rulers and a scene editor. It enables you to view scenes, add elements to scenes, modify, and arrange elements. The Canvas is also where you can **preview your work** during the editing process.

Use rulers and Canvas Snapping to position elements on the Canvas. Zoom in and out to change the view of the Canvas when you work. While editing, if you encounter any problems with the Canvas, such as lagging or elements not showing, right-click the Canvas and select **Reload Canvas** to refresh the Canvas.

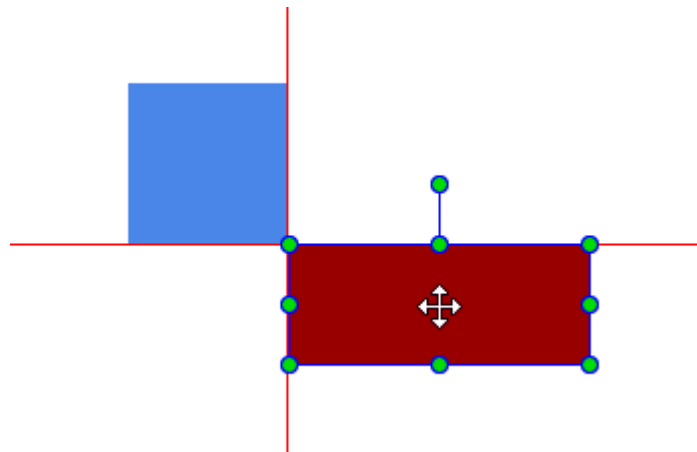
Rulers

The horizontal and vertical rulers appear along the top and left edge of the Canvas help position elements precisely. To show or hide the rulers, click **View > Rulers** from the menu bar.



Snapping on Canvas

When you move or resize elements on the Canvas, elements can be snapped to the center or edge of the Canvas or other elements thanks to Canvas Snapping. Canvas Snapping is indicated by red lines which appear when an element comes in the vicinity of the Canvas or another element.



Canvas Snapping is enabled by default. To disable this feature, click **View > Canvas Snapping**. You can temporarily disable this feature by holding down **CTRL** when moving or resizing elements.


Zooming Canvas In and Out

Zoom the Canvas to view the entire scene or a particular area of it. The minimum value for zooming out is 10% while the maximum value for zooming in is 400%.

There are several ways you can use to zoom the Canvas:

- Hold down **CTRL** and scroll the mouse wheel up to zoom in and scroll the mouse wheel down to zoom out. Alternatively, press **CTRL+=** to zoom in and **CTRL+-** to zoom out.
- Use the Zoom controls in the bottom right corner of the application window. Drag the slider to the right, toward the **+** button to zoom in, or click the **+** button multiple times. Meanwhile, drag the slider to the left or click the **-** button to zoom out. Alternatively, click the Zoom percentage level to open the Zoom drop-down list and choose the zoom level you want.



- To scale the Canvas so that it fits into the viewport size, press **CTRL+0** or click the **Fit on Screen**  button. The Canvas will zoom in or out automatically to ensure that all scene contents appear in the viewport window.

If you zoom in the Canvas too much, it will show horizontal and vertical scroll bars. Just scroll the mouse wheel to scroll the Canvas vertically. Hold down **SHIFT** and scroll the mouse wheel to scroll the Canvas horizontally.

Chapter 11: Using Timeline Pane

Timelines contain animations, as represented in a time-based view. Scenes can have more than one timeline to create complex animations and interactivity. Each scene has a main timeline which automatically starts when the scene loads while other additional timelines are called to play by actions. Timelines can be made to play, pause, seek, or change playback speed and direction to create rich and interactive animations.

For more details about commands and shortcuts in the **Timeline** pane, see **Timeline Menu**.

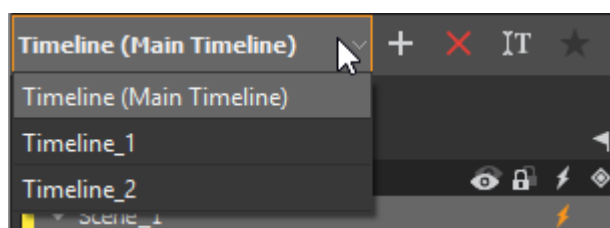
Managing Multiple Timelines

A scene may have multiple timelines, including the main timeline and additional ones. The main timeline starts off automatically when a scene loads. Meanwhile, additional timelines, which also contain animations, don't. Instead, you must use **actions and triggers** to control their playback, such as start, pause, and continue them.

Multiple timelines come into play when you create rich and interactive HTML5 animation. This feature lets you make different animations on different timelines using the same set of elements.

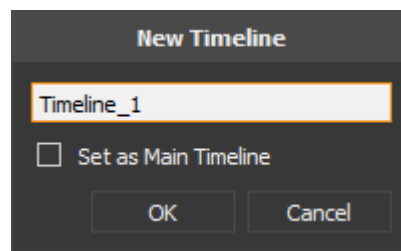
Creating Timelines

The top left of the **Timeline** pane places a set of buttons for managing multiple timelines.



To add a new timeline to a scene, do the following:


1. Click **New Timeline**  to open the **New Timeline** dialog.



2. Enter a unique name for the new timeline.
3. Select **Set as Main Timeline** if you want the new timeline to act as the main timeline, making it start once the scene loads.
4. Click **OK** to finish.

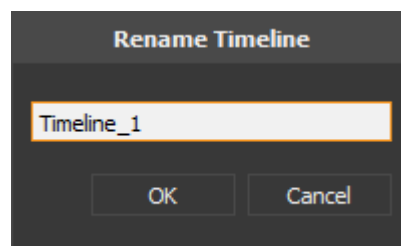
Newly created timelines are directly selected for editing. You can click the **Timeline Selector** to switch between multiple timelines of a scene.

Removing Timelines


Click **Remove Timeline**  to delete the active timeline. This button becomes available only when a scene has more than one timeline.

Renaming Timelines

Select a timeline from the **Timeline Selector**. Then, click **Rename Timeline**  to rename it.



Setting Timelines as Main Timeline

Select a timeline that isn't the main timeline from the **Timeline Selector**. Click **Set as Main Timeline**  to make that timeline the main timeline of a scene.

Navigating in Timeline

Saola Animate offers many controls to navigate the timeline by moving the Playhead. Do one or more of the following:

- Drag the Playhead to the desired location.
- Click the timeline ruler at the time where you want to move the Playhead. That way, the Playhead will immediately jump to that location.
- Change the time value in the time control (also called the current time indicator). Click on the time control to make it editable, then scroll the mouse wheel or enter a new value.

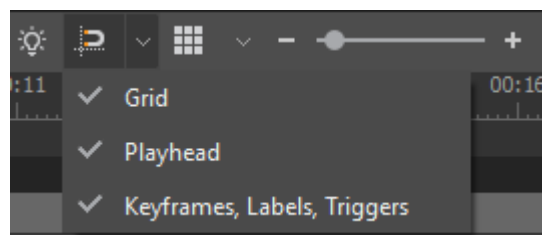
Use shortcuts or the Playback buttons to preview your animation. Press **SPACE BAR** or click **Play/Stop** or to start or stop animation in real time. Press **Home** or click **Go to Start** to jump to the start of the timeline. And press **END** or click **Go to End** to jump to the end of the animation. Also, click **Loop Playback** to play the animation in a loop.

Snapping in Timeline

Dragging the Playhead along the timeline, you may notice some snapping actions. For example, the Playhead jumps to a keyframe, a trigger, or a particular tick mark on the timeline ruler.

Timeline Snapping lets you move the Playhead and timeline items such as keyframes, labels, and triggers as you wish. It helps you easily line up keyframes and animation segments on the timeline.

Timeline Snapping is turned on by default with three modes:




- **Snap to Grid:** The Playhead and timeline items snap to the grid lines. Grid lines give precise measurements of the timeline by displaying subtle vertical lines at regular intervals. They only appear when you click the **Show Grid** button.
- **Snap to Playhead:** Timeline items snap to the Playhead.
- **Snap to Keyframes, Labels, Triggers:** The Playhead and timeline items snap to the position of timeline items.

Note: When you enable Timeline Snapping without any modes, the Playhead and timeline items will only snap to special tick marks on the timeline ruler. These tick marks appear depending on the zoom level of the timeline.

Zooming Timeline In and Out

As your animation can be fairly long, you may not be able to see all animation segments on the timeline. In that case, use the Timeline zoom controls to zoom the timeline in and out to get better view of the animation.

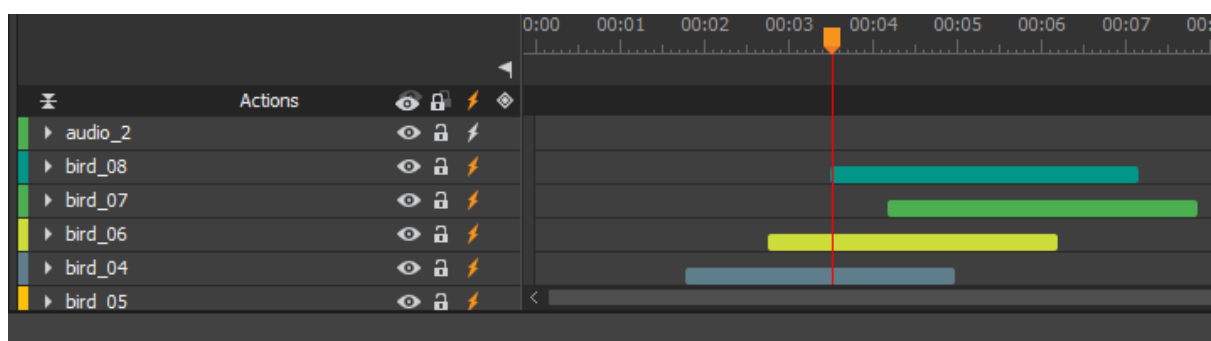


- **Zoom slider:** Drag the slider to the right, toward the + button to zoom in. Or, click the + button multiple times to zoom in to the level you want. Conversely, drag the slider to the left or click the – button to zoom out.
- **Zoom to Fit** : Click to fit the existing animation to the timeline view.


Additionally, you can use the mouse wheel to zoom the timeline. Place the cursor on the timeline, then hold down **CTRL** while scrolling the mouse wheel to zoom in or out.

Managing Timeline View


The Timeline lists a scene together with its elements and all animated properties. To the left of each element name, there's a small color bar which labels each element and its animation segments with a specific color. That way, Saola Animate helps you quickly identify which animations belong to an element.



Showing and Hiding Element Properties

If you're creating complex animations, the list of elements and properties can go on forever. To hide all animated properties of an element, click the triangle button next to the element name. If you want to expand or collapse all elements at once, click the **Expand/Collapse Lanes**  button on the Actions bar.

Filtering Elements

A scene may have a background image that stays unchanged over time and you don't want to clutter up the timeline with those static elements. To filter out elements with no animations, click the **Show Animated Elements Only**  button. This filter doesn't affect what you see on the Canvas, it only affects what you see on the timeline.



Showing, Hiding and Locking Elements


It's often hard to work with a lot of elements and complex animations. You may not select an element on the Canvas because it's behind others. To find a way out, you need look for that element on the timeline by scrolling the mouse wheel. You may also likely select the wrong element by accident because elements are placed so closely.

To solve these problems, Saola Animate provides two different controls: Visibility and Locking. These controls are available in the **Timeline** and **Elements** panes.

Showing and Hiding Elements

The Visibility decides whether to display elements in a scene while editing. This control is best used to help you hide some elements on the Canvas so that you can focus on the target elements. Bear in mind that this is only an interface feature helping you work with many elements. It does not affect the output.

In the **Timeline** or the **Elements** pane, notice the **Show/Hide** button next to the names of elements. An open eye  means that an element is visible, whereas an eye with a slash  indicates that it is invisible.

To show/hide all elements in a scene at once, click the **Show/Hide All**  button on the Actions bar.



Note:

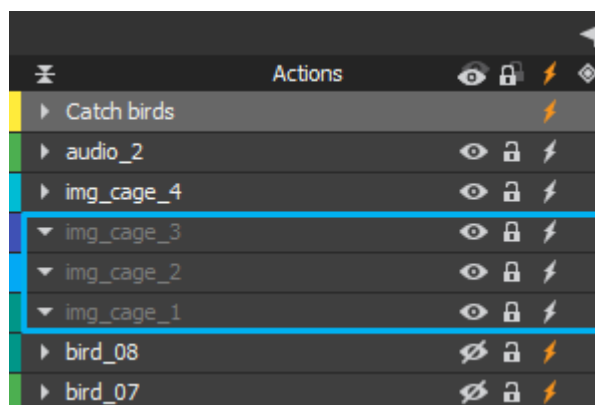
- When you hide an element, its child elements are also invisible.


- When an element is hidden on the Canvas, it can still be selected and edited from other panes.
- To control the visibility in the output, animate the Opacity or Display properties. This is useful when you want an element to fade in or out a scene. Or, you want an element to appear later in the animation and then make it disappear. Animate the Opacity from 0% to 100% to fade an element in, and from 100% to 0% to fade it out. Meanwhile, for the Display property, choose Off to make an element invisible and On to make it suddenly appear in a scene.

Locking and Unlocking Elements

Moving an element by mistake is a bad experience. To avoid that, you can lock elements in their location. That way, elements cannot be selected, moved, or edited, but they still remain visible with the Visibility state on.

To lock an element, click the **Lock/Unlock** button next to the **Show/Hide** button. By default, elements are in the “unlock” state when the button looks like an open lock . At that time, you’re free to select them and make an edit. When you lock elements in place, the button turns into a closed lock  and the element’s name gets dim.



To lock/unlock all elements in a scene at once, click the **Lock/Unlock All**  button on the Actions bar.

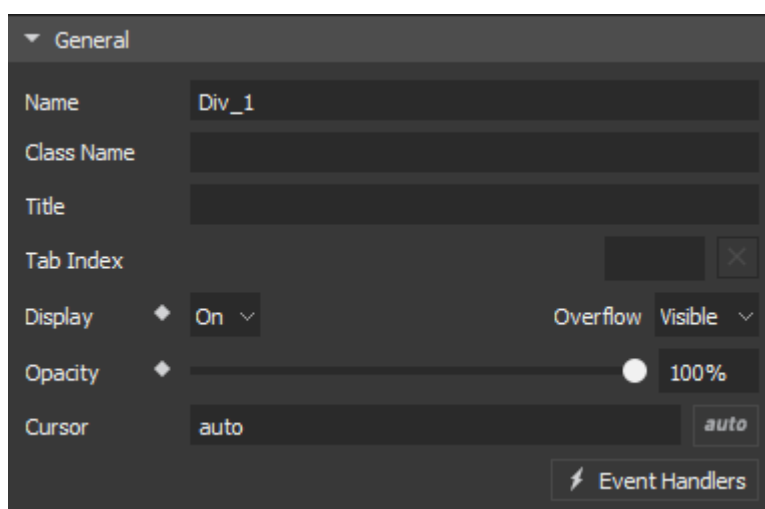
Chapter 12: Using Properties Pane

The **Properties** pane displays properties of the selected elements. It has many sections, and each section shows a group of related properties with editable text fields, lists, and other controls to adjust properties. You can click each section's title bar to collapse or expand a section.

The **Properties** pane constantly updates itself depending on the selected element type. A number of properties are consistent and appear despite what type of elements you select. In contrast, other properties are specific to a certain type of elements. For example, the **Media** section is only available for audio and video elements.

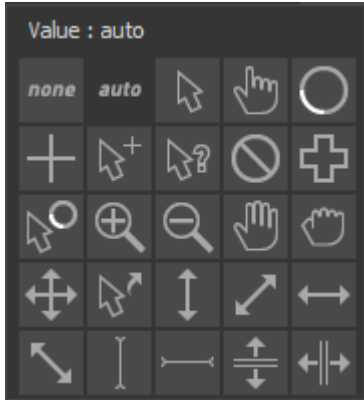
General Section

The **General** section displays information about elements and several properties.







Property	Remark	Animatable
Name	Name of the element. This property can be changed but has to be unique.	No
Class Name	Class name of the element. This property can be used by CSS and JavaScript to perform certain tasks for elements with the specified class name.	No

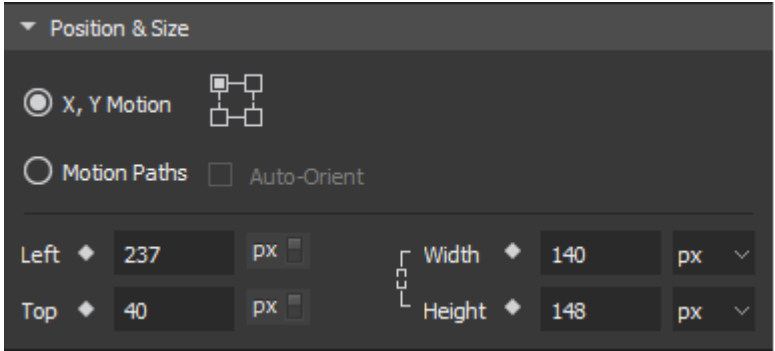
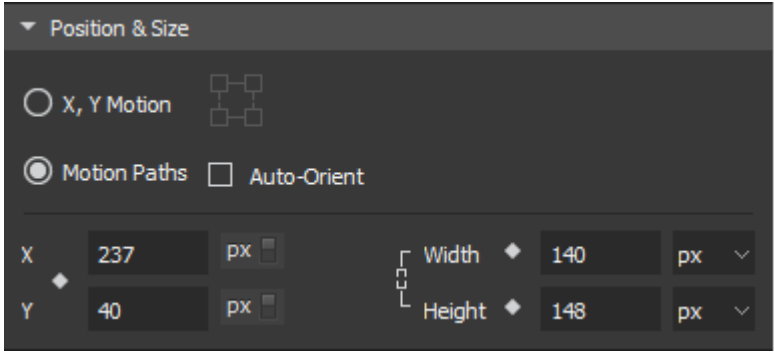
Property	Remark	Animatable
Title	Title of the element, shown when you hover over the element in a browser.	No
Tab Index	<p>Indicate if the element can be focused using the TAB key. Possible values:</p> <ul style="list-style-type: none"> • Empty: The element is not focusable. • -1: The element is focusable by mouse/touch but not be reachable using sequential keyboard navigation. • 0: The element should be focusable in sequential keyboard navigation, but its order is defined by the document's source order. • > 1: The element should be focusable in sequential keyboard navigation, with its order defined by the value of the number. 	No
Display	<p>Set the visibility state of the element. In animation, use the Display property to hide or show an element instantly.</p> <ul style="list-style-type: none"> • On: The element appears on the scene or in the output. • Off: The element doesn't appear on the scene or in the output. 	Yes
Overflow	<p>Determine how the content of the element is displayed if it overflows the element's bounds, and whether a scroll bar should appear.</p> <ul style="list-style-type: none"> • Visible: The overflow isn't clipped; it renders outside the element's box. This is default. • Hidden: The overflow is clipped; any content that extends beyond the box will be hidden. • Scroll: The overflow is clipped, but a scroll bar is added to view the rest of the content. • Auto: If the overflow is clipped, a scroll bar should be added to view the rest of the content. 	No


Property	Remark	Animatable
Opacity	<p>Set the opacity (also transparency) of the element in the range [0, 100%]. Lower values make the element more transparent. The Opacity property is often used to create fade-in and fade-out effects.</p> <p>To change the opacity, drag the opacity slider left or right.</p>	Yes
Cursor	<p>Set which mouse cursor is displayed when the mouse pointer is over an element. The default value is Auto.</p> 	No

Position & Size Section

The **Position & Size** section controls the position and size of element.

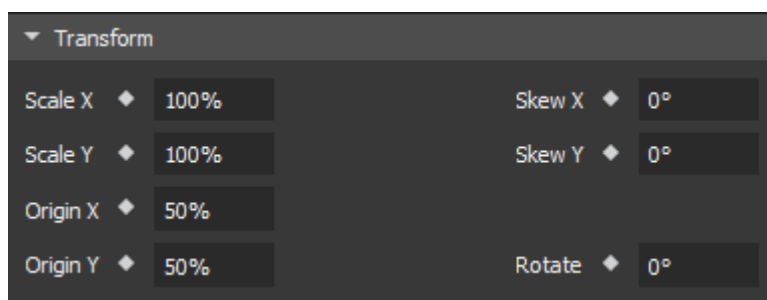
Property	Remark	Animatable
X, Y Motion	<p>By default, elements move in the X, Y Motion mode with X and Y transitions. Change elements' position and size by relative positions:</p> <ul style="list-style-type: none">  Relative to top and left (default).  Relative to top and right.  Relative to bottom and right.  Relative to bottom and left. 	Yes

Property	Remark	Animatable
	 <ul style="list-style-type: none"> • Left/Right: Set the distance in pixels (px) from the left/right edge of the scene or the enclosing container, or set the distance from the left/right as a percentage (%) of the total width of the scene or the enclosing container. • Top/Bottom: Set the distance in pixels (px) from the top/bottom edge of the scene or the enclosing container, or set the distance from the top/bottom as a percentage (%) of the total height of the scene or the enclosing container. 	
Motion Paths	<p>Animate elements along complex curves to mimic real-life motion. You can only switch to Motion Paths when the element's position is "relative to top and left". Otherwise, Motion Paths will be disabled.</p>  <ul style="list-style-type: none"> • Auto-Orient: Orient elements in relation to the direction of the motion path. • X, Y (px, %): Set the position of the element in Motion Paths mode. 	Yes

Property	Remark	Animatable
Width, Height	<ul style="list-style-type: none"> Width: Set the width of the element in pixels (px), or set the percentage (%) of the element width compared to the total width of the scene or the enclosing container. Height: Set the height of the element in pixels (px), or set the percentage (%) of the element height compared to the total height of the scene or the enclosing container element. Link width to height : Maintain the aspect ratio while editing. <p>In the responsive output, maintain the aspect ratio by setting either the width or height unit to percentages (%) and the other unit to auto. For example, set the width unit to % and the height unit to auto. That way, the element height will change proportionally with the width while the width responds to window or viewport size changes.</p>	Yes

Transform Section

The **Transform** section allows you to scale, skew, and rotate elements in two dimensional space.

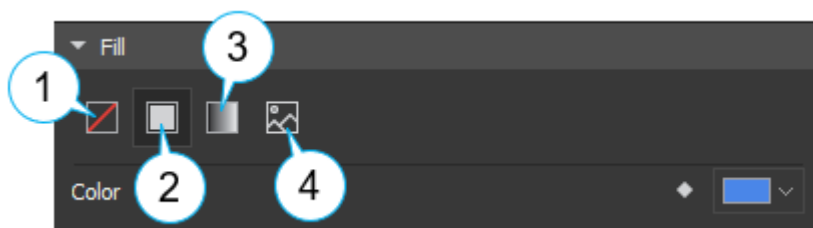


Property	Remark	Animatable
Scale X, Y	Scale the element horizontally or vertically in the range [0, 100%].	Yes

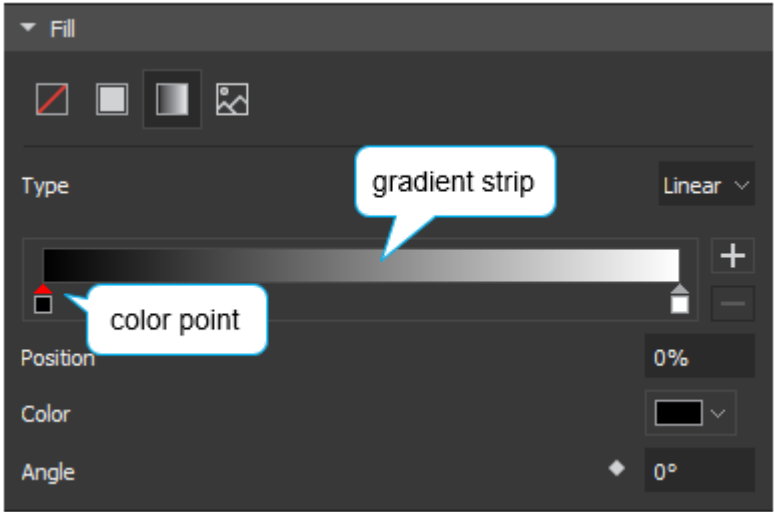
Property	Remark	Animatable
Skew X, Y	Skew the element horizontally or vertically in the range [-60, 60°].	Yes
Origin X, Y	Set the transformation-origin point in the range [0, 100%] around which the element rotate.	Yes
Rotate	Rotate the element. Negative angles make the element rotate counter-clockwise.	Yes

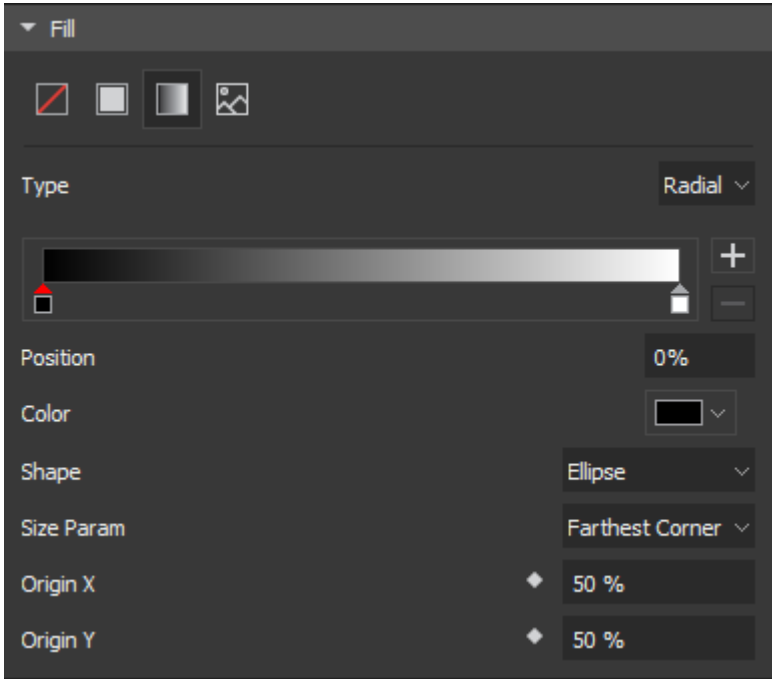
Fill Section

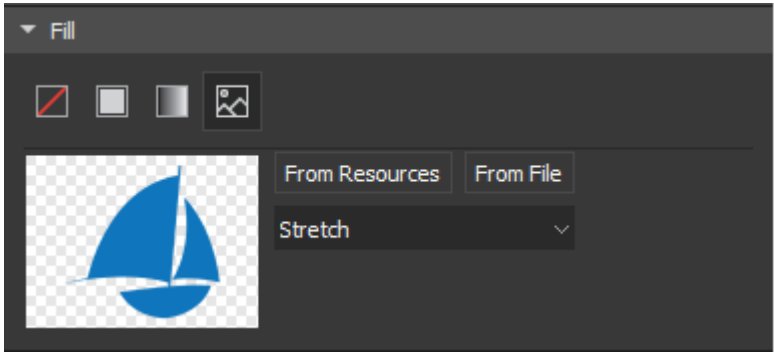
The **Fill** section sets the background fill style of elements. Four possible values: No Fill (1), Solid Fill (2), Gradient Fill (3), and Image Fill (4).



Value	Remark
No Fill	Fill the element with a transparent background (alpha = 0).
Solid Fill	Fill the background with a solid color. Solid Fill uses the RGBA color picker. The default alpha (A) is 255. The color is animatable.
Gradient Fill	<p>Fill the background with a gradient where two or more colors (with or without alpha transparency) gradually fade into other.</p> <div data-bbox="732 1675 1098 1832" data-label="Image"> </div> <p>Select either of the two gradient types from the Type combo box:</p>

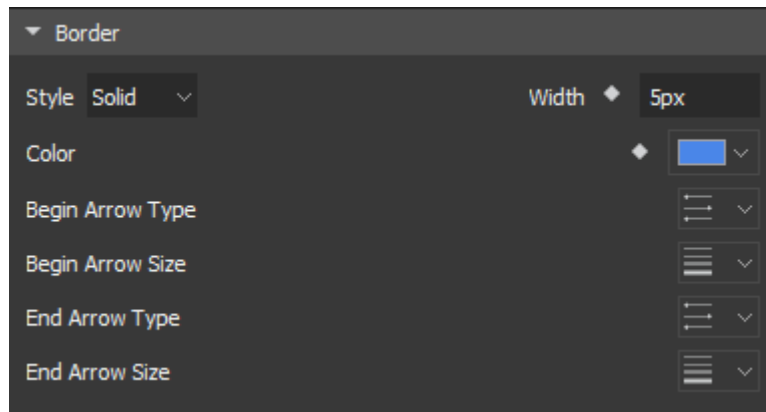
Value	Remark
	<ul style="list-style-type: none"> • Linear gradient creates a straight progression between two or more colors. A linear gradient can go horizontally (left, right), vertically (up, down), or diagonally. • Radial gradient creates a circular progression between two or more colors. It radiates from its center, either in an ellipse or in a circle. <p>a. Linear Gradient</p>  <ul style="list-style-type: none"> • Gradient strip: Show a “live” sample of the gradient. Use it to make changes to the gradient in a visual manner. • Color point: Each color point on the strip is a distinct color. The selected color point has a red triangle at the top. To add a color point, click on the strip or click Add (plus icon). In contrast, click Remove (minus icon) to remove a selected color point. • Position: Specify the position of a color point, where you want a color to “start”. To change a color point’s position, drag it along the strip or enter a new number in the Position box. • Color: Use the color picker to define a color (with or without transparency). • Angle: Set the angle of direction for the gradient line. By default, a linear gradient runs vertically from bottom to top with an angle of 0 degree. As the degree increases, the gradient direction rotates clockwise. Negative angles make the gradient run in the counter-clockwise direction.




Value	Remark
	<p>b. Radial Gradient</p>  <ul style="list-style-type: none"> • The Gradient strip, Color point, Position, and Color of radial gradients are much the same as those of linear gradients. • Shape: Define the shape of the gradient. The value can be either ellipse or circle, of which ellipse is the default. • Size Param: Define the size of the ending shape (the shape where the last color ends). Four possible values: <ul style="list-style-type: none"> ○ Farthest corner: The gradient ends at the corner that is farthest from the shape's center. This is the default value. ○ Closest corner: The opposite of Farthest corner, where the gradient ends at the corner that is closest to the shape's center. ○ Farthest side: The gradient ends at the side farthest from the shape's center. ○ Closest size: The opposite of Farthest side, where the gradient ends at the side closest to the shape's center. • Origin X, Y: Adjust the location of the center point. For example, Origin X, Y [value: 50%, 50%] sets the gradient at the center (default). Meanwhile, Origin X, Y [value: 0%, 0%] sets the

Value	Remark
	<p>gradient to start at the top left. Animate these properties to create animation where the center point moves from this position to that position.</p> <p>Note: Saola Animate is compatible with: Chrome 4+, Firefox 3.5+, IE 9+, Edge, Safari 3.1+, and Opera 11.5+. But for CSS gradient fill, newer versions are required: Chrome 10+, Firefox 3.6+, IE 10+, Safari 5.1+, and Opera 12.1+.</p>
Image Fill	<p>Fill the background with an image. Images can be chosen from the Resources pane or from your computer. Five possible fill options:</p>  <ul style="list-style-type: none"> • Stretch: Extend the image to all edges of the container by distorting or shrinking it. Stretch may only work for some abstract images, not the “figurative” ones. This is the default. • Center: Center the image inside the container. If the image is smaller than the container, you’ll see a transparent area around the edges. Otherwise, you’ll see the very center of the image, with the rest outside of view. Choose this for images with the center focus. • Tile: Repeat the same image throughout the background, making multiple tiles of the image. It is meant for small images or patterns. • Fit: Scale the image as large as possible without cropping or stretching so that the whole image is shown. The empty portion of the container is made transparent. • Fill: Scale the image as large as possible without stretching. If the proportion of the image differs from the container, the image is cropped so that it covers the entire container.

Border Section

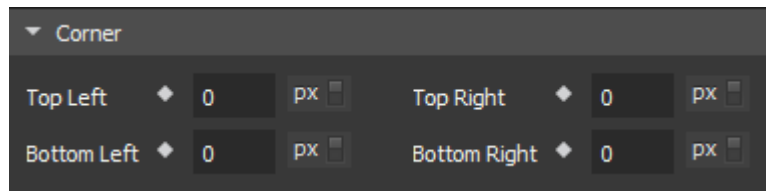
The **Border** section defines the border style of elements.



Property	Remark	Animatable
Style	Set the border style of the element. Four possible values: <ul style="list-style-type: none"> None Solid  Dotted  Dashed  	No
Width	Set the width of the border in pixels (px).	Yes
Color	Set the color of the border.	Yes
Begin Arrow Type	Set the type of the begin arrow.	No
Begin Arrow Size	Set the size of the begin arrow.	No
End Arrow Type	Set the type of the end arrow.	No
End Arrow Size	Set the size of the end arrow.	No

Corner Section

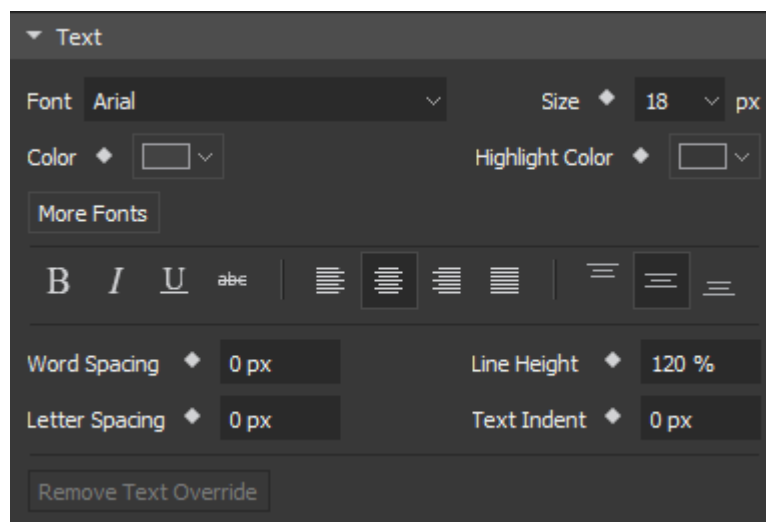
The **Corner** section is used for **DIVs** only to define the shape of the DIV border.



Property	Remark	Animatable
Top Left	Set the border top-left radius.	Yes
Top Right	Set the border top-right radius.	Yes
Bottom Left	Set the border bottom-left radius.	Yes
Bottom Right	Set the border bottom-right radius.	Yes

Text Section

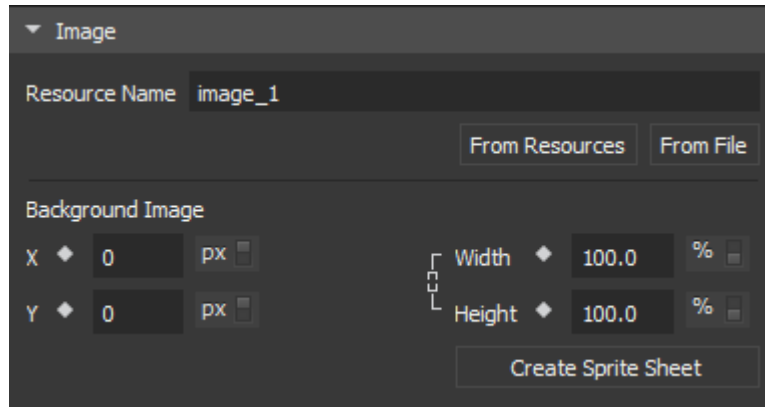
The **Text** section allows you to set and animate many text properties such as text size, color, line height, etc. This section is available for all types of elements except media (audio and video), symbol, and group elements.



Property	Remark	Animatable
Font	Set the font family of text.	No
Size	Set the font size of text in pixels.	Yes
Color	Set the color of text.	Yes
Highlight Color	Set the highlight color of text.	Yes
Bold	Make the text bold.	No
Italic	Make the text italic.	No
Underline	Underline the text.	No
Strikethrough	Strike the text through.	No
Horizontal Alignment	Align text horizontally. Possible values: Left, Center, Right, and Justify.	No
Vertical Alignment	Align text vertically. Possible values: Top, Middle, and Bottom.	No
Subscript/ Superscript	Format the text as superscript/subscript.	No
Word Spacing	Set the space between words in pixels.	Yes
Letter Spacing	Set the space between letters in pixels.	Yes
Line Height	Set the space between lines in percentages, relative to font size value.	Yes
Text Indent	Set the indent of each text block in pixels (paragraph).	Yes
Remove Text Override	Remove inline text overrides.	No

Image Section

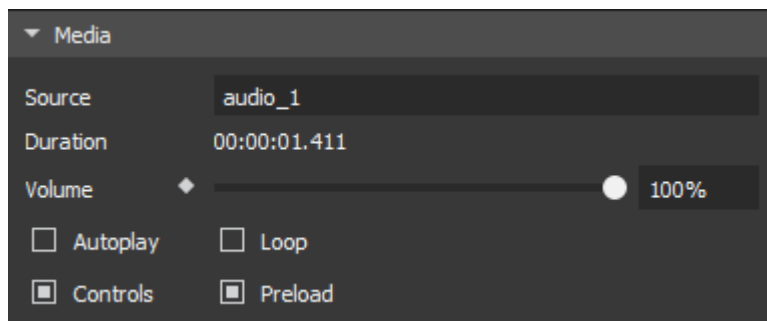
The **Image** section is available for **image elements** only to display image-specific properties.



Property	Remark	Animatable
File Name	Name of the image embedded within the image element.	No
From Resources	Select another image from the Resources pane to replace with the current image.	No
From File	Select another image from your computer to replace with the current image.	No
Background Image X, Y	Set the exact position of the background image.	Yes
Background Image Width, Height	Set the size of the background image.	Yes
Create Sprite Sheet	Create Sprite Sheet Animation .	No

Media Section

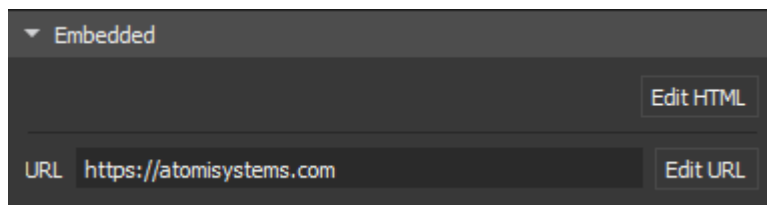
The **Media** section displays information about the audio/video and enables you to define some media options. This section is used for **audio and videos** only.



Property	Remark	Animatable
Source	Name of the audio/video. This property cannot be changed.	No
Duration	Duration of the audio/video in milliseconds. This property cannot be changed.	No
Volume	Set the volume of the audio/video in the range [0, 100%].	Yes
Autoplay	Allow the audio/video to automatically play on the timeline and desktop browsers.	No
Loop	Make the audio/video replay automatically from the beginning after it finishes.	No
Controls	Display the default player controls for the audio/video.	No
Preload	Preload the audio/video before the document loads.	No

Embedded Section

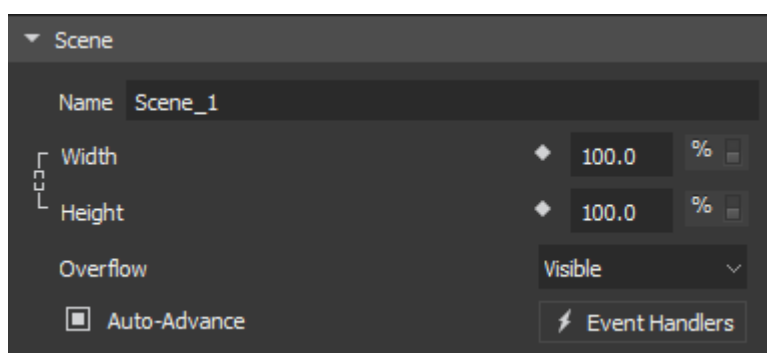
The **Embedded** section display information about the embedded HTML. It is used for **HTML widgets** only.



Property	Remark	Animatable
Edit HTML	Edit the embedded HTML.	No
URL	Address of the webpage displayed within the HTML widget.	No
Edit URL	Change the content embedded within the HTML widget.	No

Scene Section

The **Scene** section displays properties related to **scenes**.

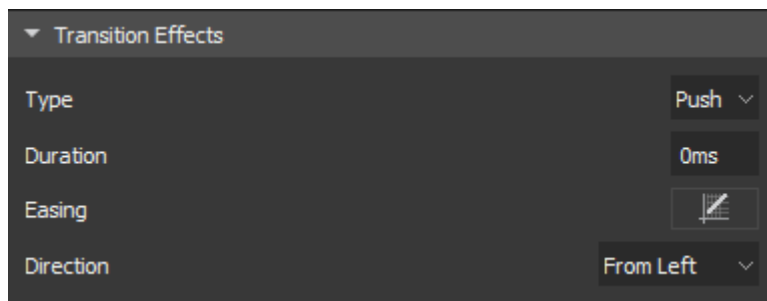


Property	Remark	Animatable
Name	Name of the scene. This property can be changed but has to be unique.	No

Property	Remark	Animatable
Width, Height	<p>Set the exact width and height of the scene. The size can be a fixed value (in pixels) or relative to the container size (in percentages).</p> <p>To maintain the aspect ratio while editing, select the Link width to height check box (the chain icon).</p>	Yes
Overflow	<p>Determine how the content of the scene is displayed if it overflows the scene's box, and whether a scroll bar should appear.</p> <ul style="list-style-type: none"> • Visible: The overflow isn't clipped; it renders outside the scene's box. This is default. • Hidden: The overflow is clipped; any content that extends beyond the box will be hidden. • Scroll: The overflow is clipped, but a scroll bar is added to view the rest of the content. • Auto: If the overflow is clipped, a scroll bar should be added to view the rest of the content. 	No
Auto-Advance	<p>Jump to the next scene automatically when the current scene ends. If you disable this feature, use actions to play scenes.</p>	No
Event Handlers	<p>Open the Scene Event Handlers dialog to control scene events and actions.</p>	No

Transition Effects Section

The **Transition Effects** section displays properties related to scene transition effects. It helps define transition effects between **scenes**.



Property	Remark	Animatable
Type	Type of the transition effect. Possible values: <ul style="list-style-type: none"> None (default type, meaning no effect) Fade Push 	No
Duration	Duration of the transition in milliseconds.	No
Easing	The timing function that updates the transition by time.	No
Direction	Direction of the Push transition effect. Possible values: From Left, From Top, From Right, From Bottom	No

Chapter 13: Using Resources Pane

The **Resources** pane lists all resources added to a project and provides many controls for managing resources. To show the pane, click **View > Resources**. Saola Animate supports the following resource types and extensions:


- **Image:** Image files (*.png, *.jpg, *.jpeg, *.bmp, *.gif, *.svg)
- **Audio:** Audio files (*.mp3, *.wav, *.m4a, *.aac, *.ogg, *.oga)
- **Video:** Video files (*.mp4, *.m4v, *.webm, *.ogv)
- **JavaScript:** JavaScript files (*.js)
- **CSS:** Style sheet files (*.css)
- **HTML:** HTML files (*.html)
- **Symbol:** Saola Animate symbols (*.saolasym)
- **Font:** **Google fonts** added to a project

The **Resources** pane offers control for filtering resource types. Just click the **Resource Type** combo box and select a type you want to see. You can also click the **View Style** combo box to change the view style. Select one of the following views:

- **List:** Each resource appears as a small icon with a label to the right.
- **Small:** Each resource appears as a small icon with a label below it. This is the default view style.
- **Medium:** Each resource appears as a medium icon with a label below it.

Managing Resources

Importing Resources

Simply drag files to the **Resources** pane to import them. Alternatively, in the **Resources** pane, click **Import**  **> From File** and choose the files you want from your computer.

For JavaScript and CSS resources, you can directly import files from URL by clicking **Import > From URL** and specify the URL.

After importing resources, you can:



- **Rename resources:** Select a resource and press **F2** on your keyboard, or right-click it and select **Rename**.

- **Preview resources:** You can preview images, audio, and videos. Double-click an image/audio/video or right-click it and select **Preview**. The **Preview** window appears, allowing you to preview the file.
- **Add resources to scene:** Drag multiple resources directly on to the Canvas or right-click them and select **Add to Scene**.


Note: Besides the supported resource types and extensions, Saola Animate lets you import any other file types into your project. These "misc" resources will be exported with the project and stored in the HTML output's resource folder.

Removing Resources

You can delete any resources that aren't in use, meaning that you have to remove all scene elements using a resource before you can delete it from the **Resources** pane. To figure out

which resources that aren't actively used, click **Select Unused Resources** . Then, press **DELETE** to remove the resources you don't want. You can also click **Delete**  on the toolbar or in the context menu to get the same result.

Exporting Resources

You can export one or multiple resources at the same time by clicking **Export** . If you're going to export a single resource, provide the file name for the file before exporting. Otherwise, if you want to export multiple resources, specify the folder path to save the files.


Editing Symbol Resources

The **Resources** pane holds any symbols you create and import into a project. Double-click a symbol to edit it as a normal project.

Script Resources

Editing Script Resources

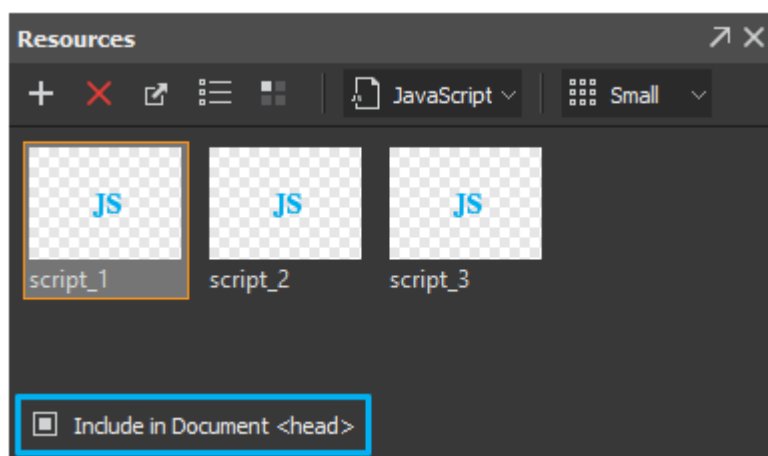
Script resources include JavaScript, CSS, and HTML files. Double-click a script resource to edit it in the script editor. This editor is similar to the **JavaScript Function** editor except that:

- It allows editing multiple script resources at once. Click **Open from Resources**  on the toolbar and choose other resources.

- You can discard all changes and return to the original files at any time by clicking **Restore to Original Files** in the bottom left corner of the editor. Alternatively, right-click a script resource and select **Update from Original Files**.



Including JavaScript and CSS in Document <head>

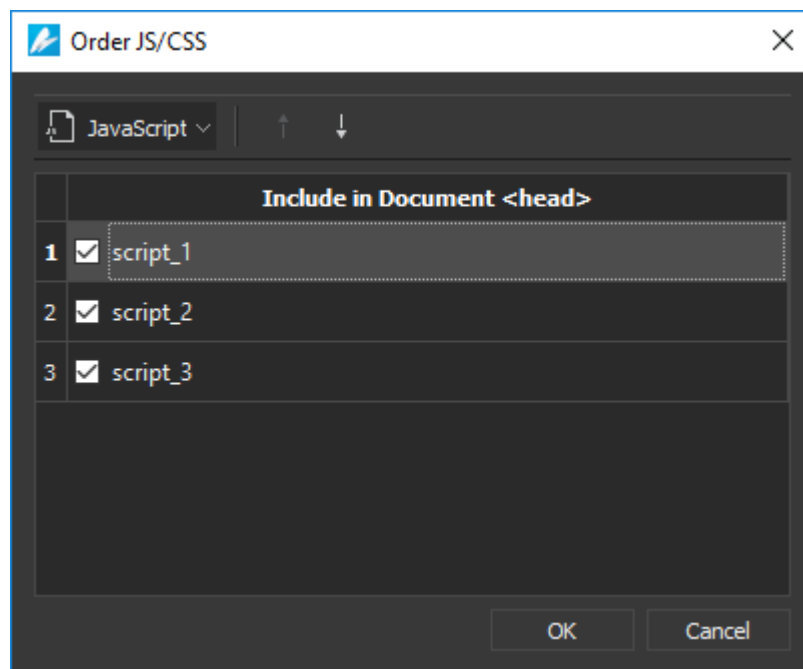
By default, JavaScript and CSS files added to a project are included in the document <head>. To disable this behavior, select the JavaScript or CSS file you don't want to be included in the document header and clear the **Include in Document <head>** check box.



Ordering JavaScript and CSS

The order in which the JavaScript or CSS files are stated matters in many cases. One script may contain dependencies on another script, so you have to make sure the dependent script is called after the other one. Meanwhile, the order of the CSS files determines which ones take precedence. CSS rules that are defined earlier have lower precedence than those defined later, meaning that styles declared later win.

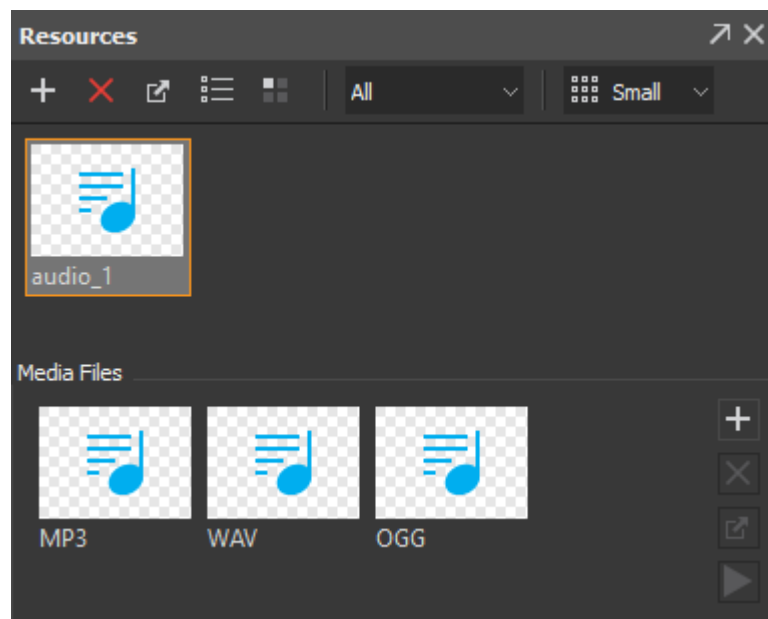
To order JavaScript and CSS resources, click **Order JS/CSS** to open the corresponding dialog. Then, choose the resource type and click the **Move Up**  or **Move Down**  button to reorder the files. You can also specify whether to include a specific file in the document <head> by selecting the check box next to the file's label.




Resource Groups for Audio and Videos

Adding audio and videos automatically creates an audio and video group to which additional formats can be added. That way, you provide a single audio/video with multiple sources for cross-browser compatibility. A browser can select the most compatible format it supports from a resource group, ignoring the other formats.


In the **Resources** pane, whenever you select an audio or video resource, the **Media Files** widget appears. Using buttons on the right sidebar or in the context menu of the selected format, you can manage multiple sources in a resource group.



Adding Formats to Groups

Click **Add**  to add additional formats to a group. If the format has already existed, you will be asked to overwrite it.

Removing Formats from Groups

Select a format and click **Delete**  to remove it from a group. If a group contains only one format, you cannot remove it.

Exporting Formats

Select a format and click **Export**  to export it to file. Provide the file name for the file before exporting.

Previewing Formats

Double-click a format to preview it in the **Preview** window. Alternatively, click **Preview**  on the toolbar or in the context menu.


Chapter 14: JavaScript

Saola Animate exports content to documents using a combination of HTML5, CSS3, and JavaScript. Users can register a JavaScript function with an event in the **Event Handlers** dialog. There are two main types of functions:

- **Normal functions:** Normal JavaScript functions that can be called in other functions. You can modify the parameters of this function type.
 - **Event handler functions:** Functions that are registered with events and have fixed parameters. Event handlers are separated into five types:
 - Document Event Handlers
 - Scene Event Handlers
 - Timeline Event Handlers
 - Element Event Handlers
 - Preloader Event Handlers
-

Creating Functions

Create a new function in the **Functions** pane. If you don't see this pane, click **View > Functions**.

1. Click **New Function**  and select the function type you want. The **JavaScript Function** editor appears, allowing you to write the script.
2. Write the script.
3. Click **OK** to apply changes and close the dialog. The function is created and listed in the pane.

Another way to create a new function is from the **Event Handlers** and **Timeline Triggers** dialogs. Do the following:

1. Add the Run JavaScript action.
2. Double-click the Params cell > **Add Function**.

Removing Functions

Select one or more functions and press **DELETE** to remove them. Alternatively, click **Delete**



on the toolbar.

Renaming Functions

Right-click a function and select **Rename**. Type a new name and press **ENTER** or any elsewhere outside to submit changes. Note that function names must not start with a number.

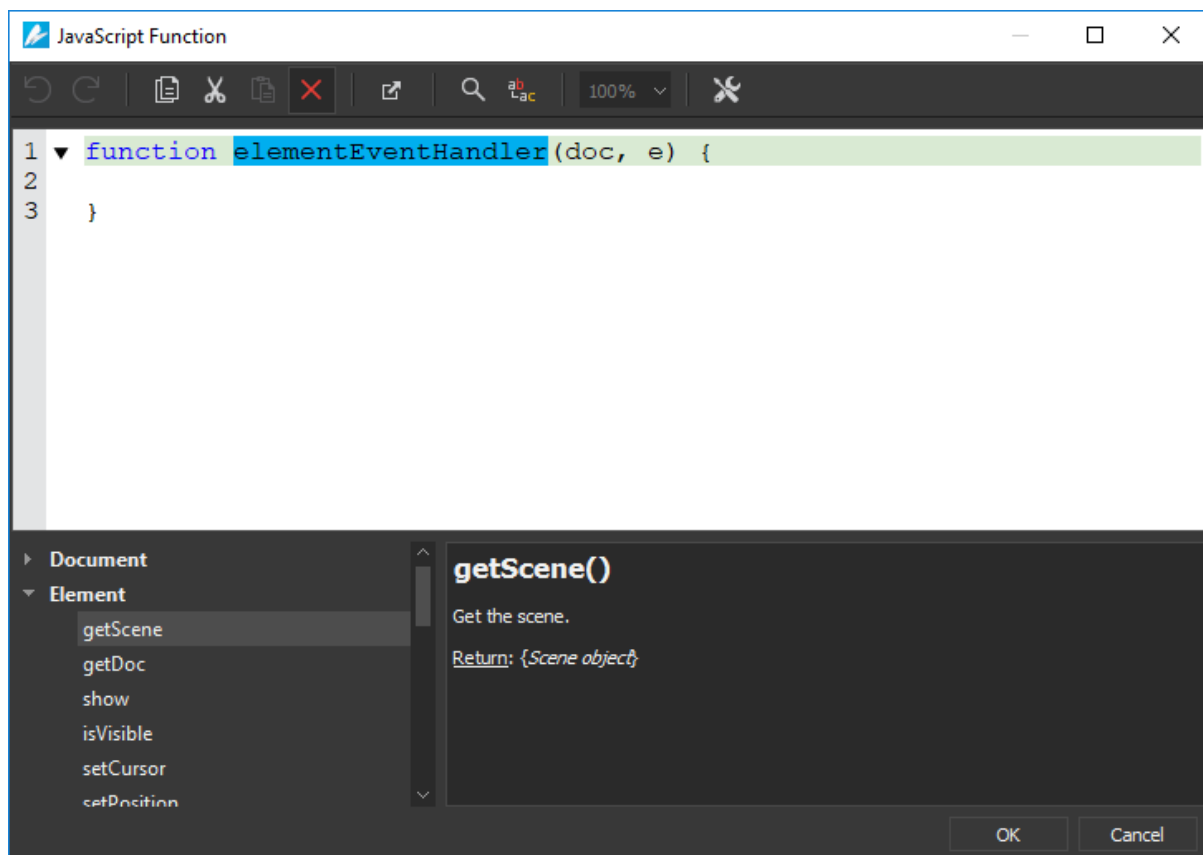
Exporting Functions to JavaScript Files

Select one or more functions and click **Export**  on the toolbar. Provide the file name for the exported file.

Editing Functions

Double-click a function to edit it in the **JavaScript Function** editor. For normal functions, you can edit their names, contents, and parameters. For event handler functions registered with events, you can only edit their names and contents because their parameters are set to fixed.

The editor comes with several features that support and speed up the coding process, including code suggestion, code folding, line number, etc. Toggle these features by clicking Options on the toolbar. For more details about the script editor, see **Code Editor Tab** and **Shortcuts for Script Editor**.

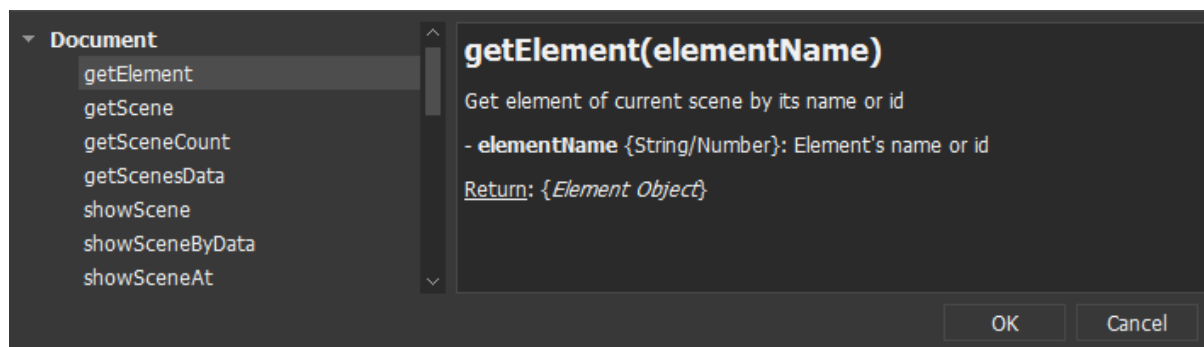


Code suggestion is a remarkable feature which predicts the rest of a word or sentence while you are typing. Press the **TAB** key to accept a suggestion, or the **DOWN ARROW** key to select and accept one of the several suggestions. To manually show suggestion, press **CTRL+SPACE** when editing.

You can also insert APIs from the **JavaScript API Browser**.

JavaScript APIs

Saola Animate publishes many methods and properties of common element types. Show these methods in the **JavaScript Function** editor by clicking **Options > Show API** on the toolbar. These methods will be listed with in-app documentation in the pane below the editing area and can be inserted into the editing area quickly by double-clicking.



Document

getElement(elementName)

Get the element of the current scene by name.

Params:

- elementName: String. Element's name.

Return: Element object.

getScene()

Get the current scene of the document.

Return: Scene object.

getSceneCount()

Get the number of the scenes.

Return: Number.

showScene(sceneName)

Show the scene by name.

Params:

- sceneName: String. Scene's name.

Return: Boolean. True if the scene was found, false otherwise.

showSceneAt(index)

Show the scene at the given index.

Params:

- index: Number. The index of the scene, the first scene has 0 index.

Return: Boolean. True if the scene was found, false otherwise.

showNextScene()

Show the next scene from the current scene.

Return: Boolean. True if the next scene exists, false otherwise.

showPreviousScene()

Show the previous scene from the current scene.

Return: Boolean. True if the previous exists, false otherwise.

getDuration()

Get the total duration (in milliseconds) of all Main Timelines of all scenes and scene transitions.

Return: Number. Duration in milliseconds.

setSpeed(speed)

Set the playback speed of the document.

Params:

- speed: Number. The playback speed of the document must be a positive value (e.g. 0.5, 1). Normal speed: 1.

Return: undefined.

getSpeed()

Get the playback speed of the document.

Return: Number.

setPlayBackward(backward)

Set the playback direction.

Params:

- backwards: Boolean. Set the playback to backward if undefined or true, forward if false.

Return: undefined.

isPlayBackward()

Check if the playback direction is backward.

Return: Boolean. True if play backward, false if forward.

play()

Play the mainstream (Main Timeline or scene transition).

Return: undefined.

pause()

Pause the mainstream (Main Timeline or scene transition).

Return: undefined.

isPlaying()

Check if the mainstream is playing or paused.

Return: Boolean. True if the mainstream is playing, false if paused.

getTimeline(timelineName)

Get the timeline of the current scene by name. If timelineName is undefined, the timeline is the Main Timeline.

Params:

- timelineName: String. Timeline's name.

Return: Timeline object.

Timeline

getDoc()

Get the current document.

Return: Document object.

getTimestamp()

Get the current timestamp (in milliseconds), relative to the beginning of the timeline.

Return: Number.

getDuration()

Get the duration (in milliseconds) of the timeline. Do not take media and symbols into account.

Return: Number.

play(timestamp, speed, backward)

Play the timeline at the given timestamp with speed and playback direction.

Params:

- timestamp: Number/String. Timeline's timestamp or label. Play from the current position if not specified.
- speed: Number. Playback speed. Playback speed. Keep the current speed if this speed is undefined or invalid (≤ 0).
- backward: Playback direction. Keep the current direction if undefined, play backward if true, forward if false.

Return: undefined.

pause(timestamp)

Pause the timeline at the given timestamp

Params:

- timestamp: Number/String. Timeline's timestamp or label. Pause at the current position if not specified.

Return: undefined.

isPlaying()

Check if the mainstream is playing or paused.

Return: Boolean. True if the mainstream is playing, false if paused.

seek(timestamp)

Seek timeline to timestamp.

Params:

- timestamp: Number/String. Timeline's timestamp or label.

Return: undefined.

setSpeed(speed)

Set the playback speed.

Params:

- speed: Number. Must be positive value, e.g. 0.5, 1, 2,...

Return: undefined.

getSpeed()

Get the playback speed.

Return: Number.

setPlayBackward(backward)

Set the playback direction.

Params:

- backward: Boolean. Play backward if undefined or true, forward if false.

Return: undefined.

isPlayBackward()

Check if the playback direction is backward.

Return: Boolean. True if play backward, false if forward.

Element

getScene()

Get the scene.

Return: Scene object.

getDoc()

Get the current document.

Return: Document object.

show(show)

Show/Hide the element.

Params:

- show: Boolean. Show if true or undefined, hide if false.

Return: undefined.

isVisible()

Get the visibility state.

Return: Boolean.

setCursor(cursor)

Set the cursor for the element, following CSS cursor.

Params:

- cursor: String. Follow **CSS Cursor**.

Return: undefined.

setPosition(left, top)

Set the position of the element.

Params:

- left, top: Number/String. Value of left/top.

Return: undefined. The left/top can be number (e.g. 50) or string (e.g. '50px', '50%'). If the left/top is number, the unit is pixel.

setLeft(left)

Params:

- left: Number/String. Value of left.

Return: undefined. The left can be number (e.g. 50) or string (e.g. '50px', '50%'). If the left is number, the unit is pixel.

setTop(top)

Params:

- top: Number/String. Value of right.

Return: undefined. The top can be number (e.g. 50) or string (e.g. '50px', '50%'). If the top is number, the unit is pixel.

setRight(right)

Params:

- right: Number/String. Value of right.

Return: undefined. The right can be number (e.g. 50) or string (e.g. '50px', '50%'). If the right is number, the unit is pixel.

setBottom(bottom)

Params:

- bottom: Number/String. Value of bottom.

Return: undefined. The bottom can be number (e.g. 50) or string (e.g. '50px', '50%'). If the bottom is number, the unit is pixel.

setSize(width, height)

Params:

- width, height: Number/String. Width/Height of the element.

Return: undefined. The width/height can be number (e.g. 50) or string (e.g. '50px', '50%'). If the width/height is number, the unit is pixel.

setWidth(width)

Params:

- width: Number/String. Width of the element.

Return: undefined. The width can be number (e.g. 50) or string (e.g. '50px', '50%'). If width is number, the unit is pixel.

setHeight(height)

Params:

- height: Number/String. Height of the element.

Return: undefined. The height can be number (e.g. 50) or string (e.g. '50px', '50%'). If the height is number, the unit is pixel.

setBorderStyle(border)

Set the border style of the element.

Params:

- border: CSS border style value. Possible values: 'none', 'solid', 'dotted', 'dashed'.

Return: undefined

setBorderWidth(width)

Set the border width of the element.

Params:

- width: Number/String. The width can be number (e.g. 10) or string (e.g. '10px', '10%'). If the width is number, the unit is pixel.

Return: undefined.

setBorderColor(color)

Set the border color of the element.

Params:

- color: String. Name of the color, following CSS color, e.g. 'red', 'rgb(255, 0, 0)'.

Return: undefined.

fillSolid(color)

Fill the background with a solid color.

Params:

- color: String. Name of the color, following CSS color, e.g. 'red', 'rgb(255, 0, 0)'.

Return: undefined.

fillNone()

Set the fill style to nofill.

Return: undefined.

setTitle(title)

Set the tittle (tooltip) for the element.

Params:

- title: String. Title of the element.

Return: undefined.

setText(htmlText)

Set html text. Do not call this method for elements that can't have text, e.g. image, audio, video, symbol.

Params:

- htmlText: String. Text in html format.

Return: undefined.

Event

preventDefault()

Prevent default actions of the event.

Return: undefined.

stopPropagation()

Stop event propagation.

Return: undefined.

stopImmediatePropagation()

Stop event propagation immediately.

Return: undefined.

Preloader

getProgress()

Get the current loading progress, from 0 - 1.

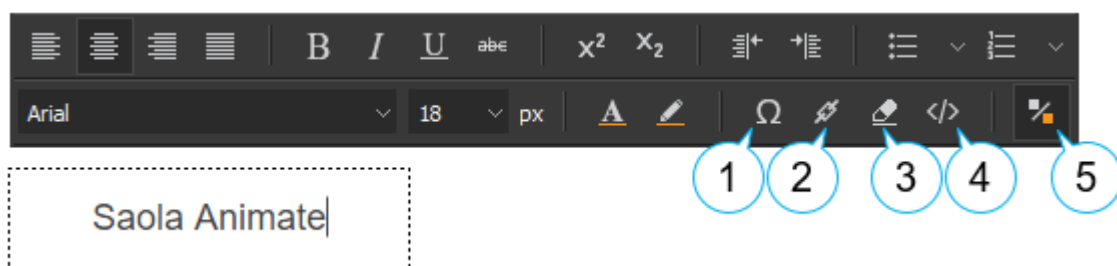
Return: Number.

Chapter 15: Using Text Editor

Saola Animate provides numerous tools for editing rich text. It allows changing text font, size, color, styling, spacing, etc. Find these text editing tools in two widgets: the **Floating Text Format** toolbar and the **Properties** pane.

Text can be added to any text-based elements, including text, DIVs, and shapes. Insert these elements into a scene, then double-click them to open the inline text editor and type text.

To format text, use keyboard shortcuts or commands in the **Floating Text Format** toolbar. You can also set the default text style of elements in the **Properties** pane > **Text** section.



There are some commands and options you need to mind:

1. **Insert Special Character:** Insert a special character.
2. **Insert Link:** Insert a hyperlink into text by specifying both the URL and the anchor text.
3. **Remove Format:** Return the selected text to its default formatting.
4. **Edit HTML:** Edit the HTML of the text directly with valid HTML tags and styles.
5. **Floating Text Format Toolbar:** Show or hide the **Floating Text Format** toolbar.


Chapter 16: Using Color Picker

Saola Animate uses RGBA colors. The color picker lets you select a color visually to change the color of the border, the fill, text, etc. Open the color picker by clicking the **Color** button which can be found in many places such as the **Properties** pane, the **Floating Text Format** toolbar, etc.



1. **Grayscale:** Colors ranging from black to white.
2. **Standard colors:** A list of 10 standard colors.
3. **Color shades and tints:** Colors created by adding black and white to standard colors.
4. **Opacity slider:** Adjust the opacity or transparency of a color in the range of 0 to 255. 0 opacity means completely transparent.
5. **More Colors:** Choose a custom color from the Select Color dialog.
6. **Eyedropper:** Pick a color from within the application window.
7. **Recently used colors:** Colors selected by users (by using **More Colors** and **Eyedropper** options).

Chapter 17: Event - Action References

Events are occurrences that determine when to trigger an action. When an event is detected, actions fire as a response. Events and actions are controlled in the **Event Handlers** dialog. Open this dialog by clicking the **Event Handlers**  buttons which appear as a lightning bolt in many panes.

Events

Event Name	Remark
Element Events	
Mouse Click	Occur when users click an element.
Mouse Double Click	Occur when users double-click an element.
Mouse Down	Occur when users press the mouse button over an element.
Mouse Up	Occur when users release the mouse button over an element.
Mouse Move	Occur when the mouse pointer is moving while it is over an element.
Mouse Over	Occur when the mouse pointer moves onto an element, or onto one of its children.
Mouse Out	Occur when users move the mouse pointer out of an element, or out of one of its children.
Mouse Enter	Occur when the mouse pointer moves onto an element.
Mouse Leave	Occur when the mouse pointer moves out of an element.
Focus	Occur when an element gets focused.
Swipe Left	Occur when users swipe over an element from right to left.

Event Name	Remark
Swipe Right	Occur when users swipe over an element in from left to right.
Scene Events	
Scene Activate	Occur when a scene starts displaying.
Scene Deactivate	Occur right before a scene ends.
Scroll	Occur when users scroll a scene. The scroll bar appears when the scene content extends the window's borders.
Key Down	Occur when users press a keyboard key.
Key Up	Occur when users release a keyboard key.
Orientation Change	Occur when users rotate a mobile device vertically or horizontally.
Window Resize	Occur when a browser window is resized.
Element Events	Scene has all element events (Mouse Click, Mouse Over, Mouse Leave, etc.)
Document Events	
Create	Occur right after a document has been created but hasn't displayed any content yet. Users can bind actions to this event to customize the player or change the document's data.
Ready	Occur after the Document Create event, when a document is ready to play but hasn't displayed any content yet, except a preloader (if any).
Destroy	Occur when a document is being removed from page.
Script Error	Occur when errors arise from the functions used in the Run JavaScript action.

Event Name	Remark
Scene Events	Document has all scene events (Mouse events, Keyboard events, Orientation Change, etc.)
Timeline Events	
Update	Occur when the timeline updates itself constantly when the Playhead reaches every tick mark on the timeline.
Play	Occur when the playback of the timeline changes from pause to play.
Pause	Occur when the playback of the timeline changes from play to pause.
Complete	Occur when the timeline plays until the end and stops.
Speed Change	Occur when the playback speed of the timeline changes.
Reverse	Occur when the playback direction of the timeline changes (forward and backward).
Seek	Occur when users seek a specific time in the timeline, causing an instant jump to a new time.
Preloader Events	
Start	Occur when the preloader starts.
Progress	Occur when the preloader is in progress, from start to end.
Complete	Occur when the preloader loads all resources completely.
Item Start	Occur when the preloader starts loading one resource.
Item Complete	Occur when the preloader finishes loading one resource or a load error happens.

Actions

Action Name	Target Object	Parameter	Remark
Timeline Actions			
Start Timeline	Timeline	Direction (Forward, Backward), Speed (%)	Start the playback of the target timeline.
Play Timeline	Timeline	Direction (Forward, Backward), Speed (%)	Resume the playback of the target timeline.
Play Timeline From	Timeline	Direction (Forward, Backward), Time or Timeline Label, Speed (%)	Resume the playback of the target timeline from a specific time or label.
Pause Timeline	Timeline		Pause the playback of the target timeline.
Pause Timeline At	Timeline	Time or Timeline Label	Jump to and pause at a specific time or label in the target timeline.
Toggle Play/Pause Timeline	Timeline		Toggle the playback of the target timeline.
Seek Timeline	Timeline	Time or Timeline Label	Jump to a specific time or label in the target timeline without affecting whether or not the timeline is playing or paused.
Symbol Actions			
Start Symbol	Symbol	Direction (Forward, Backward), Speed (%)	Start the playback of the target symbol.

Action Name	Target Object	Parameter	Remark
Play Symbol	Symbol	Direction (Forward, Backward), Speed (%)	Resume the playback of the target symbol.
Play Symbol From	Symbol	Direction (Forward, Backward), Time or Timeline Label, Speed (%)	Resume the playback of the target symbol from a specific time or label.
Pause Symbol	Symbol		Pause the playback of the symbol timeline.
Pause Symbol At	Symbol	Time or Timeline Label	Jump to and pause at a specific time or label on the timeline of the target symbol.
Toggle Play/Pause Symbol	Symbol		Toggle the playback of the target symbol.
Seek Symbol	Symbol	Time or Timeline Label	Jump to a specific time or label on the timeline of the target symbol without affecting whether or not the timeline is playing or paused.
Media Actions			
Start Media	Audio, Video	Speed (%)	Start the playback of the target audio or video.
Play Media	Audio, Video	Speed (%)	Resume the playback of the target audio or video.
Play Media From	Audio, Video	Time, Speed (%)	Resume the playback of the target audio or video from a specific time.

Action Name	Target Object	Parameter	Remark
Pause Media	Audio, Video		Pause the playback of the target audio or video.
Pause Media At	Audio, Video	Time	Jump to and pause at a specific time of the target audio or video.
Mute Media	Audio, Video		Mute the target audio or video.
Media Volume	Audio, Video	Volume (%)	Change the volume of the target audio or video.
Toggle Mute/Unmute Media	Audio, Video		Toggle the mute state of the target audio or video.
Mute All Media	None		Mute all audio and video element in a scene.
Toggle Play/Pause Media	Audio, Video		Toggle the playback of the target audio or video.
Seek Media	Audio, Video	Time	Jump to a specific time of the target audio or video without affecting whether or not it is playing or paused.
Play Video Full Screen	Video		Play the target video in full screen.
Visibility Actions			
Hide Element	All Element Types		Hide the target element.
Show Element	All Element Types		Show the target element.

Action Name	Target Object	Parameter	Remark
Toggle Hide/Show Element	All Element Types		Toggle the visibility state of the target element.
Other Actions			
Jump To Scene	Scene	Transition Type, Duration (ms), Easing, Direction	Change to the target scene.
Go To URL	None	URL, New Window (True, False)	Open a webpage in a browser.
Change Text	Div, Text, Shape	Text	Change the text that appears in the target element.
Run JavaScript	None	Function (see JavaScript)	Invoke a JavaScript function.

Chapter 18: Customizing Preloader

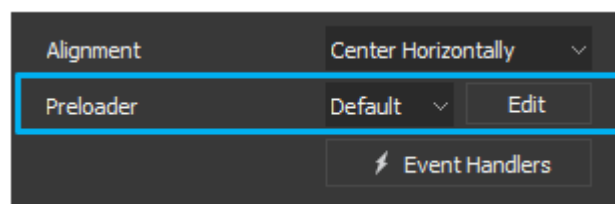
When your animation loads from a web server, images, media files, and all other components will be transferred to the user's browser. If your project is complex and graphics-intensive, it may take a few seconds to download. It is when you need to use a preloader to indicate that the content is being loaded.

By default, Saola Animate documents preload all images before showing a scene. To show the loading progress, they display a loading indicator including a spinner and a percentage sign.



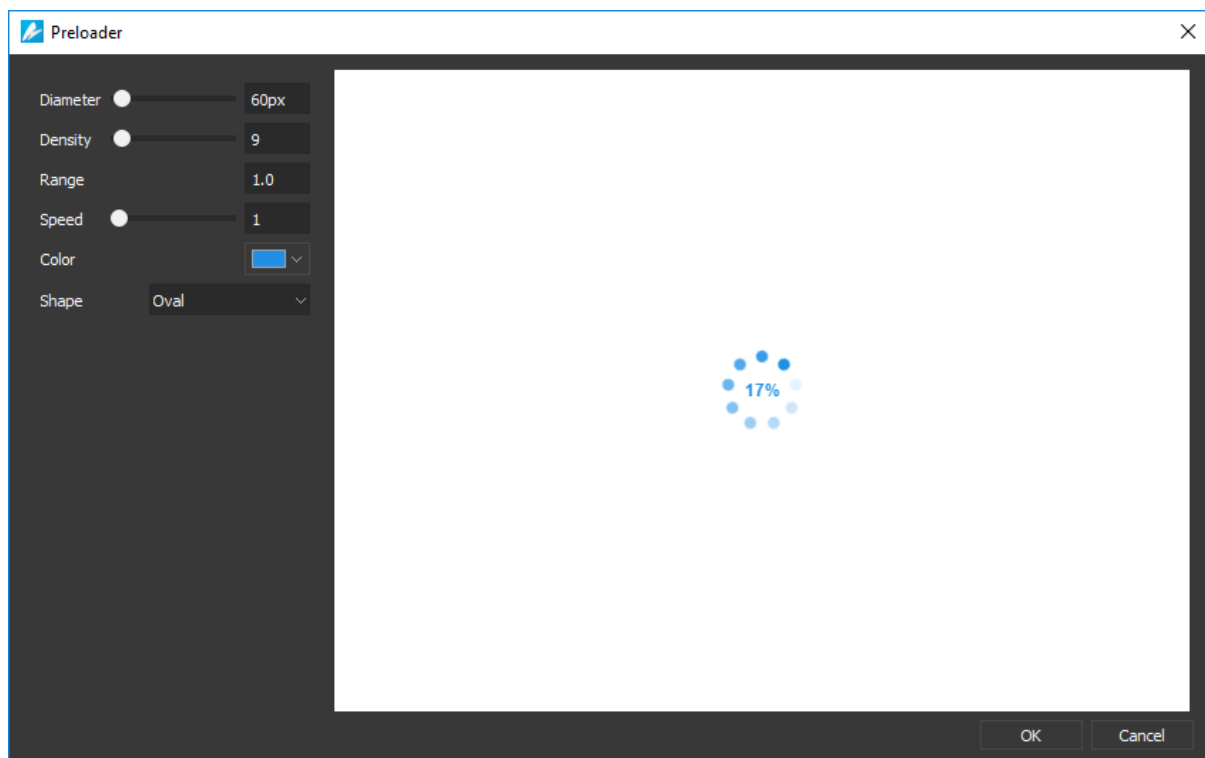
You can customize this loading indicator or create your own loading screen. Do the following:

1. Open the **Document** pane. If you don't see this pane, click **View > Document**.
2. Saola Animate offers two preloader types: **Default** and **Custom**. Select either of the two and click **Edit** to modify as you like.



Default Preloader

When you choose Default preloader, the **Preloader** window appears, allowing you to customize the loading indicator. All changes you make are immediately reflected in the preview window.



The loading indicator has six editable properties. To adjust their values, you can drag the handle on the slider, scroll the mouse wheel, or enter a new value.

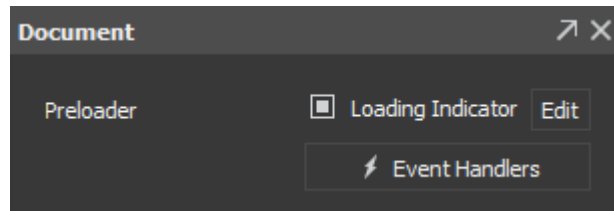
- **Diameter:** Set the diameter of the loading indicator.
- **Density:** Set the number of shapes drawn on the loading indicator.
- **Range:** Set the amount of the modified shapes in percentages. Specifically, it sets what range of the shapes should be scaled and/or faded. The shapes that are out of this range will be scaled and/or faded with a minimum amount only. If the range is 0.1, every shape out of the range will be scaled and/or faded to 10% of the original values. The visually acceptable range value should be between 0.4 and 1.5. The default value is 1.0.
- **Speed:** Set the speed of the loading indicator animation. This value tells the loading indicator how many shapes to skip by each tick.
- **Color:** Set the color of the loading indicator in RGB.
- **Shape:** Set the type of the loading indicator shapes, including oval, spiral, square, rectangle, and rounded rectangle.

Custom Preloader

When you choose Custom preloader, Saola Animate opens a preloader document where you can create a loading screen of your own. A custom preloader is a document, so you can

create animation as normal. You can also use **preloader events** to add interactivity. The custom preloader document will close if you close the main document or if you switch from Custom to Default preloader.

The loading indicator appears by default, but you can clear the **Loading Indicator** check box to turn it off. Of course, you can modify it as normal.







Chapter 19: Menus, Toolbars and Shortcuts



In Saola Animate, the menu bar contains a set of seven drop-down menus. These menus enable you to interact with the content of the application in many different ways. For example, the **File** menu provides commands for creating a new project, opening an existing one, saving, previewing, and exporting project to HTML5.

This chapter lists all the menus, toolbar buttons, and default shortcuts in Saola Animate. Note that you can **customize all shortcuts** for your preferences.

File Menu






The **File** menu controls major events such as creating a new project, opening an existing project, previewing scene and project in a browser, etc.








Command	Icon	Default Shortcut	Remark
New		CTRL+N	Create a new project.
Open Project		CTRL+O	Open an existing project.
Recent Projects			List of recently opened projects.
Save		CTRL+S	Save the current project.
Save As > Other Project			Save the project under a new name.
Save As > Package		CTRL+SHIFT+P	Save the project to package.
Close		CTRL+F4	Close the current project.
Preview Current Scene in Browser		CTRL+ALT+S	Preview the current scene in a default browser.









Command	Icon	Default Shortcut	Remark
Preview Project in Browser		CTRL+ALT+P	Preview the current project in a default browser.
Export to HTML5		CTRL+ALT+E	Export the current project to HTML5.
Exit			Close the application.

Edit Menu

The **Edit** menu lets you arrange elements on the Canvas. From here, you can also create new symbols.


Command	Icon	Default Shortcut	Remark
Undo		CTRL+Z	Revert the last actions. If you want to revert several actions at once, click the down arrow next to the button and select the action in the list that you want to undo.
Redo		CTRL+Y	Revert the last “undone” actions. If you want to redo several actions, click the down arrow next to the button and select the action in the list that you want to redo.
Cut		CTRL+X	Remove the selected objects and place them on the Clipboard.
Copy		CTRL+C	Copy the selected objects onto the Clipboard.
Paste		CTRL+V	Paste the contents of the Clipboard at the current location.

Command	Icon	Default Shortcut	Remark
Paste Here			Paste the copied elements at the position of the cursor.
Paste with Animation			Paste the copied elements with their existing animations.
Delete		DELETE	Delete the selected objects.
Select All		CTRL+A	Select all objects.
Group		CTRL+G	Group the selected elements together to create a new group element.
Ungroup		CTRL+SHIFT+G	Ungroup a group.
Convert to Symbol			Convert the selected elements to a symbol.
Size > Make Same Width			Match the width of all other elements with the width of the reference element.
Size > Make Same Height			Match the height of all other elements with the height of the reference element.
Size > Make Same Size			Match the width and height of all other elements with those of the reference element, respectively.
Alignment > Align Left		SHIFT+L	Align the left side of all other elements with the left of the reference element.
Alignment > Align Right		SHIFT+R	Align the right side of all other elements with the right of the reference element.
Alignment > Align Top		SHIFT+T	Align the top of all other elements with the top of the reference element.

Command	Icon	Default Shortcut	Remark
Alignment > Align Bottom		SHIFT+B	Align the bottom of all other elements with the bottom of the reference element.
Alignment > Align Center		SHIFT+E	Align the center of all other elements with the vertical center of the reference element.
Alignment > Align Middle		SHIFT+M	Align the center of all other elements with the horizontal center of the reference element.
Display Order > Bring to Front			Place the selected element at the very top of the stack.
Display Order > Bring Forward			Send the selected element higher by one level in the stack.
Display Order > Send to Back			Place the selected element at the very bottom of the stack.
Display Order > Send Backward			Send the selected element lower by one level in the stack.
Preferences			View and edit the global settings.

Scene Menu











The **Scene** menu helps you manage scenes.

Command	Icon	Default Shortcut	Remark
Blank Scene			Insert a blank scene.

Command	Icon	Default Shortcut	Remark
Duplicate Scene			Duplicate the selected scene to a new scene.









Element Menu




The **Element** menu lets you insert all types of elements into a project.





Command	Icon	Default Shortcut	Remark
Rectangle Div			Insert a rectangle div.
Rounded Rectangle Div			Insert a rounded rectangle div.
Ellipse Div			Insert an ellipse div.
Text			Insert a text.
HTML Widget			Insert an HTML widget.
Image			Insert an image.
Audio			Insert an audio.
Video			Insert a video.
Symbol			Insert a symbol.
Shape			Insert a shape. Click the down arrow to choose the shape you want.

Timeline Menu

The **Timeline** menu lets you control timelines and animations.






Command	Icon	Default Shortcut	Remark
Play/Stop		SPACE	Switch the control to play or stop the selected action on the timeline, starting from the current position of the Playhead.
Go to Start		HOME	Restore the Playhead to the beginning of the timeline.
Go to End		END	Bring the Playhead to the end of the animation on the timeline.
Loop Playback			The animation will be played endlessly (in a loop). When the Playhead reaches the end of the animation, it will return to the beginning and continue playing.
Auto-Keyframe Mode		K	Automatically generate keyframes when creating animations.
Add Keyframe			Create a property keyframe for the selected element.
Insert Label		ALT+L	Insert a label at the Playhead position.
Insert Trigger		ALT+T	Insert a trigger at the Playhead position.
Event Handlers			Open the Timeline Event Handlers.
Create Transition		CTRL+SHIFT+T	Create transitions for an animation segments that don't have transitions.
Remove Transition		SHIFT+DEL	Remove transitions from animation segments.

Command	Icon	Default Shortcut	Remark
Invert Animation Segment			Swap the starting and ending keyframes of an animation segment.
Show Animated Elements Only			Hide elements which don't have property animations.
Snapping		ALT+;	<p>Snap the Playhead, keyframes, labels, and triggers to each other or to particular tick marks on the timeline. Snapping helps easily line up keyframes and animation segments on the timeline.</p> <p>Three snap modes available: Grid, Playhead, and Keyframes, Labels, Triggers.</p>
Snap To > Grid			The Playhead and timeline items snap to the grid lines which only appear when you click the Show Grid button.
Snap To > Playhead			The keyframes, labels, triggers snap to the Playhead.
Snap To > Keyframes, Labels, Triggers			The Playhead and timeline items snap to the position of timeline items.
Show Grid			Show grid lines on the timeline. Grid lines give precise measurements of the timeline by displaying subtle vertical lines at regular intervals.
Grid			Use the Grid submenu to set the time increments, such as half-second, quarter second, 24 frames per second, etc.

Command	Icon	Default Shortcut	Remark
Zoom Out		-	Reduce the time scale of the timeline.
Zoom In		=	Expand the time scale to see more details.
Zoom to Fit		\	Fit the existing animation to the timeline view.
Expand/Collapse All		CTRL+SHIFT+.	Expand or collapse all animated properties of elements.

View Menu

The **View** menu gives you access to customize the workspace.

Command	Icon	Default Shortcut	Remark
Language			Choose the language of the user interface.
Rulers			Show/hide the rulers on the Canvas.
Canvas Snapping			Turn Canvas Snapping on/off.
Zoom > Zoom In			Expand the Canvas scale.
Zoom > Zoom Out			Reduce the Canvas scale.
Zoom > Zoom Level			Set the zoom level of the Canvas.
Zoom > Fit on Screen			Set the scene content to fit into the viewport size.
Toolbars > File			Show/hide the File toolbar.

Command	Icon	Default Shortcut	Remark
Toolbars > Edit			Show/hide the Edit toolbar.
Toolbars > Insert			Show/hide the Insert toolbar.
Toolbars > Modify			Show/hide the Modify toolbar.
Properties			Show/hide the Properties pane.
Scenes			Show/hide the Scenes pane.
Document			Show/hide the Document pane.
Timeline			Show/hide the Timeline pane.
Elements			Show/hide the Elements pane.
Resources			Show/hide the Resources pane.
Functions			Show/hide the Functions pane.
Breakpoints			Show/hide the Breakpoints pane.

Help Menu




The **Help** menu provides information about the current version, the documentation, user community and helps check for updates.















Command	Icon	Default Shortcut	Remark
About			Provide information about the current version.
Help Contents		F1	Launch the Help file.





Command	Icon	Default Shortcut	Remark
User Community			Launch the default browser to access the user forum/FAQ.
Check Updates...			By default, Saola Animate periodically checks for updates when it starts. Disable this feature by clicking Edit > Preferences > General > Automatic Updates .
My Account			Open your account webpage where you can update your profile, manage your subscriptions, etc.
Sign Out			Sign out of your account.

Floating Text Format Toolbar








The **Floating Text Format** toolbar appears when you edit text inline. It provides necessary commands for formatting text.



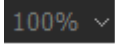

Command	Icon	Default Shortcut	Remark
Font Name			Change the font of the selected text.
Add fonts			Add Google fonts to a document.
Font Size			Change the size of the selected text.
Align Left		CTRL+L	Align the selected text left.
Center		CTRL+E	Center the selected text.
Align Right		CTRL+R	Align the selected text right.

Command	Icon	Default Shortcut	Remark
Justify		CTRL+J	Justify the selected text.
Bold		CTRL+B	Make the selected text bold.
Italic		CTRL+I	Make the selected text italic.
Underline		CTRL+U	Underline the selected text.
Strikethrough			Strike the selected text through.
Superscript		CTRL+SHIFT+=	Format the selected text as superscript.
Subscript		CTRL+=	Format the selected text as subscript.
Decrease Indent			Move the selected text closer to the margin.
Increase Indent			Move the selected text father away from the margin.
Bullets			Create a bulleted list.
Numbering			Create a numbered list.
Text Color			Change the text color of the selected text.
Highlight Color			Highlight the selected text.
Insert Special Character			Insert a special character to the text.

Command	Icon	Default Shortcut	Remark
Insert Link			Insert a hyperlink to text.
Remove Format			Return the selected text to its default formatting.
Edit HTML			Edit the HTML of the text.
Floating Text Format Toolbar			Show/hide the Floating Text Format toolbar when editing text.

Shortcuts For Script Editor

Command	Icon	Default Shortcut	Remark
Undo		CTRL+Z	Revert the last actions.
Redo		CTRL+Y	Revert the last “undone” actions.
Copy		CTRL+C	Copy the selected items onto the Clipboard.
Cut		CTRL+X	Remove the selected items and place them on the Clipboard.
Paste		CTRL+V	Paste the contents of the Clipboard at the current location.
Delete		DELETE	Delete the selected items.
Export to File			Export the current function to file.

Command	Icon	Default Shortcut	Remark
Search		CTRL+F	Display the Find search box where you can find specific text.
Replace		CTRL+H	Display the Find search box where you can replace specific text.
Zoom In/Out		CTRL+Mouse Wheel Up/Down	Expand/reduce the text size.
Options			Show/hide editor features and access the Preferences dialog > Code Editor tab .

Chapter 20: Customizing Saola Animate

You can customize many aspects of Saola Animate to suit your particular needs.

Changing Language

By default, the display language is set to English (United States). If you prefer another language, click **View > Language** and select another language. The change will be applied the next time you open the application.

Another way to customize the display language is use the **Preferences** dialog. Do the following steps:

1. Click **Edit > Preferences** to open the **Preferences** dialog.
2. In the **General** tab, select the appropriate language in the **Language** drop-down list.
3. Click **Apply** to submit changes.

Customizing User Interface

Showing and Hiding Panes and Toolbars

Click the **View** menu and do one of the following:


- Show or hide four toolbars on the main toolbar, including File, Edit, Insert, and Modify.
- Show or hide many panes, including Properties, Scenes, Document, Timeline, etc.

Alternatively, right-click the main toolbar to toggle toolbars and panes. Active items have a check mark next to their labels. Also, you can click the **Close** button in the top right corner of a pane to close it.

Saola Animate saves the project settings when you close a project. The next time you open the application, the settings of the last saved project will be automatically applied to the new one.

Changing Pane Location

Drag panes except the **Timeline** pane from their current location by dragging their title bar.

Alternatively, click the **Float**  button in the top right corner of a pane to undock it. After undocking a pane, you have three available options:

- Make the pane free-floating anywhere on the screen, even outside the application window.
- Dock the pane in a highlighted drop zone so that it stands together with other panes.
- Add the pane to a tabbed group which contains several panes sharing the same drop zone. To change the tab order, drag the pane left or right by its tab. To remove the pane from the group and make it free-floating, drag it outside the group by its title bar or its **Float** button.

Changing Pane Size

Changing pane size is useful when you want to make your work area larger or want to view more of the information that appears in a pane.

- To resize a docked pane, move the pointer over the left or right edge of the pane (for vertical panes) or over the top edge of the pane (for horizontal panes). When the pointer turns into a resize pointer, drag the boundary to resize the pane to the dimensions you want.
- To resize a floating pane, hover over any edge or corner of the pane until the pointer turns into a resize pointer. Then, drag the boundary until the pane is the width/height you want.

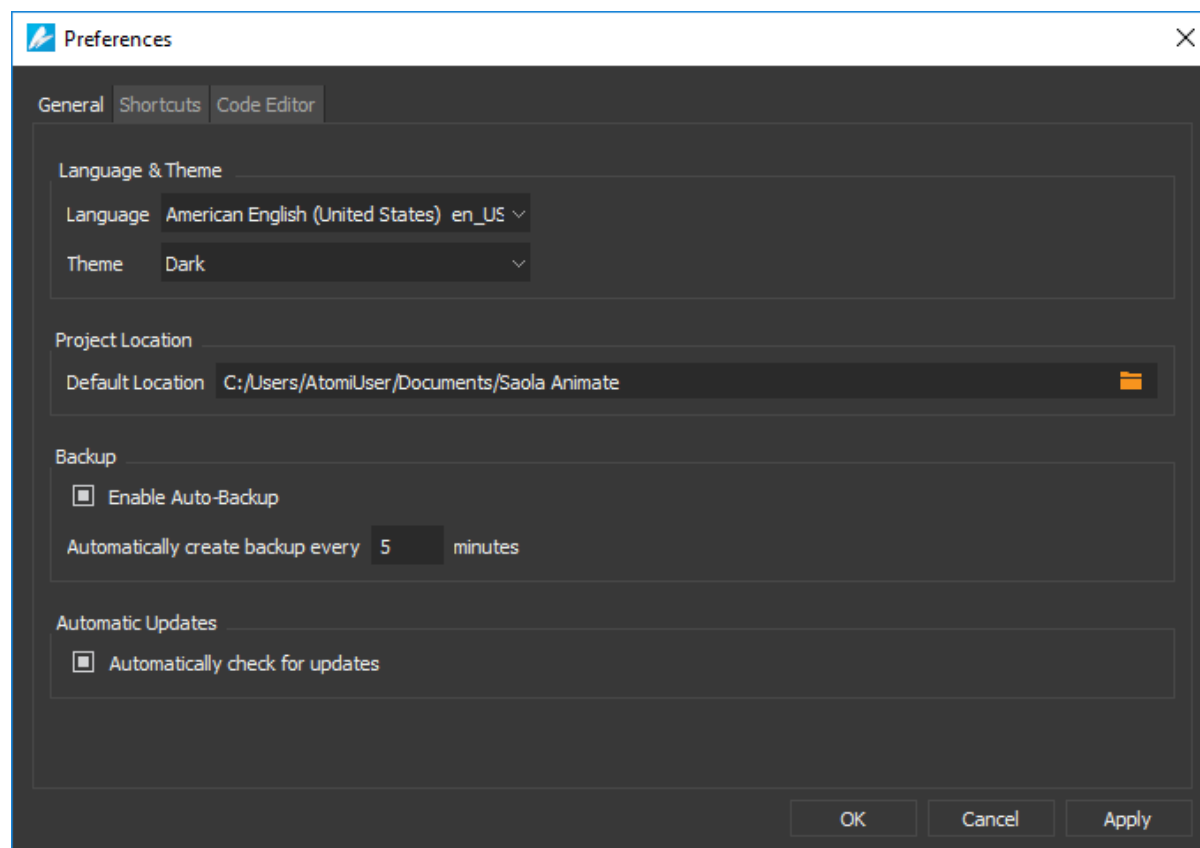
Customizing Preferences

Use the **Preferences** dialog to set how you want Saola Animate to operate. To open the dialog, select **Edit > Preferences** from the menu bar. The dialog has three tabs, each tab serves a different category.

Tab	Category
General	General options applicable to the entire application.
Shortcuts	Shortcuts used for various operations.
Code Editor	Options to customize the JavaScript Function editor.

General Tab

The **General** tab is the first tab in the **Preferences** dialog where you can set a few preferences for how Saola Animate works.




Language and Theme

By default, the display language is set to English (United States). If you want to use another language, select a new one from the **Language** drop-down list. The language change will be applied the next time the application starts. You can also **change the language** by clicking **View > Language** and choose your desired language from the list.

Regarding theme, Saola Animate currently offers the Dark theme only. Other theme options will be available in the next versions.

Project Location

By default, all new projects are stored in the Documents folder. You are allowed to change the default project location by entering a new location in the **Default Location** box. You can navigate to a location using the **Browse**  button.

No matter the default location, you can always store your project in a different place. See **Creating Projects** for more details.

Backup

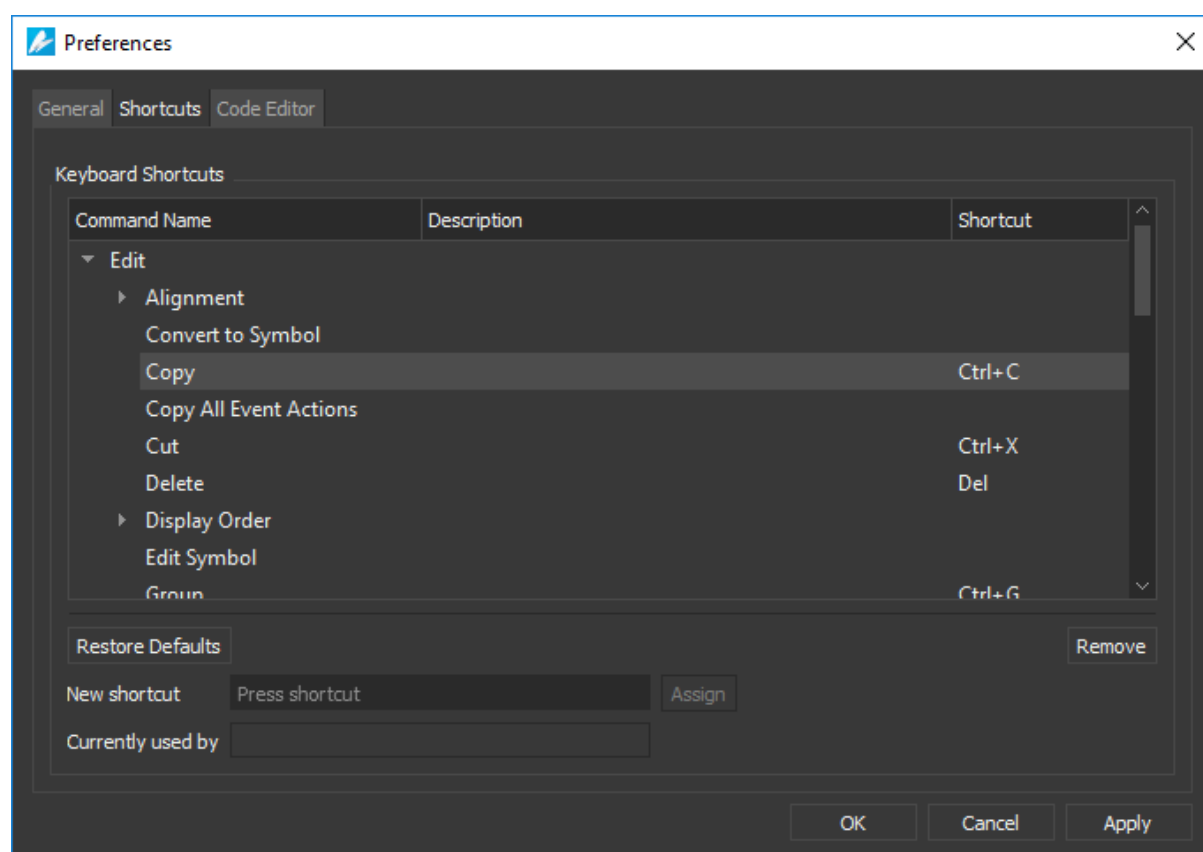
Auto-Backup allows you to recover your work if Saola Animate crashes and you haven't saved your work. By default, Auto-Backup is made available and set to automatically save a backup of your work every five minutes. You can shorten or lengthen this interval by entering a new interval (in minutes) in the box.

Automatic Updates

Saola Animate periodically checks for updates when it starts. This is selected by default.

Shortcuts Tab

The **Shortcuts** tab shows all keyboard shortcuts in Saola Animate. You can assign keyboard shortcuts to commands or remove any keyboard shortcuts you don't want.



All major commands are grouped functionally in a hierarchical tree. The tree consists of nodes, each of which contains a group of related commands. Show the commands by clicking the triangle on the left of each node. The **Shortcut** column shows any keyboard shortcuts that are currently assigned to a command or item.

Assigning Shortcuts

1. Select a command or item that you want to assign a shortcut.
2. Click inside the **New shortcut** box and press the combination of keys you want to assign.
3. Notice the **Currently used by** box to see if the combination of keys is already assigned to a command or item. If it is in use, press a different combination.
4. Click **Assign** to assign the new shortcut.
5. Click **Apply** to submit changes.

For example, select the **Copy** command in the tree view, then press **CTRL+A** in the **New shortcut** box. The combination is automatically captured and displayed. You are informed that the combination is currently assigned to the **Select All** command.

Now you have two options: Either try another shortcut (and repeat the cycle), or press **Assign** to reassign the **CTRL+A** shortcut to the **Copy** command. If you choose the second option, this shortcut will no longer be assigned to the **Select All** command.

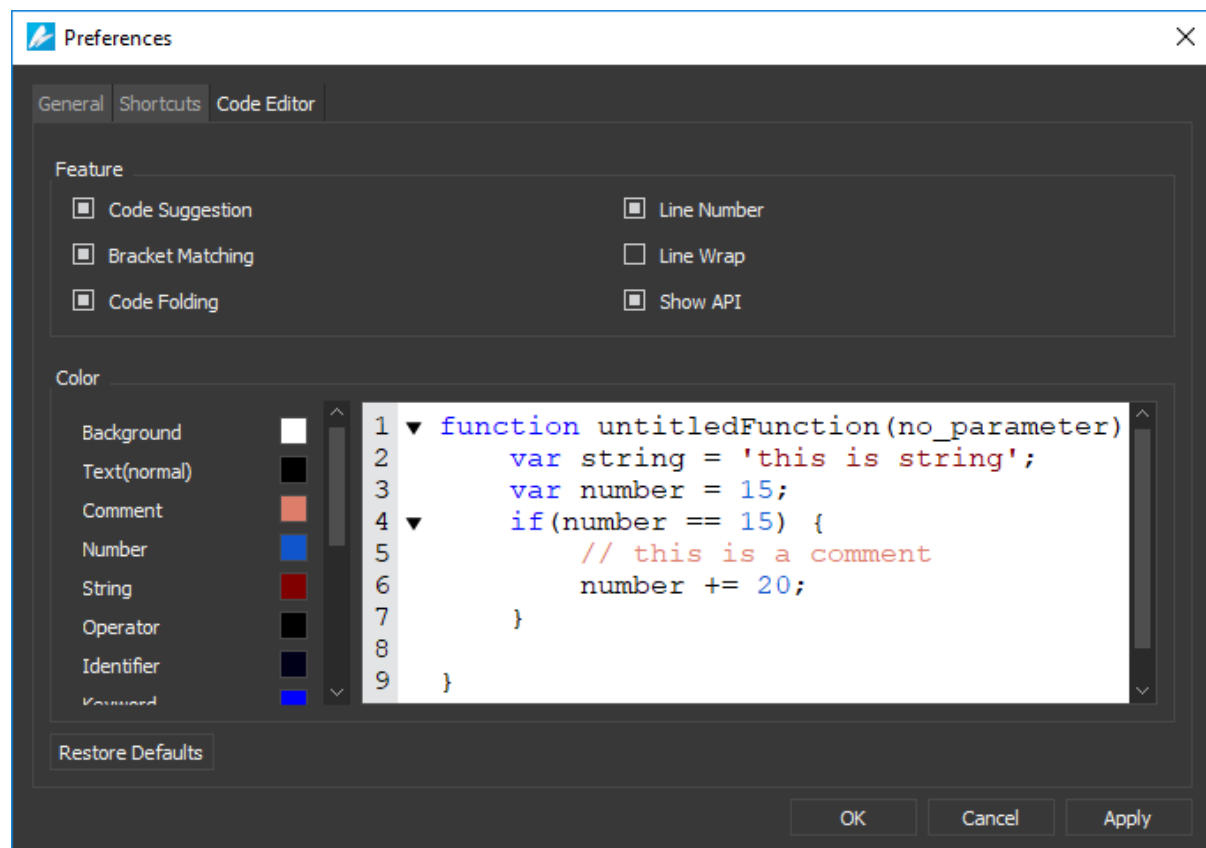
Removing Shortcuts

1. Select a command or item that you want to remove its shortcut.
2. Click **Remove**.
3. Click **Apply** to submit changes.

You can go back to the original settings at any time by clicking **Restore Defaults**.

Code Editor Tab

Saola Animate offers a **feature-rich editor** for script writing and editing. Switch to the **Code Editor** tab to customize coding preferences. If your experiments go haywire, you can always restore to the defaults by clicking **Restore Defaults**.



General Settings

- **Code Suggestion:** Provide suggestions while you type into the editing area.
- **Bracket Matching:** Highlight matching sets of brackets.
- **Code Folding:** Allow collapsing or folding the display of a block of code.
- **Line Number:** Display line numbers along the side of the code.
- **Line Wrap:** Wrap the code so that each line fits in the viewable window.
- **Show API:** Show a list of APIs in the pane below the editing area.

Syntax Highlighting

Customize syntax highlighting preferences by clicking the color rectangle next to each element and changing the color. All changes are immediately reflected in the preview editor on the right.