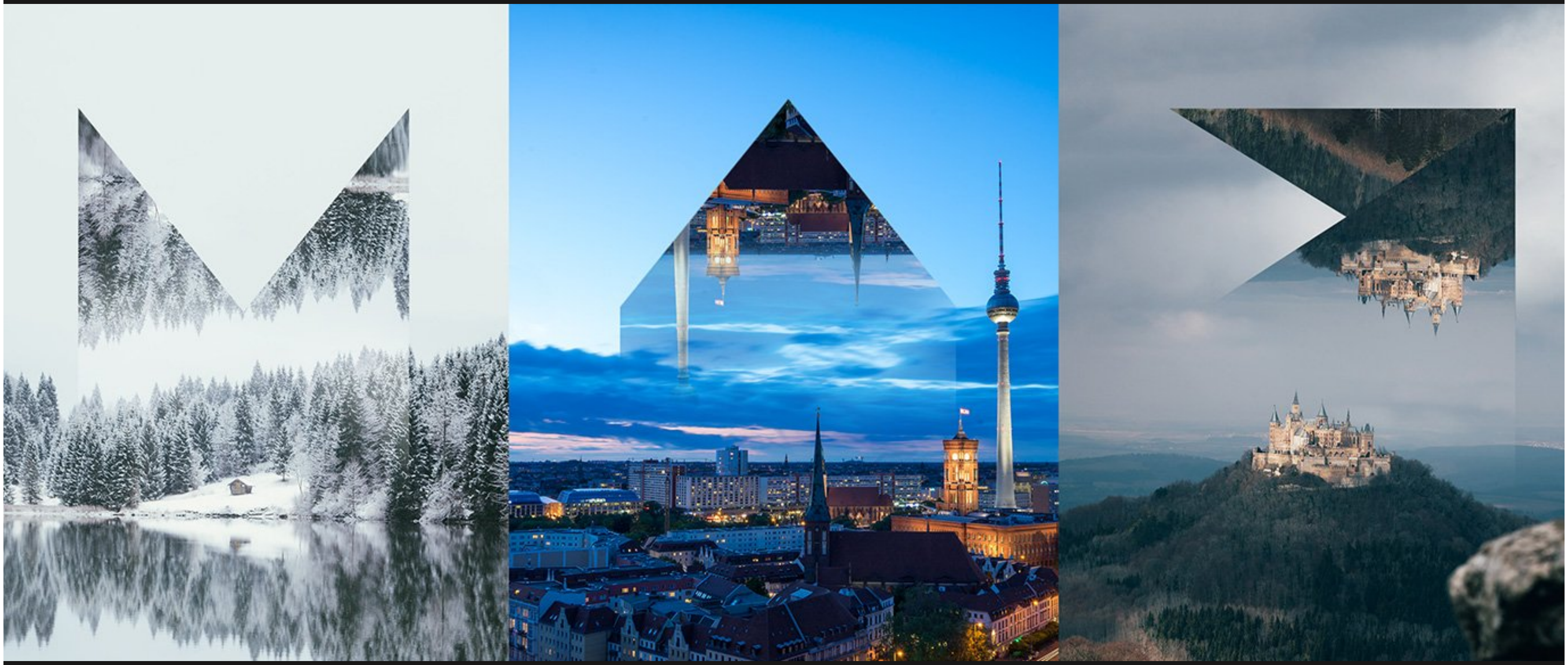
Adobe Animate CC:

Introduction to Animation and Interactivity

Adobe MAX 2016 | Session L1600 | Joseph Labrecque



Contents

Abstract

Adobe Animate CC is a professional tool used to create a diverse variety of project types including rich animations, compelling interactives, award-winning games, popular mobile apps, and more. Learn how to get started creating your own engaging animated and interactive content across multiple platforms using Animate CC in this lab.

Joseph Labrecque, interactive content designer and developer, will guide you through the following topics:

- Capabilities of Animate CC to create a multitude of project types targeting a variety of platforms
- Latest new features in Animate CC that you can use to really push the creative envelope
- Time-tested tips and techniques that let you work fast and produce compelling content for your projects

Objectives

Participants should come away from the lab with the following:

- A solid idea of how to use Animate CC in their creative work
- The ability to create assets within Animate CC.
- How to animate content using a variety of techniques.
- How to add interactive elements to animated content.
- An understanding of a variety of target platforms now available.

About the Instructor

Joseph Labrecque

Senior Interactive Software Engineer - Artist - Author – Absintheur
Adobe Education Leader - Adobe Community Professional

Joseph Labrecque is primarily employed by the University of Denver as a senior interactive software engineer specializing in the creation of expressive desktop, web, and mobile solutions. His work incorporates a strong focus on the Adobe Flash Platform alongside more general web standards initiatives involving the use of HTML5, CSS, JavaScript, and related technologies.

He is also the proprietor of Fractured Vision Media, LLC; a digital media production company, technical consultancy, and distribution vehicle for a variety of creative works. Joseph authors video courses and written works through organizations which include Lynda.com, LinkedIn Learning, Peachpit Press, Train Simple, Pluralsight, Apress, Packt Publishing, O'Reilly Media, Brainbuffet, and video2brain.

Joseph is an Adobe Education Leader and Adobe Community Professional.

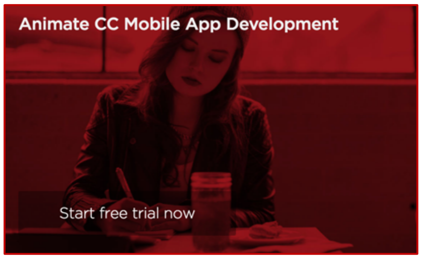
Recent Publications

If interested in learning more about using Animate CC to create content targeting Flash Player, Adobe AIR, HTML5 Canvas, HD Video, WebGL, Apple iOS, and Google Android… have a look at the following publications!

Lynda.com | LinkedIn Learning - Animate CC 2017: New Features

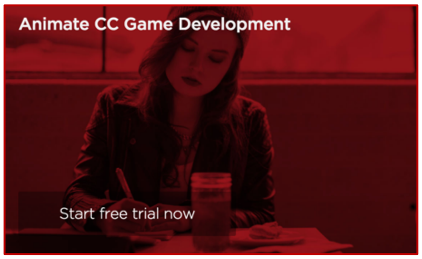
The new Adobe Animate is here and better than ever. Join Joseph Labrecque for a tour of the new features in Adobe Animate CC 2017, the keyframe-based animation software formerly known as Flash Professional. Learn about the new Camera tool for more expressive storytelling, vector brush creation and management, HTML5 components for interface design and video playback, publishing enhancements such as a completely rewritten animated GIF generator, and smaller but no less impressive workflow enhancements like the ability to share symbols and entire animations through CC Libraries.

Pluralsight - Animate CC Mobile App Development



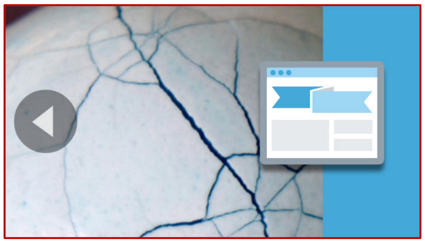
In this course, you will see how to take advantage of mobile device hardware using new API's in ActionScript 3.0.

Pluralsight - Animate CC Game Development



In this course, you are going to learn how to design and develop web-based games for HTML5 Canvas using Adobe Animate CC.

Lynda.com | LinkedIn Learning - Learn Adobe Animate CC: Data-Driven Animation



Learn to create data-driven animations, such as a data-driven image carousel and an interactive RSS feed, using Adobe Animate CC—the newly relaunched version of Flash Professional. Joseph Labrecque takes you through both projects step-by-step, providing easy-to-follow instructions for interacting with JSON and XML data sets and using jQuery to ingest and parse data in HTML5 Canvas-based projects.

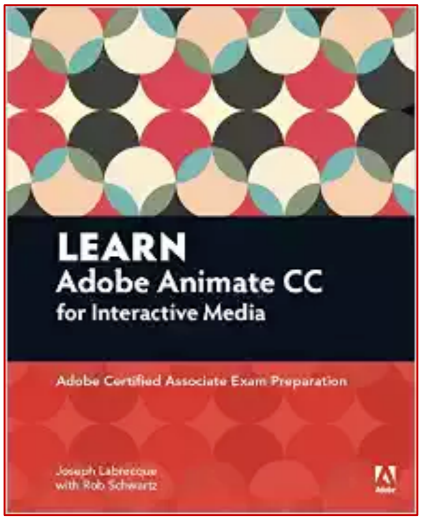
Lynda.com | LinkedIn Learning - Learn Adobe Animate CC: The Basics

Get started with Adobe Animate CC. Create rich graphics and interactive animations. Learn how to import assets, create vector objects, work with text and images, use tweens, manage your timeline, and publish content to various formats to remain compatible with new and emerging web standards, including HTML5.

Lynda.com | LinkedIn Learning - Adobe Animate CC: New Features

Learn all about the new features and enhancements in the June 2016 release of Adobe's new animation platform, Animate CC.

Adobe Press - Learn Adobe Animate CC for Interactive Media



Knowing the industry-standard animation and interactivity tool Adobe Animate CC can help you get a foothold in the exciting web design and mobile app-development world. This study guide uses more than 6 hours of video integrated with text to help you gain real-world skills that will get you started in your career designing and building interactive media using Adobe Animate CC.

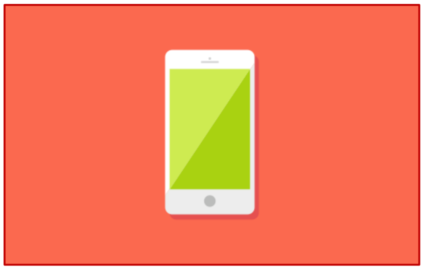
Train Simple - Animate CC Game Development

In this course you are going to learn how to design and develop web-based games for HTML5 Canvas using Adobe Animate CC. We are going to look at the entire game development process, including building the game world and environment, devising core game mechanics, creating a multi-state, interactive player sprite, along with a set of obstacles and rewards.

Lynda.com | LinkedIn Learning - Adobe Animate CC: First Look

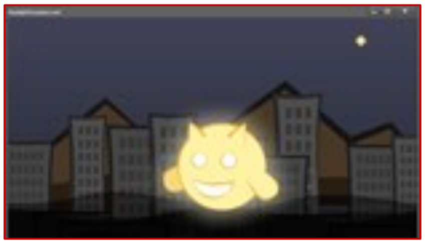
Get your first look at the new era in Adobe animation: Adobe Animate. Learn about the new tools and improved interface in Animate CC.

TrainSimple - Animate CC Mobile App Development



This course is designed for experienced Flash and ActionScript designers and developers. Joseph Labrecque will show you how to take advantage of mobile device hardware using new API's in ActionScript 3.0. You'll see how you can produce apps and publish them for mobile devices.

Lynda.com | LinkedIn Learning - HTML5 Canvas and WebGL in Flash Professional CC



Create a shared pool of content for a set of related projects—an HTML5 video, a WebGL-rendered interactive ad, and an HTML5 Canvas game—using Flash Professional CC.

More Publications…

Can be found at the following addresses:

Books: http://josephlabrecque.com/books/

Videos: http://josephlabrecque.com/books/video.html

Adobe Animate CC (formally Flash Professional)

Flash Professional had a strong showing at Adobe MAX last year. If you were able to attend those sessions, or any of the subsequent presentations I have given around the product, you will know that it has received a lot more resources, leading to many more enhancements over the past year than many years previous.

Something is very clear: Adobe cares very deeply about Flash Professional once again and wishes to restore it as the premiere animation and interaction tool – but now as a platform agnostic creativity application not restricted to any single platform.

As a point of clarity – to reinforce this renewed focus – Adobe announced the decision to rebrand Flash Professional CC as Animate CC in late 2015.

> *To more accurately represent its position as the premier animation tool for the web and beyond, Flash Professional will be renamed Adobe Animate CC, starting with the next release in early 2016.*

What does all this mean for Flash Player and AIR? Nothing has changed in terms of what can be accomplished with these targets in regard to previous versions of the application. In fact… all of the enhanced drawing tools and additional features that are currently available can absolutely be used when targeting ActionScript-based .fla documents.

> *Animate CC will continue supporting Flash (SWF) and AIR formats as first-class citizens. In addition, it can output animations to virtually any format (including SVG), through its extensible architecture.*

All platforms benefit from new tooling and enhancements present in Animate CC!

The message that is most important in regard to this subject is that Flash Player and AIR remain two targets among many. The Flash Runtimes are absolutely supported within Animate CC and will continue to be supported in the same way. In fact, the new SWF Archive format relies upon them!

These runtimes are very important to a lot of people and Adobe will not abandon them. I have a few more words on the subject of Flash Player and AIR below, suffice to say that these targets exist within Animate CC and remain a first class target platform.

On a personal note… I feel really quite validated by this change. I have been pressing the fact that Flash Professional, now Animate CC, is one of the most creative tools out there… with lengthy emails documenting this struggle over the past few years.

The variety of different project types available to us as creative people through this single application is astounding. Many of us have been holding high the banners for Flash Professional even when much of Adobe has refused to even notice them laying trampled and broken in the mud. Our work around these technologies has been marginalized and diminished – yet many have not given up. There have been handfuls of supporters who have fought alongside us in this effort – some within Adobe but most on the outskirts just building cool stuff every day despite what we've had to deal with.

Today… as clearly evident at Adobe MAX 2016… things have changed.

Drastically.

Flash Professional, as Adobe Animate CC, is now poised to regain its proper place aside such giants as Illustrator and Photoshop as a top tier application in the Creative Cloud portfolio.

No more messing around… let's create some amazing experiences regardless of platform! Choice in technology is a grand thing – and POWERFUL.

New in Adobe Animate CC

The various releases of Animate CC this past year have been the biggest in quite some time! There are so many new tools and features for drawing, animating, export, workflow, and more.

A list of most of these features is provided for you, here.

Adobe Animate CC [16.0 November 2016]

Virtual Camera

HTML5 Canvas Video Component

HTML5 Canvas UI Component Set

Code Snippets for Components

Export Image

Export Animated GIF

Publish SWF Archive

Brush Creation

Brush Management

Brush Tilt and Pressure

Stage Clipping and Outline

Share Documents, Symbols, and Brushes

Adobe Animate CC [15.2.1 August 2016]

Google Fonts for HTML Canvas

Expanded Brush Sizes

Publish Profile Format

Timeline, Onion Skin, and Key Frame Enhancements

Adobe Animate CC [15.2 June 2016]

Graphic Symbol Frame Picker

Vector Pattern Brushes

User-Defined Onion Skin Colors

Transparent and Responsive OAM

Enhanced Brush Editor

Layer Transparency

Bitmap Snapping

Responsive / Centered Canvas

Merge JavaScript in HTML

HiDPI and Retina Publishing

Updated Libraries Panel

AS3/JS Pin Script

Canvas Preloaders

Transparent Canvas Background

Advanced Ps and Ai Import

Latest Flash Player and AIR

Windows Performance

Adobe Animate CC [15.1 February 2016]

Vector Art Brushes

Integrated Brush Library

Creative Cloud Libraries

Tagged Color Swatches

SVG Import

Improved Brushes

4K+ Video Export

Colored Onion Skinning

HTML5 Canvas Templates

OAM Publishing

Projector Publishing

Stage Rotation Tool

Typekit Integration

Timeline Audio Splitting

Whole-Stage Scaling

Adobe Stock Integration

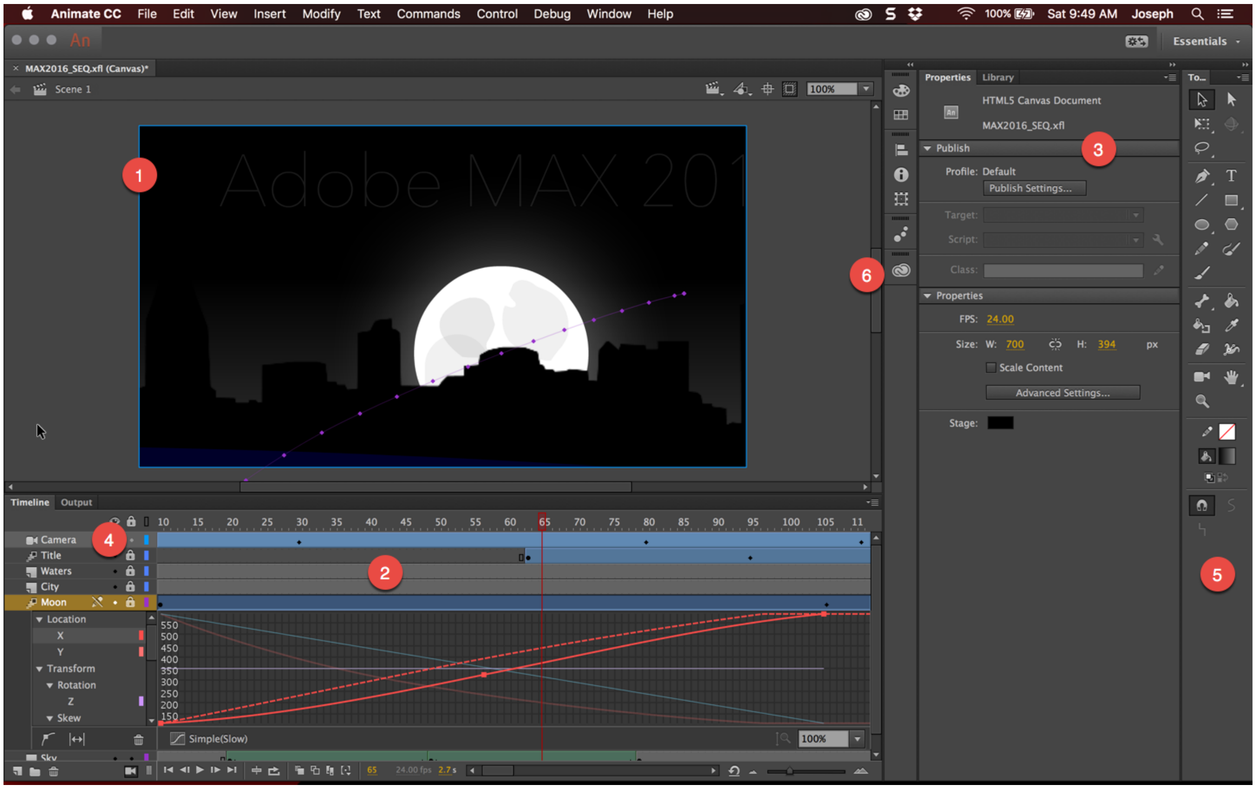
Enhanced Drawing Objects

Getting Acquainted with the Workspace

The first thing we will do is get a good look at the various tools and panels available to us within Animate.

Some of these may be familiar for those who have worked with Animate in the past, or even other creative tools like Photoshop or Illustrator. Other tools may be brand new – such as the new Camera tool.

Major workspace areas are highlighted below.



HTML5 Canvas Document
MAX2016_SEQ.xfl
Profile: Default
FPS: 24.00
Size: W: 700 H: 394 px
☐ Scale Content
Stage:

1: The Stage and surrounding Pasteboard

Anything within the Stage is visible within your project. The Pasteboard is the surrounding area, which is off-stage and not normally visible. Animate CC has a new stage clipping feature which hides anything on the pasteboard.

2: The Timeline

The Animate Timeline is where you manage Layer order and animate content across Frames. It is also where the powerful Motion Editor can be accessed.

3: The Library and Properties Panel

The Properties panel will display the properties of any selected object for manipulation. When no objects are selected, it will display properties for the document itself.

The Animate Library is where any of your symbols and imported assets are stored. Vector shapes which have not been made into Symbols will not appear within the Library – only existing on the Stage.

4: The Camera

Brand new to Animate CC is a Camera which can be used to zoom, pan, rotate, and even add tint and color effects to your content. It's an amazing new feature which makes animating the entire stage a joyful experience in creative expression!

5: The Tools Panel

This is the spot from where all out our creative tooling can be accessed! Brushes, shapes and primitives, the Pen Tool, the new Camera tool, and more!

6: Creative Cloud Libraries

Animate CC includes integration with Creative Cloud Libraries! Access all of your generated content including shapes, colors, and brushes for use in your Animate projects.

A new feature of this panel is that you can share symbols and documents to your CC Libraries and other users!

SEQUENCE 0: Creating a New Document

With any project in Animate CC, the first thing we have to do is make some decisions around which platform we want to target, create a new document targeting that platform, and then configure certain aspects of the project before saving the document.

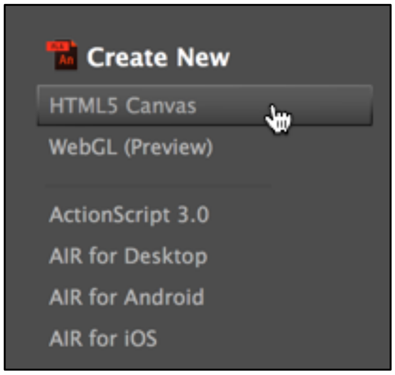
Creating a New Project

The first thing we must do is create a new project in Animate. No matter what our target platform is, the document type will be an Animate CC authoring file of type .FLA (or .XFL, if you want to use an uncompressed file type).

Targeting Flash Player? FLA. Targeting HTML5 Canvas? FLA. Targeting WebGL, Apple iOS, or Google Android?

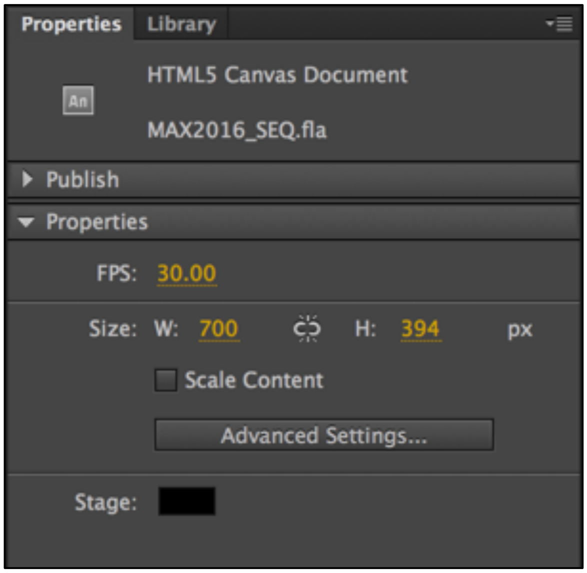
FLA. FLA. FLA.

Why is this important? Because you are able to copy assets and animations across all of these different publish targets – since they all use the exact same authoring file type!



We will be working with a HTML5 Canvas document for this entire project. To create a new document targeting HTML5 Canvas, choose File > New from the application menu and then click on HTML5 Canvas.

Alternatively, you can choose HTML5 Canvas from the Create New column in the Welcome Screen.



HTML5 Canvas Document

MAX2016_SEQ.fla

FPS: 30.00

Size: W: 700 H: 394 px

□ Scale Content

Stage:

With a new document created, we can now configure properties such as the stage width and height, background color, frames-per-second, and more. With no other assets selected upon the document stage, the properties panel will reflect properties pertaining to the document itself.

Set the following properties for this document:

- FPS: 30
- Width: 700
- Height: 394
- Background Color: Black / #000000

About HTML5 Canvas and CreateJS

Canvas is a new element in HTML5, which provides APIs that allow you to dynamically generate and render graphics, charts, images, and animation. The presence of the Canvas API for HTML5, strengthens the HTML5 platform by providing two-dimensional drawing capabilities. You can use the traditional Animate timeline, workspace, and tools to create content, but produce HTML5 output.

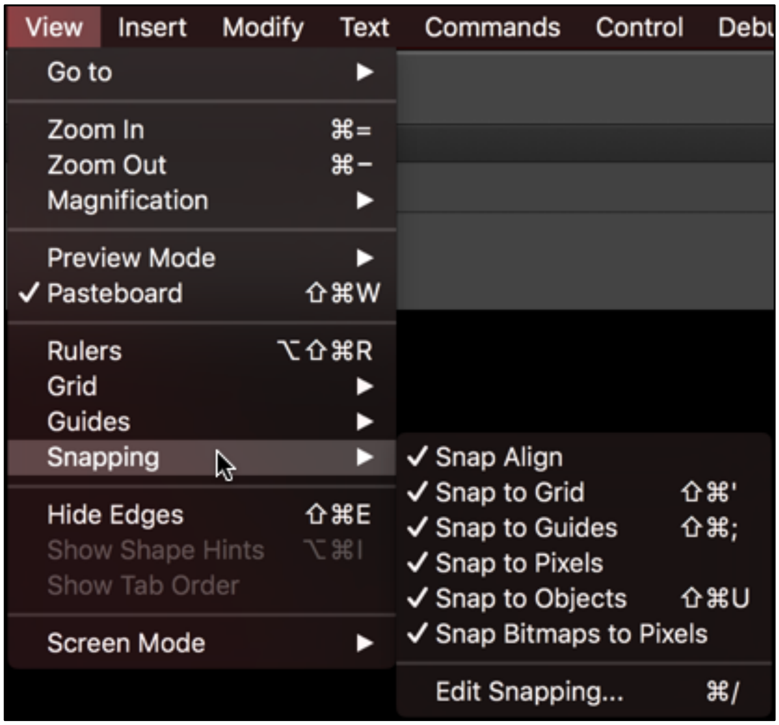
Animate CC is integrated with CreateJS, which enables rich interactive content on open web technologies via HTML5. Animate CC generates HTML and JavaScript for content (includes bitmaps, vectors, shapes, sounds, tweens, and so on) created on stage. The output can be run on any device or browser that supports HTML5 Canvas.

For an overview of CreateJS, visit http://www.createjs.com/

Guides, Rulers, and Snapping

Something advisable when setting up a new project in Animate is to configure the snap settings, rulers, and guides.

The first thing to do is set the snap settings of your project to be as tight or forgiving as is required. I find that the stricter settings can really help when drawing out assets across the stage.



Access these setting via View > Snapping from the application menu. As you can see above, there are quite a number of options to toggle!

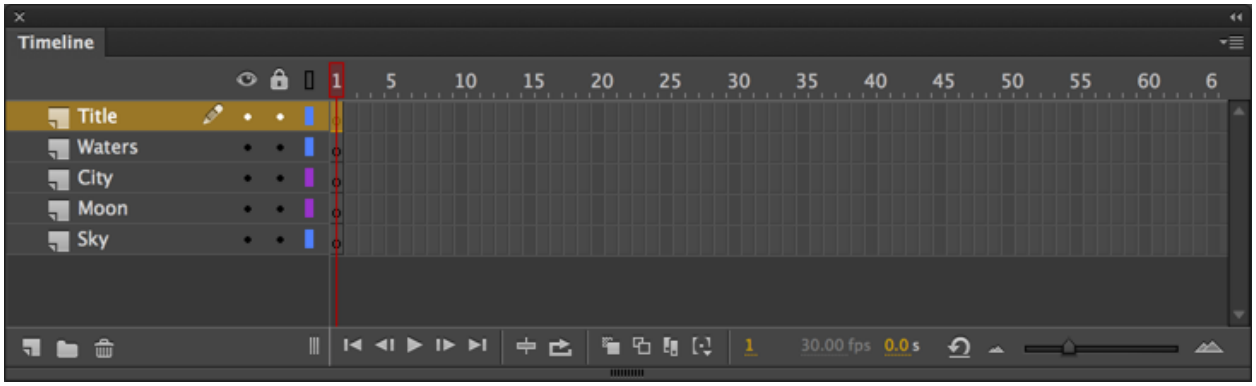
Additionally, if you want to use guides for your project, you will want to enable rulers via View > Rulers.

- Guides themselves can be pulled from the rulers and overlaid across the stage.
- Guides can assist with measurement and visualization.
- Guides are great for general user interface layout.

Layers and the Timeline

The last thing to address before moving on is the creation and proper naming of layers upon the timeline.

Generally, you'll always want to have any asset you are animating on its own layer. Even with objects which are not being animated – they often benefit from the organizational structures and labeling provided by a well-organized timeline.



Use the New Layer icon at the bottom-left of the timeline to insert 4 additional layers – making 5 total layers for now. You'll next want to double-click each layer name to rename the layers to something more descriptive.

Beginning from the top-most layer, proceed in naming your layers as follows:

- Title
- Waters
- City
- Moon
- Sky

Saving your Project

Now, be sure and save your project before moving on to the next sequence. You can name it whatever you like… though I recommend something like "MAX2016_SEQ0.fla" in order to have your files conform with each project sequence.

SEQUENCE 1: Building Creative Assets

Adobe MAX 2016

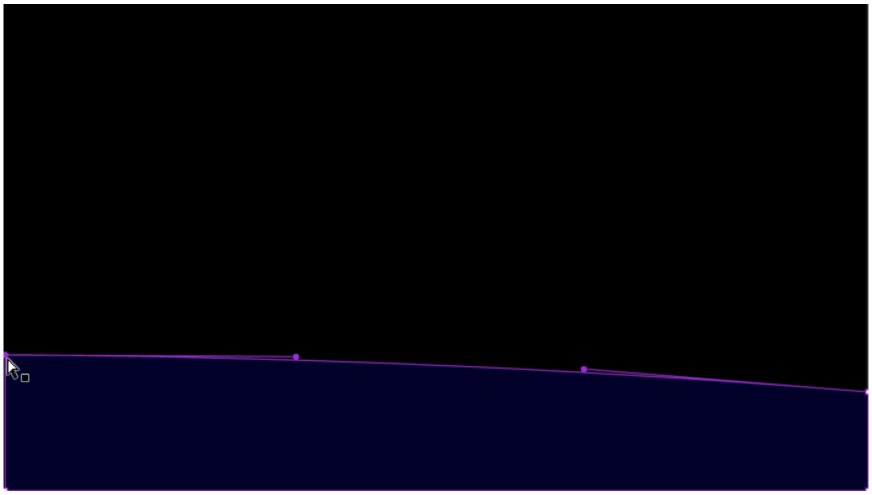
While some animation tools are primarily built to animate imported assets created in other applications, Animate CC contains a full set of creative tooling within the application itself as well. In this sequence, we'll use these tools to draw a number of different assets for our project.

Drawing the Waters

We will now use a number of the shape and selection tools available in Animate to design some waters low against the nighttime sky.

- Select the Waters layer and choose the **Rectangle** tool. Create a rectangle along the bottom portion of the stage with no stroke and the fill color of #000029 – a dark blue.

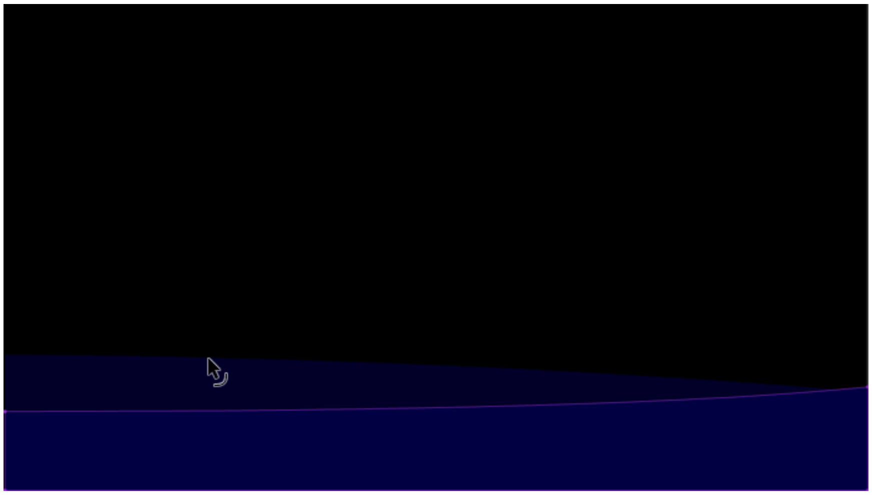
- Now, choose the **Subselection** tool and modify the various anchor points until the shape appears similar to what you see in the figure below. Of course, you can be more playful with this, if desired! That's the water which will appear farthest away from the camera.



- For the closer water, select the Waters layer and create a similar shape – this time with a lighter blue color - #000042. Again, no stroke is needed unless you desire.

- Now, choose the **Subselection** tool again and modify the various anchor points until the shape appears as below. We now have both the waters aligned just as we like.

- You can also select the **Selection** tool and hover over an unselected object edge until the cursor displays an arc. You can then use the cursor to pull and push to modify a shape's appearance. Try it out!



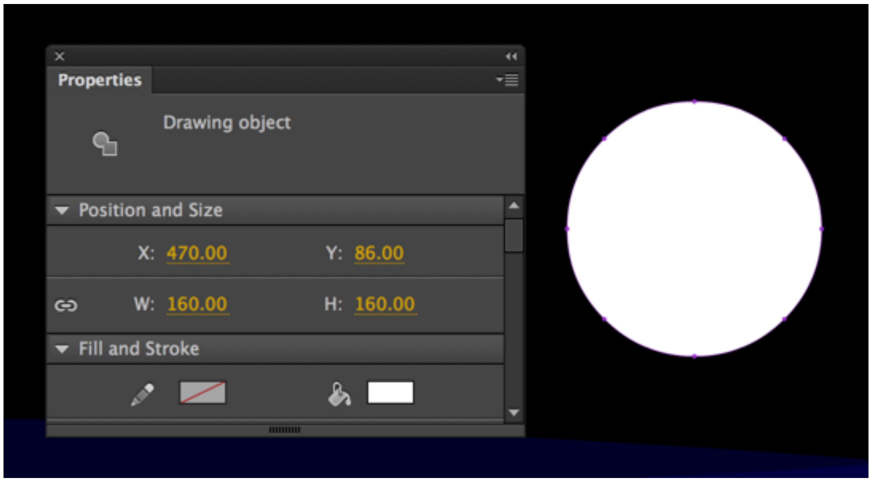
The sky is pretty empty right now. Let's fix that.

Drawing the Moon

Since Adobe MAX 2016 is taking place so close following Hallowe'en, let's go ahead and create a spooky moon that can be animated across the stage.

- Select the Moon layer within the Timeline. You may want to lock all other layers to be sure you are drawing in the correct layer.
- Using the Oval tool, choose pure white for the fill and disable the stroke by choosing the swatch with the red line through it.
- Now draw out a nice circular moon that measures about 160x160 in height and width. Holding down the SHIFT key while you draw will allow you to

create a perfect circle in place of an oval.



Properties

Drawing object

Position and Size

X: 470.00 Y: 86.00

W: 160.00 H: 160.00

Fill and Stroke

- You may want to also create this as a Drawing Object. If you didn't create it with Drawing Object mode enabled, there is a small icon that lets you convert a shape to a drawing object within the Properties panel.

Adding Features to the Moon

Anyone who has looked up into the night sky and seen the Moon knows that there are various darker features across its face. Let's go ahead and use the excellent vector art brush features to add some aberrations to our Moon face.

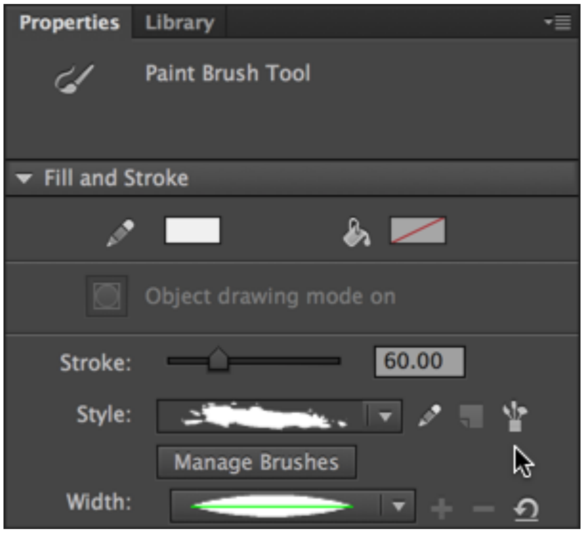
Animate CC has had a Pencil tool for strokes and a Brush tool for fills for quite a long time. They are great for creating free-form shapes, but the new Paint Brush tool is something else altogether. It allows you to draw out paths with rich vector art brushes applied along them.

This art can take the form of Art Brushes or Pattern Brushes. You can use brushes that come bundled within the Brush Library, brushes created with the Adobe Capture CC mobile app, or even brushes created within Animate CC itself with this new release!

- Select the Moon layer within the Timeline. You may want to lock all other layers to be sure you are drawing in the correct layer.

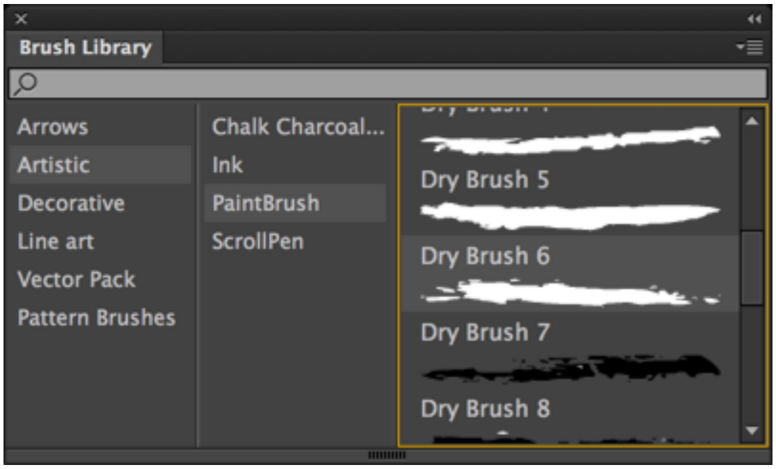
- Choose the Paint Brush tool and choose an off-white color variant for your stroke. Set the stroke value to 60 or so and be sure Object Drawing mode is on. You may also wish to apply a width profile to give the strokes even more variance – you can adjust all of this later, as well.

- Open the Brush Library by clicking the icon that looks like a set of paint brushes in a can to open the Brush Library.

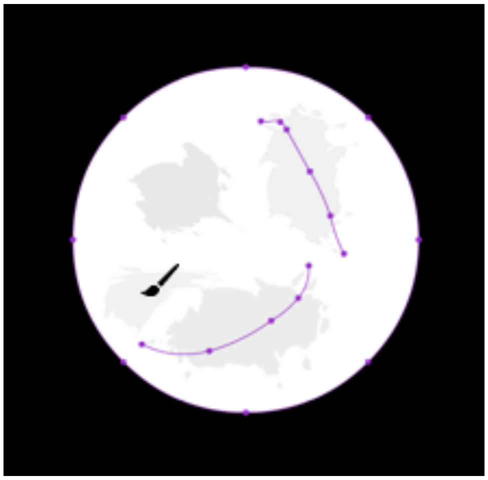


Stroke: 60.00

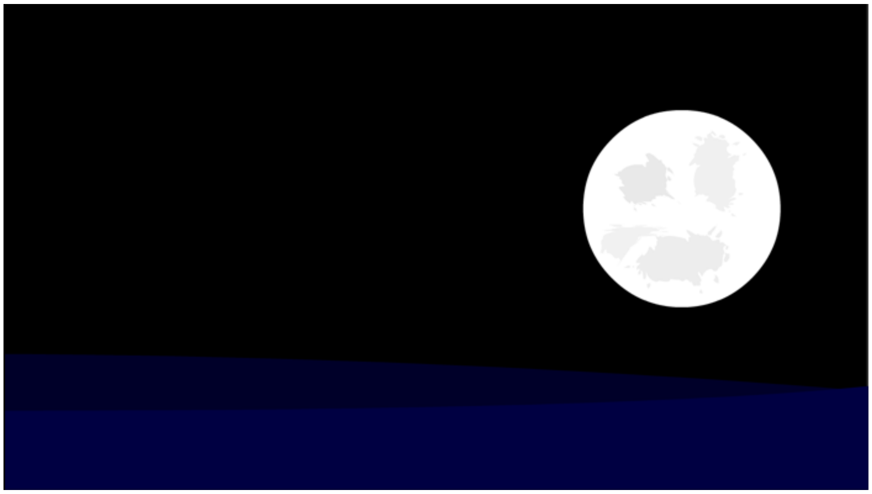
- Drill down from Artistic > PaintBrush and select the brush named Dry Brush 6. Once you double-click a brush with the Paint Brush tool selected, that brush will be applied to the Style dropdown in the Properties panel and you can then close the Brush Library.



- Now all you need to do is paint some of these features across the face of the Moon. Feel free to play with the different properties of each stroke to provide the look you are going for.



- Lock the Moon layer and survey your work. You will now have a pale Moon hanging above the waters against a black sky. Slick.



Next, we'll make some adjustments to the overall background.

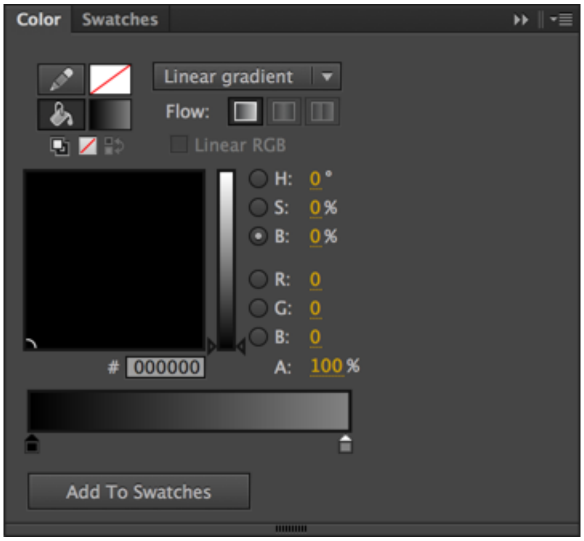
Building the Sky

As the Moon rises into the sky, we want to apply a shift in the ambient light across our stage. To do this, we'll create a shape that fills the entire stage and uses a nice gradient swatch to apply ambient light across our scene.

- Choose the Sky layer and lock down all other layers. This should almost be habitual as you build out any Animate project that involves a number of layers like this.

- Select the Rectangle tool and draw a shape which consists of only fill and no stroke across the entire stage at 394 pixels in height and 700px in width. The x and y position should be exactly 0,0. Be sure this is the case!

- With the Selection tool, select the new shape and open the Color panel. Choose to create a new Linear Gradient which goes from clack to a medium grey – something like #828282.



Linear gradient

Flow:

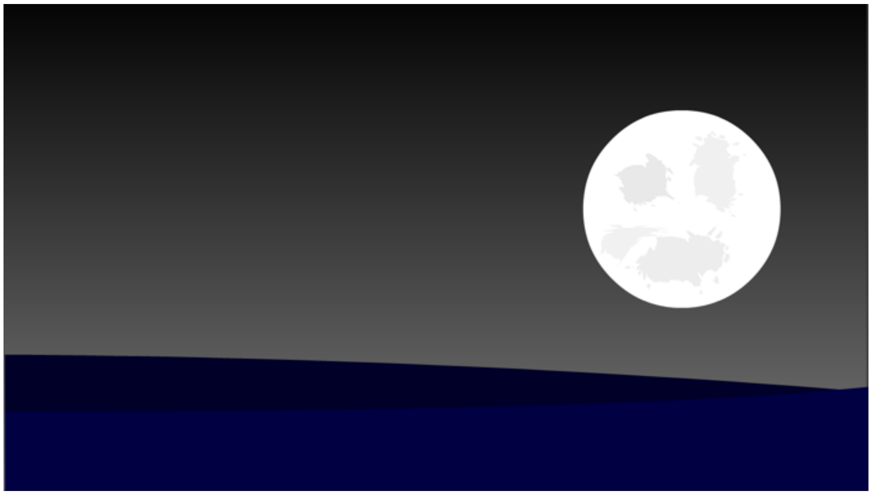
☐ Linear RGB

○ H: 0 °
○ S: 0 %
● B: 0 %
○ R: 0
○ G: 0
○ B: 0
A: 100 %

000000

- Click Add to Swatches to add this to your Swatches panel. You can use the Color and Swatch panels in this way to manage colors within your Animate project. You can also designate certain swatches you create as Tagged Swatches which can persist across multiple objects even when changes to a swatch as applied.

- One last thing here. You'll see that the gradient applied to your shape goes from left to right. We want it to go up to down. Select the Gradient Transform tool (nested with the Free Transform tool) and click on the sky shape. Use the overlayed tools to rotate the how the gradient is applied. Holding SHIFT will constrain your rotation to 45 degrees, making this really simple.



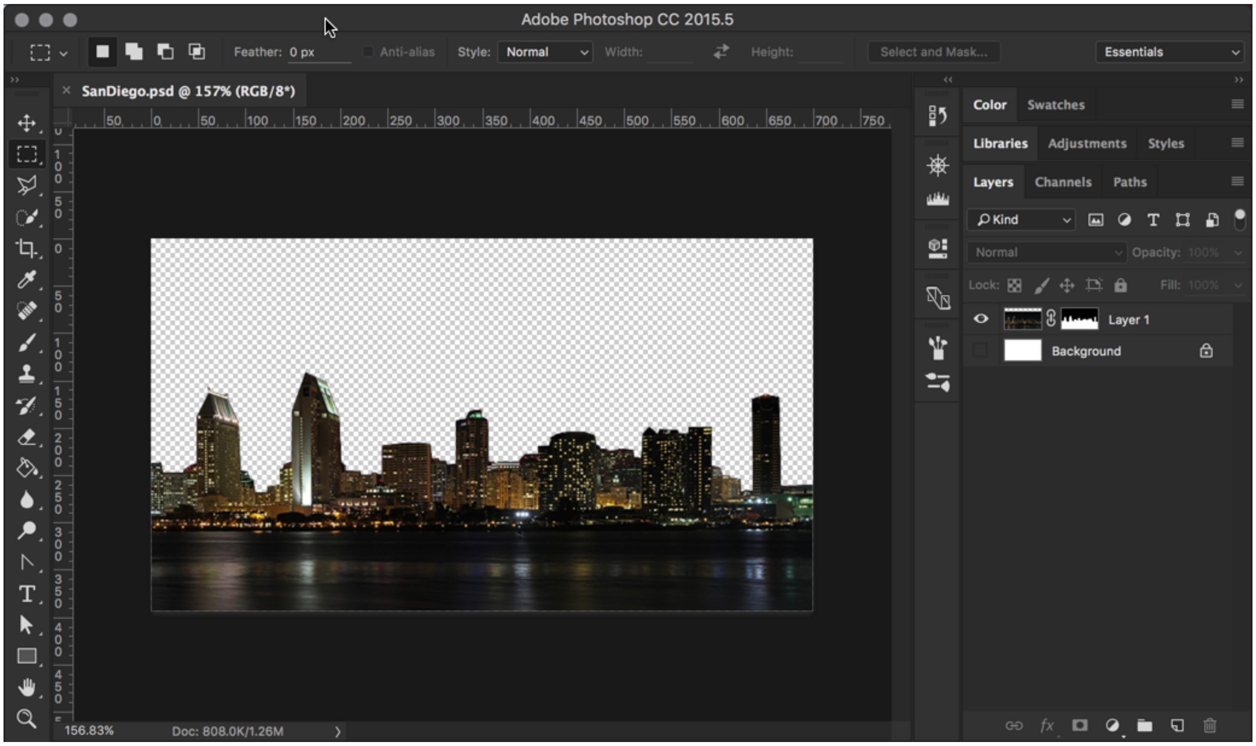
That'll do for now – we'll make further adjustments later on.

Importing the City Skyline

San Diego! A real change from having Adobe MAX in Los Angeles the past number of years. San Diego is a beautiful city and the cityscape across the bay is magnificent at night. Let's add the city skyline to our scene.

I've prepared an asset called SanDiego.png which we will import into our project. In the figure below, we see the file undergoing preparations in Adobe Photoshop CC.

The image was licensed from Adobe Stock, brought into Photoshop, and cropped to the size of our Animate project stage. Finally, the sky was masked out using the masking tools in Photoshop and exported as a PNG with transparency channel.



A transparent PNG can be effectively layered within our Animate project. This is exactly what we need for our San Diego skyline.

Okay – let's integrate the PNG image into our Animate project by importing it.

- First, select the City layer and lock down every other layer.
- With the City layer selected and unlocked, choose File > Import > Import to Stage from the application menu.
- Browse to the file SanDiego.png and choose it with your cursor. Click Open.
- The bitmap image is now brought into the selected layer and placed upon the stage at both an x and y position of 0 – the upper left. Since the image is the exact width and height of the stage, there is no need to adjust its position.

Additionally, since we are using a PNG with transparency, we can see everything behind the image as well.

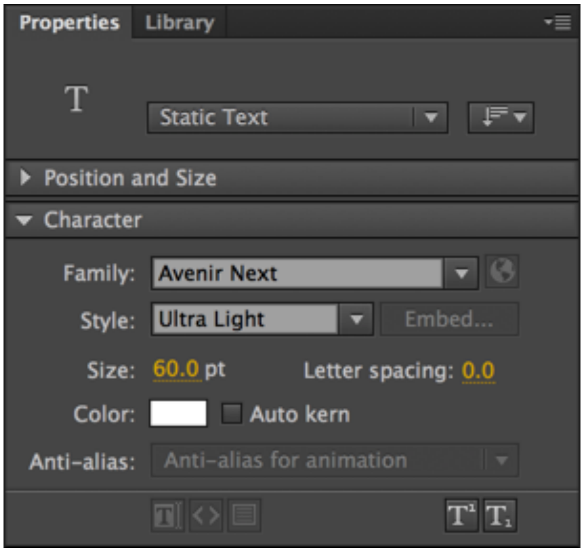
Things are beginning to look pretty slick but we are not nearly finished!

Title Creation and Layout

The last thing we'll do before moving on to the next sequence is create some text on the stage. It'll read "Adobe MAX 2016" and we'll even animate it, along with most of the other assets we've created, in the next sequence.

- Select the Title layer, unlock it if need be, and lock all other layers.
- Choose the Text tool from the Tools panel and set the following properties: Static Text at 60pt – white – choose a thin font family like Avenir Next ultra

light or Museo Sans 100 – something light or even ultralight and thin.



Static Text

Family: Avenir Next

Style: Ultra Light

Size: 60.0 pt

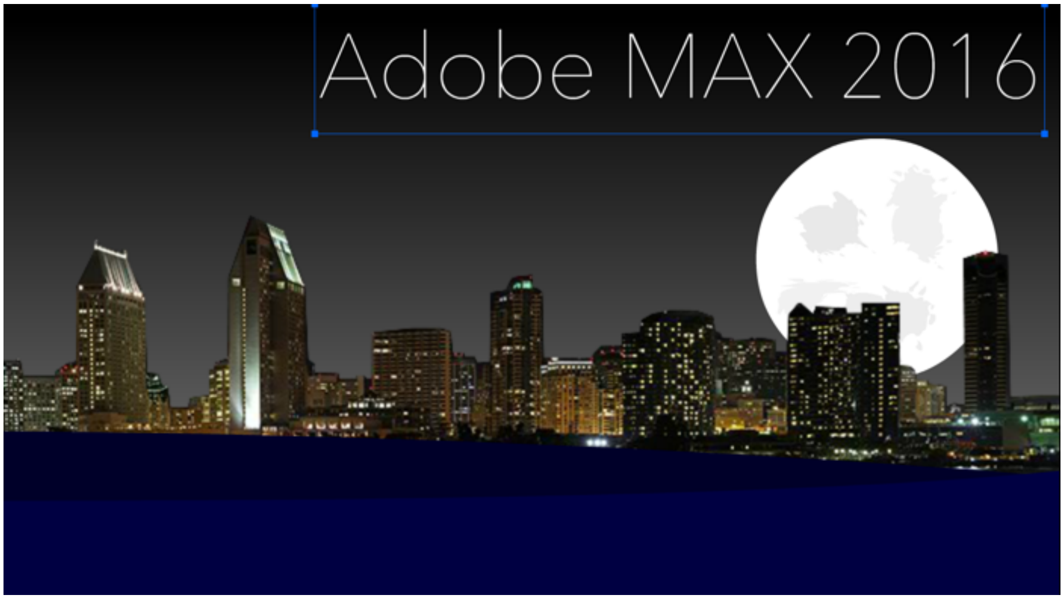
Letter spacing: 0.0

Color:

□ Auto kern

Anti-alias: Anti-alias for animation

- Now draw a text block across the upper right of the stage and type in the text "Adobe MAX 2016".



Okay! Let's move on to the next sequence to add some animation to all we've built.

SEQUENCE 2: Animating Creative Content

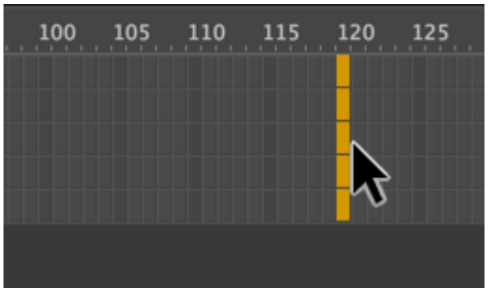
Adobe MAX 2016

With our assets created and properly placed across the Stage and Timeline, it's now time to perform some animation. We will use a variety of symbol types and tweening techniques to animate all aspects of the project in this sequence… and even tap into the new Camera tool to perform some whole-project animation!

Prepare the Timeline for Animation

We have a number of great assets assembled but as of yet – none of our content is animated. We will want to remedy that. The first thing we must do is extend our frame span across the timeline.

- Click and drag across all layers on frame 120 and choose Insert > Timeline > Frame from the application menu.



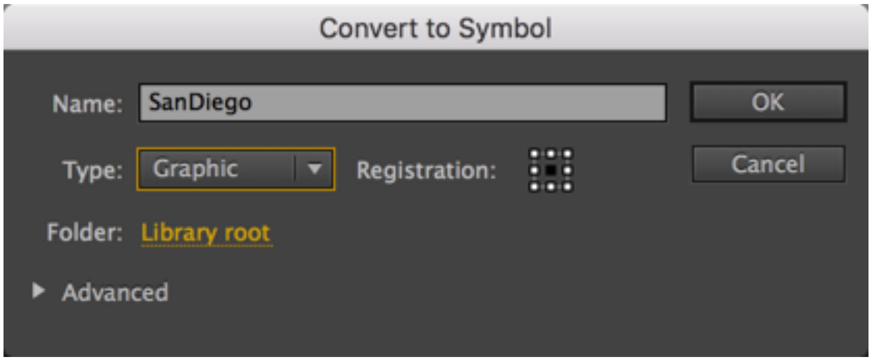
- Scrub the playhead across the newly formed frame span to see that we now can move from frame 1 all the way to frame 120 and the content of each layer is visible across all of these frames.

Turn the City Black

The next thing we'll do is turn the city into a slick, dark silhouette. To do this, we will convert out imported SanDiego.png into a Graphic symbol and apply some color effects to it.

- First, unlock all the layers in the timeline.
- Using the Selection tool, click the SanDiego.png bitmap image on the stage to select it.
- With the bitmap image selected, choose Modify > Convert to Symbol from the application menu. The Convert to Symbol dialog will appear.

- Provide a name of "SanDiego" and choose Graphic as a symbol type for the dropdown. Now hit OK.



Convert to Symbol

Name: SanDiego

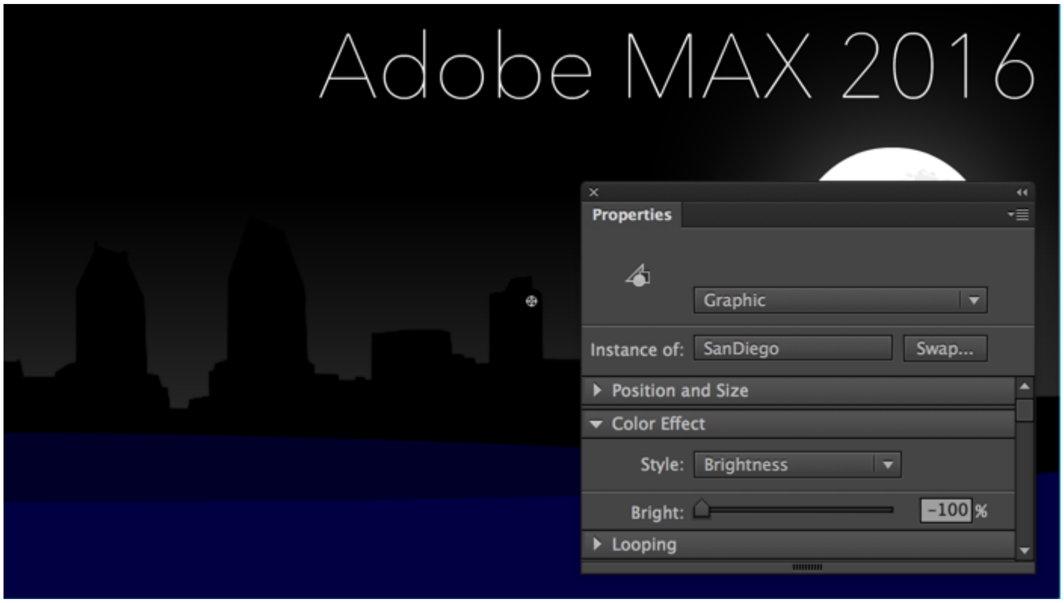
Type: Graphic

Registration:

Folder: Library root

- With the newly created Graphic symbol instance selected, twirl down the Color Effect section of the Properties panel.

- In the Style dropdown, choose Brightness and take the slider all the way down to -100%. This will render the entire city as a black silhouette.



Graphic

Instance of: SanDiego

Style: Brightness

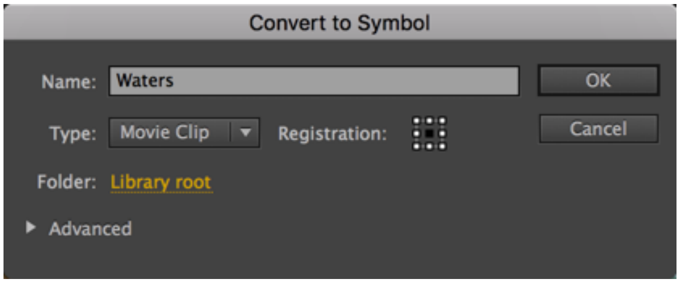
Bright: -100 %

Animate the Waters

To animate the waters, we will first modify the pair into a MovieClip symbol, and then within that symbol perform a Shape tween upon each water shape within.

A nested animation like this, within a MovieClip symbol, will loop over and over again, independent of the main timeline and the number of frames it exists across. It's perfect for anything that requires continual motion.

- We will first use the Selection tool to select both shapes within the Waters layer.
- With both selected, choose Modify > Convert to Symbol from the application menu. The Convert to Symbol dialog will appear.
- Provide a name of "Waters" and choose Movie Clip as a symbol type for the dropdown. Now hit OK.



Convert to Symbol

Name: Waters

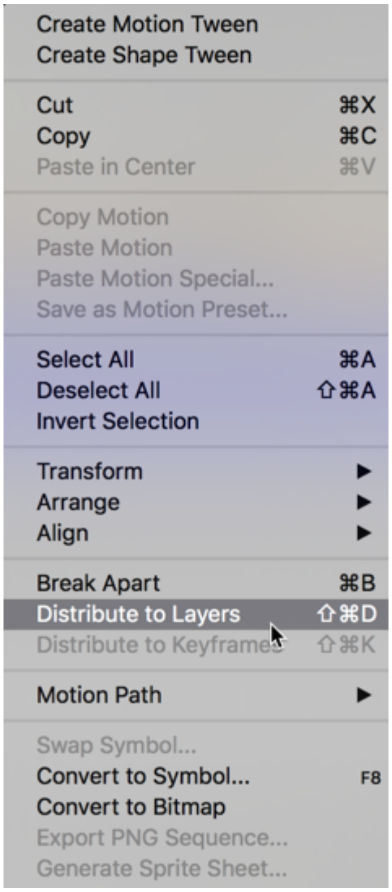
Type: Movie Clip

Registration:

Folder: Library root

- Open the Library panel and double click upon the Waters symbol to edit its contents.
- We will want each shape on its own layer for animation purposes. Select both shapes and right-click. In the contextual menu that appears, choose

Distribute to Layers. This places each shape on its own layer.

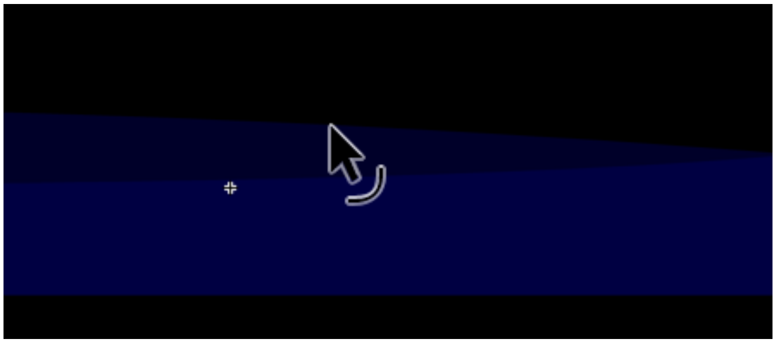


- Make sure that the layers are stacked properly and then rename the foremost layer to Fore and the other to Back.

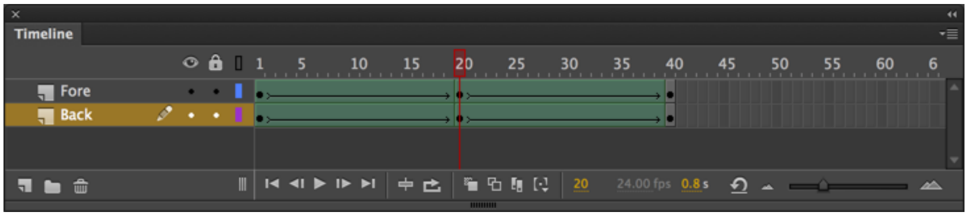
- Extend the frame span of each layer to frame 40.

- You will want to insert keyframes at frame 20 and once more at frame 40 across each layer. To do this, position the playhead at those frames, select the frames across each layer, and choose Insert > Timeline > Keyframe. This will duplicate the previous keyframe at another frame which we can then modify.

- At frame 20, using the Selection tool, hover over the upper edges of each shape until the cursor changes to display an arc and pull them up or down to create a bit of motion.



- The final thing we will do is select all of our frames and right-click to summon the contextual menu. Choose Create Shape Tween and each frame span will take on a green color with arrows pointing from keyframe to keyframe.



- Scrub the playhead to see how Animate CC fills in the frames between the keyframes with motion.

As the name suggests, Shape Tweens are only usable upon vector shape objects within Animate. They can be used to tween one shape into another, move shapes around, change shape color across time, or even modify a gradient transform in an animated fashion.

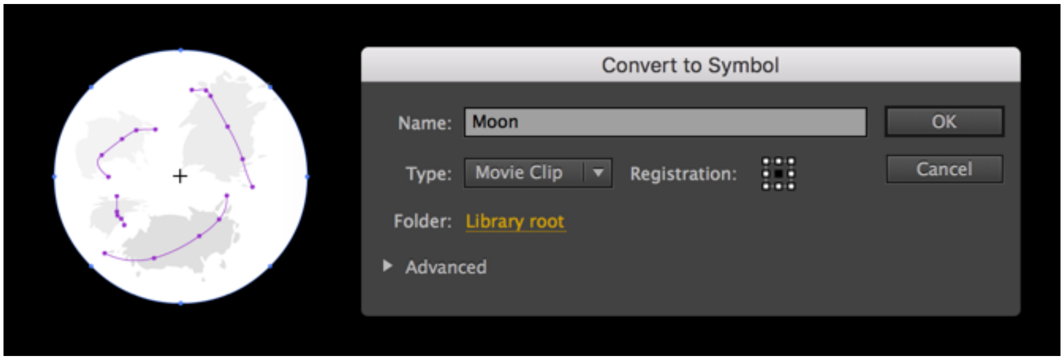
Animate the Moon

A Motion Tween is probably the easiest tween type to use of the 3 (4 counting armatures) that Animate provides.

The reason I state this is that unlike the 2 other tween types, there is no need to expressly insert keyframes across the Timeline. Just move the playhead around, tweak your properties, and Animate does the rest.

We'll use a Motion tween to animate the Moon – but before we do that, we'll need to convert our Moon assets into a MovieClip symbol.

- Unlock the Moon layer and lock all the others.
- Choose the Selection tool and draw a selection rectangle around the Moon and any additional textured assets you created to add variance to the Moon face.
- With all aspects of the Moon selected, choose Modify > Convert to Symbol from the application menu. In the dialog that appears, provide a symbol name of "Moon" and select Movie Clip as the symbol type - hit OK.


Convert to Symbol

Name: Moon

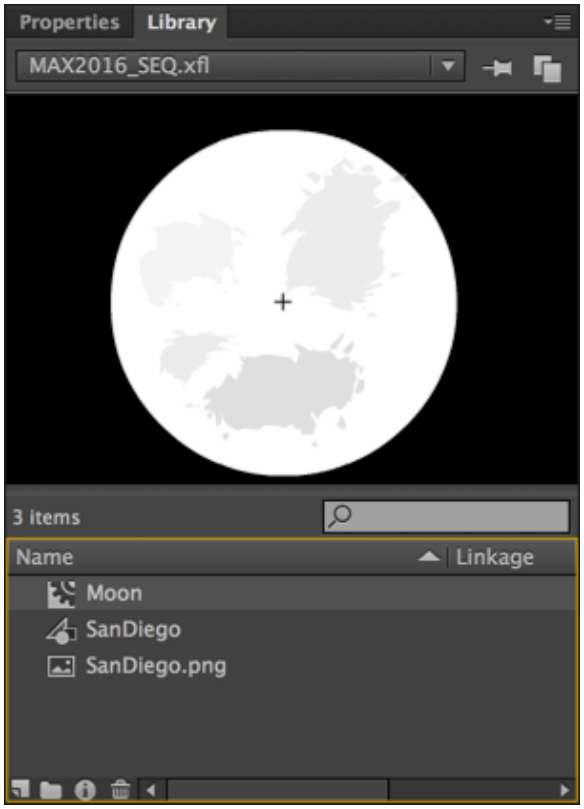
Type: Movie Clip

Registration:

Folder: Library root

- Select the new Moon symbol instance on the stage – look at the Properties panel to verify that you are now using an instance of the Moon symbol.

A MovieClip Symbol has a timeline just like that of the main stage, and is the primary symbol for creating custom functionality within Animate. When a symbol is created, it resides in the project Library.


Properties Library

MAX2016_SEQ.xfl

3 items

Name	Linkage
Moon	
SanDiego	
SanDiego.png	

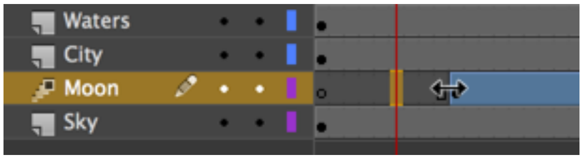
The symbol acts as a blueprint to instruct Animate on how to construct an instance of that symbol. Instances are representations of the symbol within the project stage and timeline. You will notice the SanDiego Graphic symbol we created earlier is there – as well as the imported bitmap image.

We'll now animate our Moon instance using a Motion Tween.

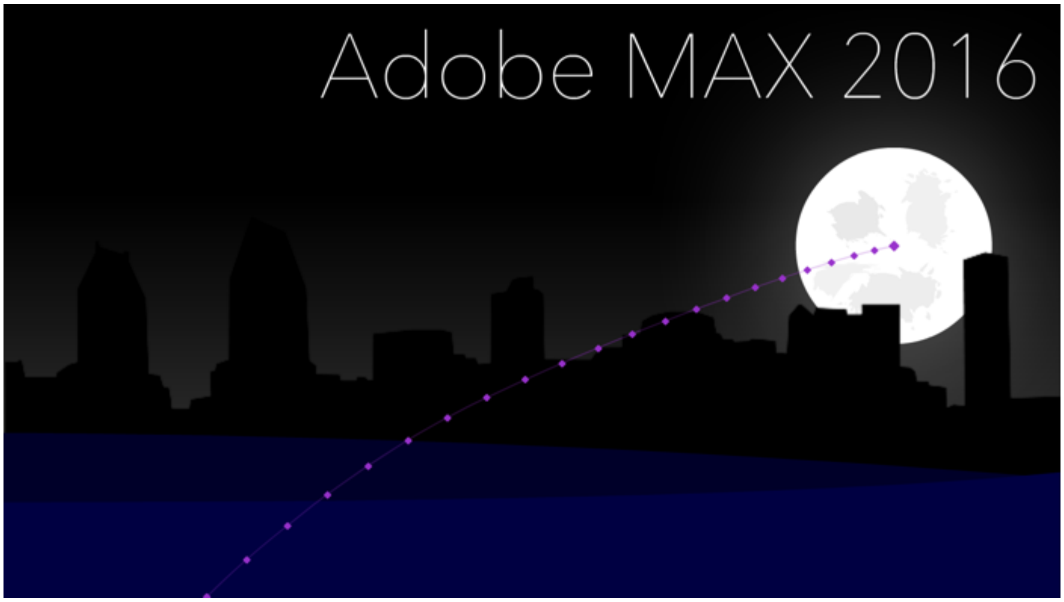
- Right-click upon any of the frames in our Moon layer and choose Create Motion Tween from the menu which appears. All frames will turn a light blue, indicating a Motion Tween.
- On frame 1, move the Moon instance to an x position of 110 and a y position of 420
- Move the playhead to frame 120 and change the x position to 350 and y to 197. Notice a diamond-shaped keyframe appears on this frame.

Motion Tweens work differently from Shape Tweens in that we do not need to explicitly insert keyframes. Simply moving the playhead across the timeline and adjusting instance properties is enough to automatically for keyframes for us.

- Let's tweak the tween a bit. We want the Moon to first appear at frame 10 – so let's shift the tween over by 10 frames. Hover over the very edge of the leading keyframe until the cursor becomes a double-headed arrow. We can now click and drag to shift the keyframe (and entire tween) to the right.



- You may also notice that since creating the Motion Tween that the path the Moon instance travels across the stage is indicated by a visible motion guide. Just as with any path in Animate CC, we can use the Selection tool to hover other this path to push and pull it and form a more natural arc for the movement to follow.

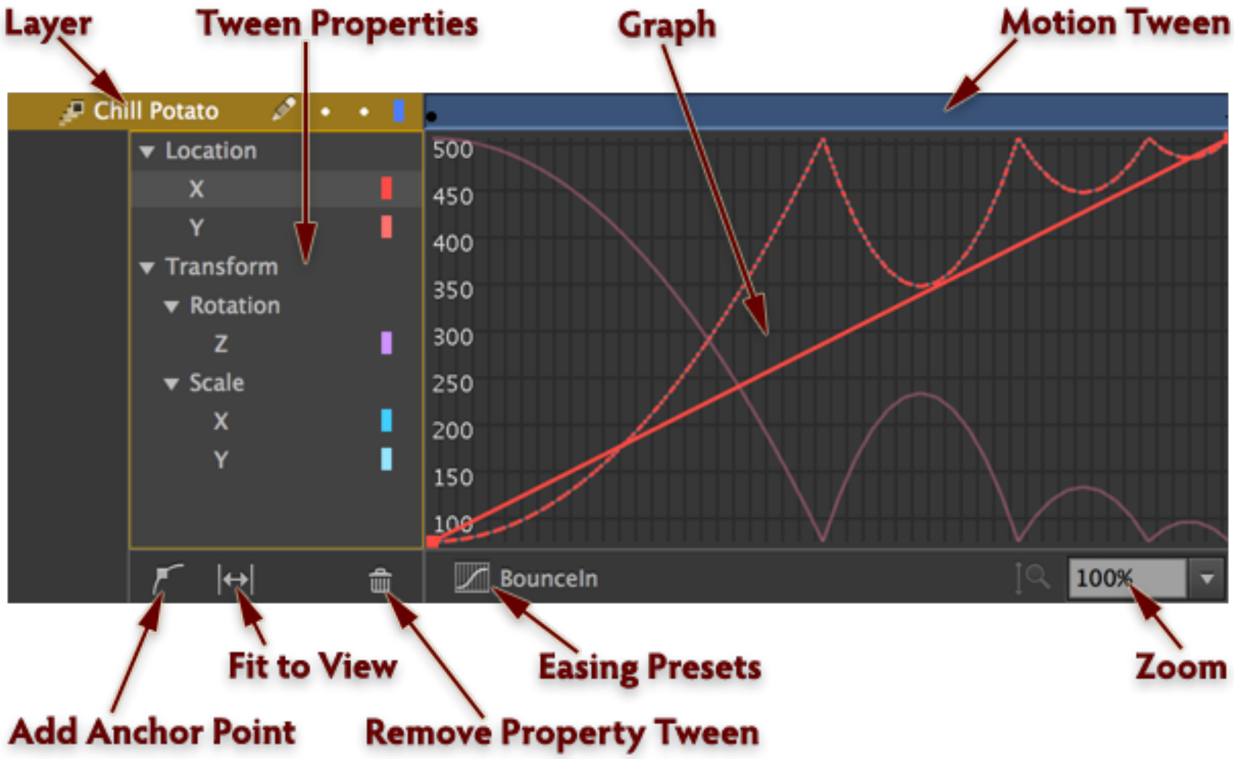


Each dot you see along the motion guide indicates one frame!

Using the Motion Editor

The Motion Editor is a mechanism that can be used exclusively with Motion Tweens to perform advanced easing upon any tweenable property of a symbol instance being animated through an applied Motion Tween.

When using a Shape Tween, a Classic Tween, or animation with an IK Armature, the Motion Editor will not be available – yet other easing mechanisms can still be applied through the Properties Panel.



As you can see – there are quite a few tools, functions, and display adjustments to consider while using the Motion Editor. Here follows a rundown of everything detailed in the above figure.

Layer: This is the layer which contains the Motion Tween that the Motion Editor is tied to.

Motion Tween: Double-click upon the Motion Tween to expand or collapse the Motion Editor.

Tween Properties: This column will contain all the properties of the instance being tweened which currently have some sort of value change attributed to them.

Various position, scale, skew, rotation, color, and filter properties can all be manipulated through the Motion Editor.

Graph: The graph area displays all of the different property curves for this tween.

Selecting a particular property along the side will highlight the associated property curve. The selected curve can then have its anchor points modified using the Selection Tool.

Add Anchor Point: Allows you to add anchor points to a property curve.

Once an anchor point is added, this tool is automatically deselected so that the new anchor point can be further manipulated with the default Selection Tool.

Fit to View: This button will expand the Motion Editor along the entire width of the Timeline, making it easier to work within the graph area and further manipulate the various property curves.

Remove Property Tween: Selected property tweens from the left-hand column can be removed with this function. They can be added back in by adjusting those properties across the Motion Tween as you normally would, and then further tweaking them within the Motion Editor.

Easing Presets: Upon choosing this, a small interface will appear which allows you to apply easing presets to the selected property tweens.

Zoom: Will zoom the graph area in height for more precise modification of property tween curves.

There are many options available within Animate CC to add a bit more weight and realism to your animated content.

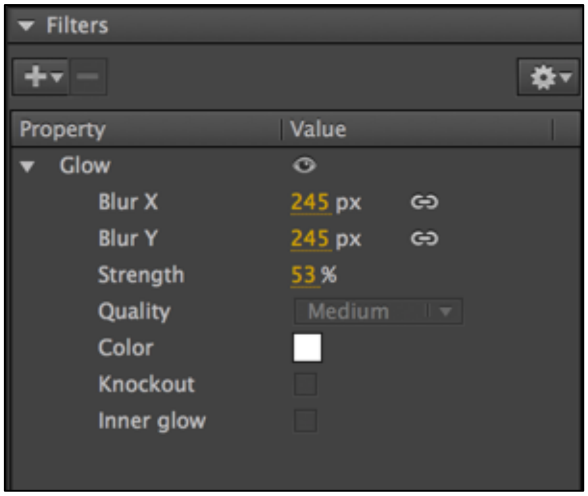
Easing with the Properties panel or with the Motion Editor can enhance any project type, including rich ads, web animations, interactives, and more!

Adding some Moonlight

One more thing before moving on… the Moon needs moonlight. Let's give it a mysterious glow!

- Using the **Selection** tool, select the Moon instance and take a look at the **Properties** panel.

- Within the **Filters** section, click the + icon to add a **Glow** filter to this instance. It should be a mysterious, diffuse glow… so set the **Blur** across both axes to 245 and bring the strength down to 53%. Moonlight is cool and white… very different from sunlight… so adjust the color accordingly.

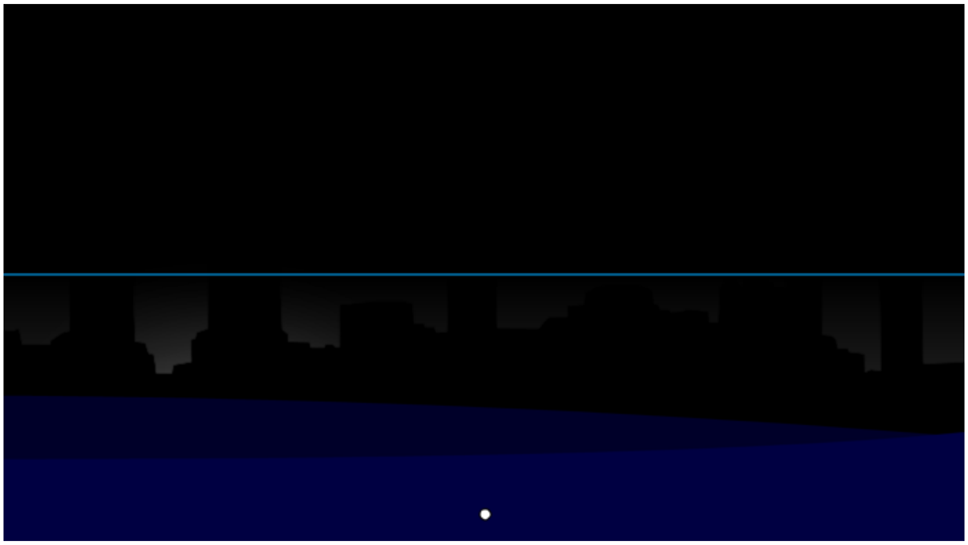


Property	Value
Glow	
Blur X	245 px
Blur Y	245 px
Strength	53 %
Quality	Medium
Color	■
Knockout	□
Inner glow	□

Animate the Sky Gradient

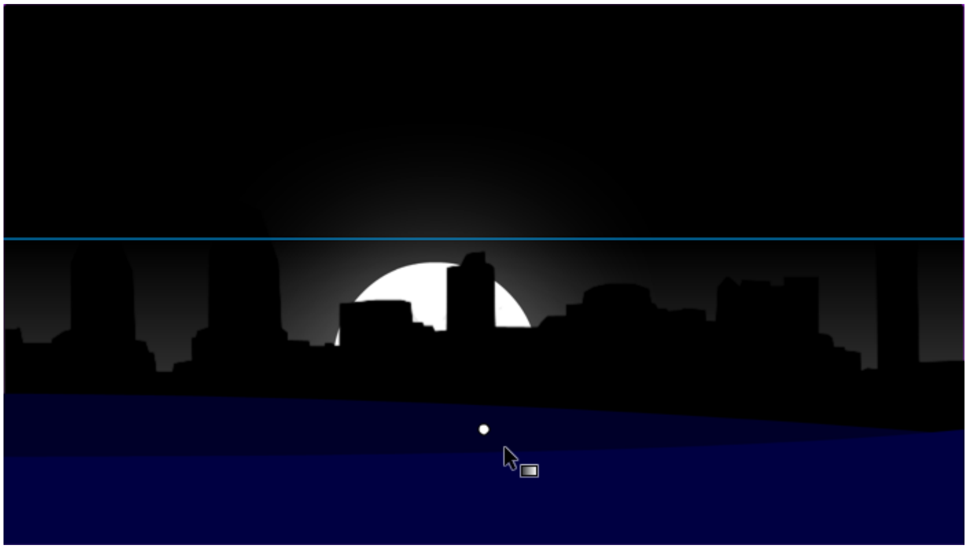
We have a nice gradient wash to our Sky which runs from pure black to a lighter grey color. The idea here is to have the gradient appear as if it is derived from the Moon itself… from a technical aspect… we want the gradient to bring emphasis to the darkened City silhouette.

- Unlock the Sky layer and lock down all others.
- Select the **Gradient Transform** tool and click on the shape within the Sky layer to bring up the Gradient Transform tool overlay.
- Shift the gradient so that the bases of the building receive some definition… but the tops are swallowed by darkness.

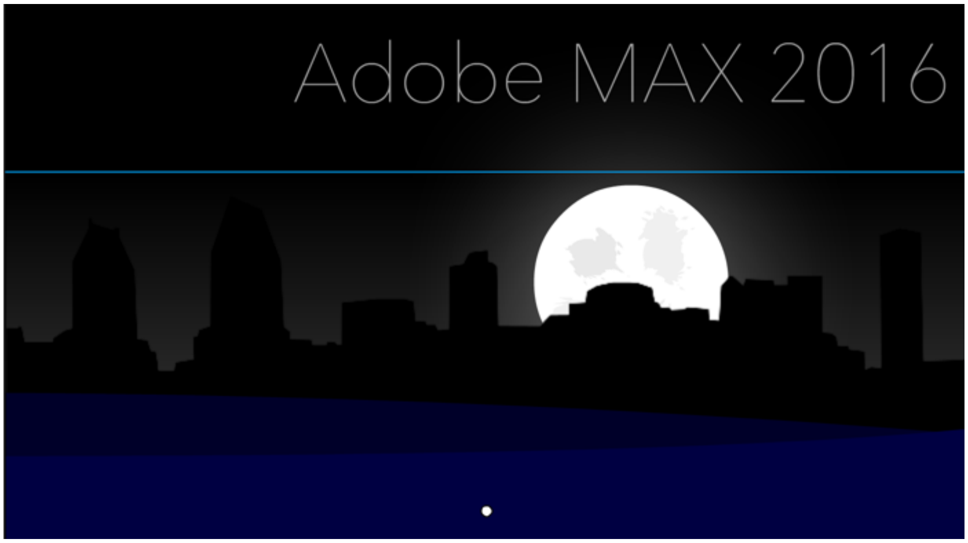


- Move the playhead to frame 20 and insert a keyframe by choosing **Insert > Timeline > Keyframe** from the application menu.
- Use the **Gradient Transform** tool to once again adjust the gradient wash upon the shape. There are no further adjustments to make upon this frame.
- Move the playhead to frame 50 and insert another keyframe.
- At this frame, the Moon is beginning to rise behind the City. Using the **Gradient Transform** tool, adjust the gradient so that it rises toward the tops of

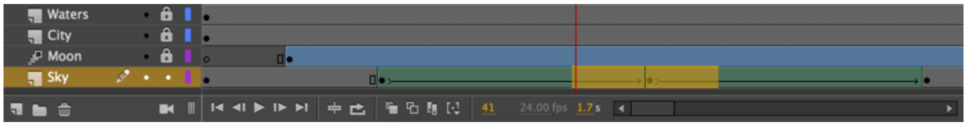
the buildings along with the Moon.



- Move the playhead to frame 80 and insert a final keyframe.
- Using the **Gradient Transform** tool once more, adjust the gradient so that it rises even farther across the sky – illuminating all of the buildings which compose the City.



- Finally, select a set of frames between all three keyframes, including the center keyframe, right-click this selection and choose **Create Shape Tween** from the menu that appears.



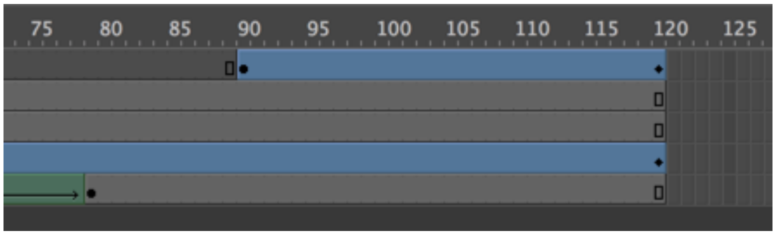
We now have an animated sky – drenched with cold moonlight.

Title Fade Transition

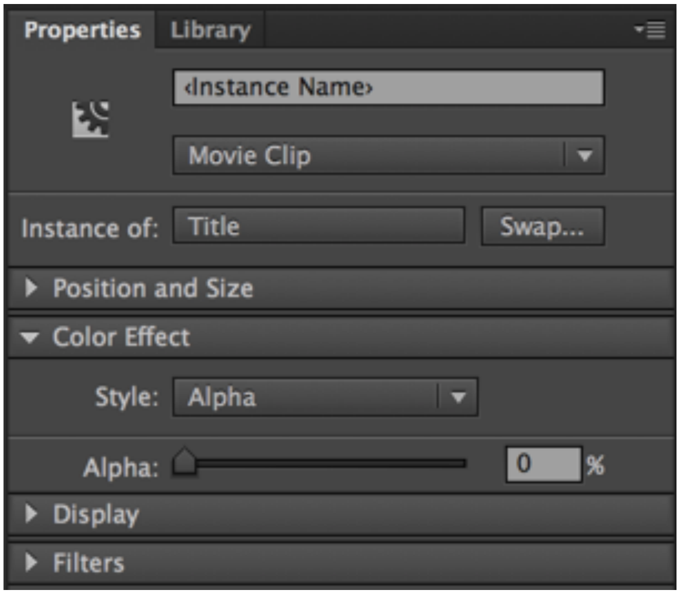
The final asset to animate on its own is our Title text. Toward the end of our animation, we will want this to fade in and remain visible thereafter.

- Unlock the Title layer and lock all the others.
- Using the Selection tool, click on the Text to select it and choose Modify > Convert to Symbol from the application menu. A dialog will appear immediately.
- Give the new symbol the name of "Title" and choose Movie Clip for its type. Click OK and the Text object is now contained within a MovieClip symbol… and instance of which now exists on the stage.
- With the Text now converted… click anywhere upon the frames within the Title layer and right-click. Choose Create Motion Tween from the menu that appears.
- Using the cursor, hover over the left edge of the initial keyframe on the Motion Tween which was just created. Once the cursor becomes a double-headed arrow… click and drag to the right – resizing the entire frame span – until you reach frame 90. At this point, you can release the cursor. The Title instance

now exists only on frames 90-120.



- Move the playhead to frame 90 and use the Selection tool to select the Title instance on the stage.
- With the Title instance selected, have a peek at the Properties panel and in the Color Effect section, be sure Alpha is selected from the Style dropdown and bring its value all the way down to 0%.



Instance of: Title

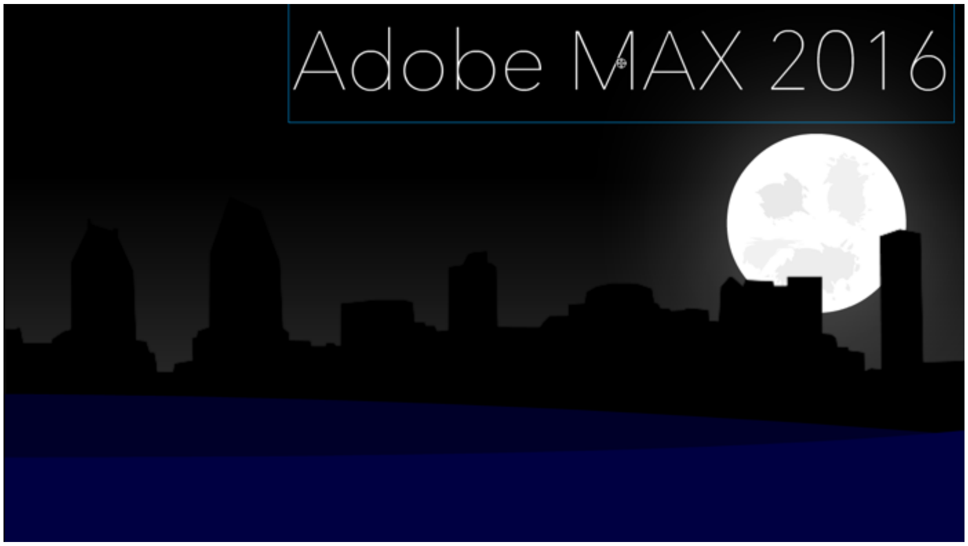
Style: Alpha

Alpha: 0 %

- Move the playhead to frame 120 and select the now-invisible Title instance on the stage. Even though you can no longer see it, clicking the spot upon which it exists will enable this selection.

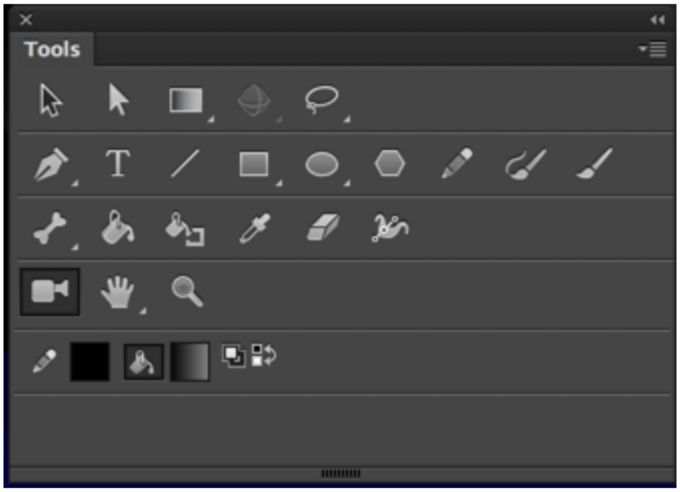
Additionally, you can enable outlines for that layer and even non-visible entities such as this will appear as outlines.

- Have another look at the **Properties** panel. Bring the **Alpha** value all the way back up to 100%. You've now created a quick fade-in for the Title.



Camera Animation

The new **Camera** functionality present within Animate CC allows animators to simulate the user of a real-life camera. Previously, animators relied on third party extensions of varying quality and compatibility, or modified their animations to mimic a camera's movement.



Animators can use the following camera features that are integral to any motion film for efficient storytelling:

- Panning with the subject of the frame
- Zooming in the object of interest for dramatic effect
- Zooming out of a frame to remind the viewer of a larger picture
- Modify the focal point to shift the attention between subjects
- Rotating the camera view
- Apply color effects and tint to an entire scene

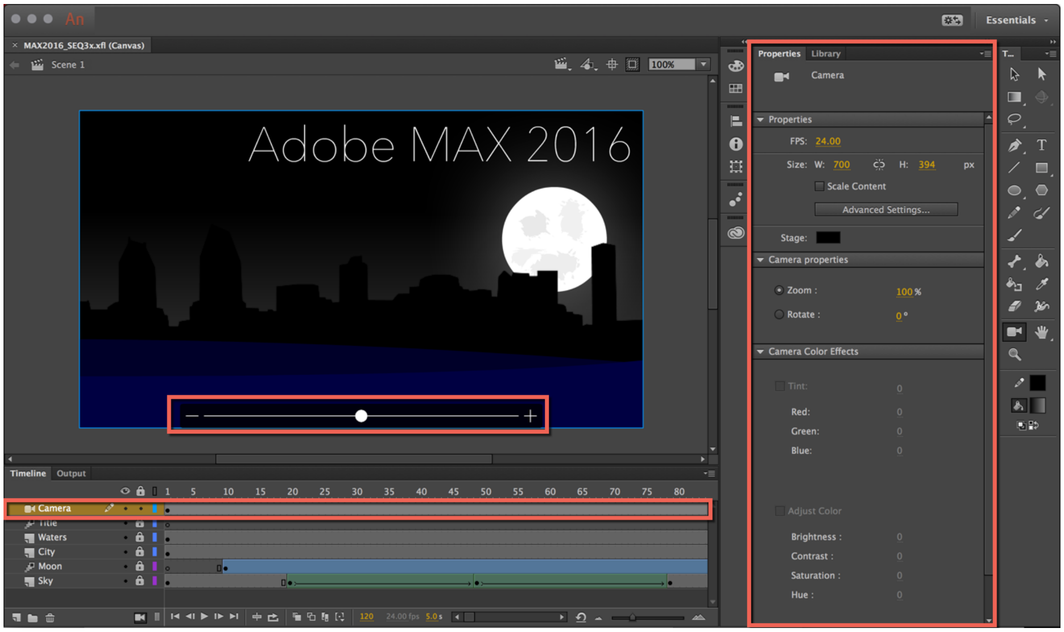
When you set a camera view for your composition, you look at the layers as though you were looking through that camera. The camera layer behaves like a regular object; you can add tweens or keyframes… and animate it like other objects in the document.

Using the Camera tool in Animate CC, we can zoom, pan, and even rotate our entire scene as a whole unit.

Let's use the Camera to start out with a focus close to the shoreline, hold this a bit, and then zoom back to view the entire stage as the moon rises.

- The first thing to do when employing the **Camera** tool is to choose it from the **Tools** panel. Click the icon that looks like a camera, as seen in the figure above.
- This enables the **Camera** for the document, displaying a special Camera layer within the Timeline, Camera overlays on the stage, and Camera Properties

within the Properties panel.



Camera

Properties

FPS: 24.00

Size: W: 700 H: 394 px

□ Scale Content

Advanced Settings...

Stage:

Camera properties

● Zoom : 100 %

○ Rotate : 0 °

Camera Color Effects

□ Tint: 0

Red: 0

Green: 0

Blue: 0

□ Adjust Color

Brightness : 0

Contrast : 0

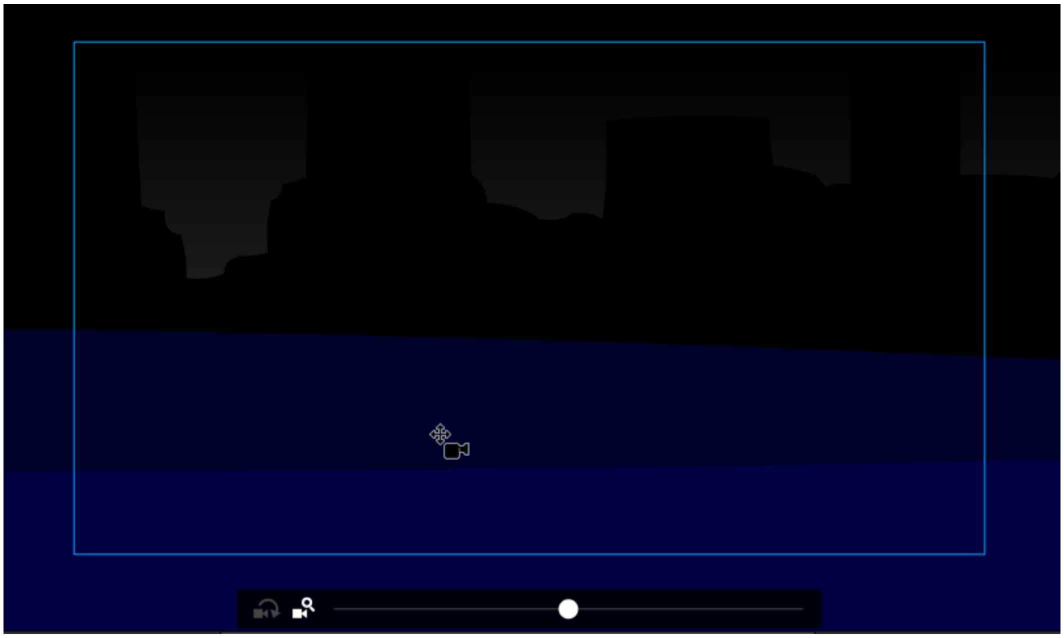
Saturation : 0

Hue : 0

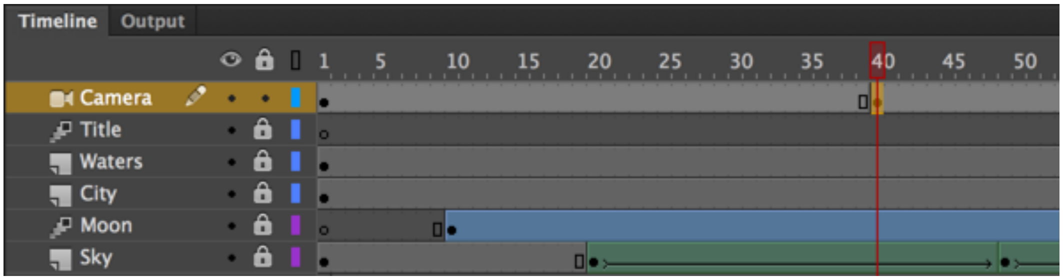
With the Camera enabled for our document, we can now create some animated camera movements across the entire stage.

- Select the **Camera** tool and you'll notice we get a special **Camera** layer above all of our other layers and a nice **Camera** control overlay below the stage.
- Use the overlay in zoom mode (default) to zoom in about 240%.

- Now click and drag the stage to pan the Camera so that the lower left portion of the stage is in view – with a focus on the water.



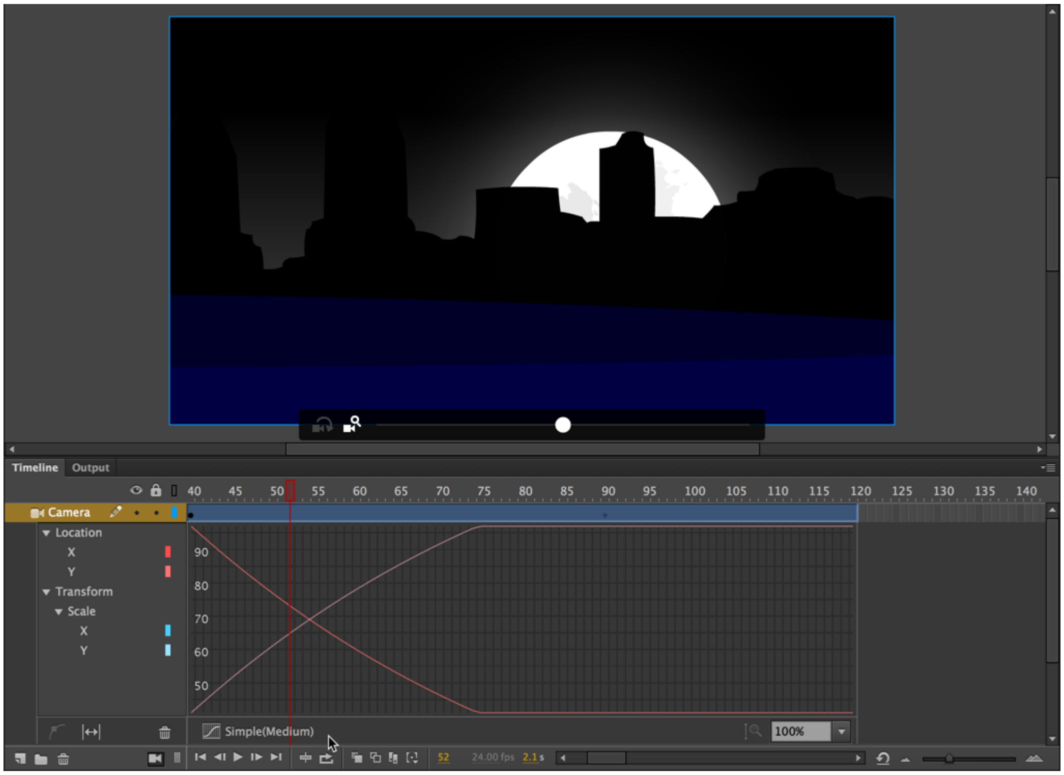
- Move the playhead to frame 40 and choose Insert > Timeline > Keyframe from the application menu to insert a new keyframe. This will hold the Camera at this view for the span of 40 frames.



- Right-click on any of the frames which follow frame 40 and choose Create Motion Tween. This produces a Motion Tween from frame 40 all the way to frame 120.

- Move the playhead to frame 90 and zoom back out to about 102% using the Camera tool. You will also need to pan the Camera view to re-center the scene.

- The final thing we will do is use the **Motion Editor** to add a simple ease to our Camera motion. Double click the Motion Tween to reveal the Motion Editor and add an ease of type **Simple (Medium)** to all **Location** and **Transform** properties. This will really add a bit of realistic movement to our Camera.



Camera Tint and Color Effects

You will notice we have the ability to apply tints and effects to our Camera as well. However, since we are using HTML5 Canvas as a document type, these are not available to us.

If you want to use all Camera effects – choose an ActionScript 3.0 document type.

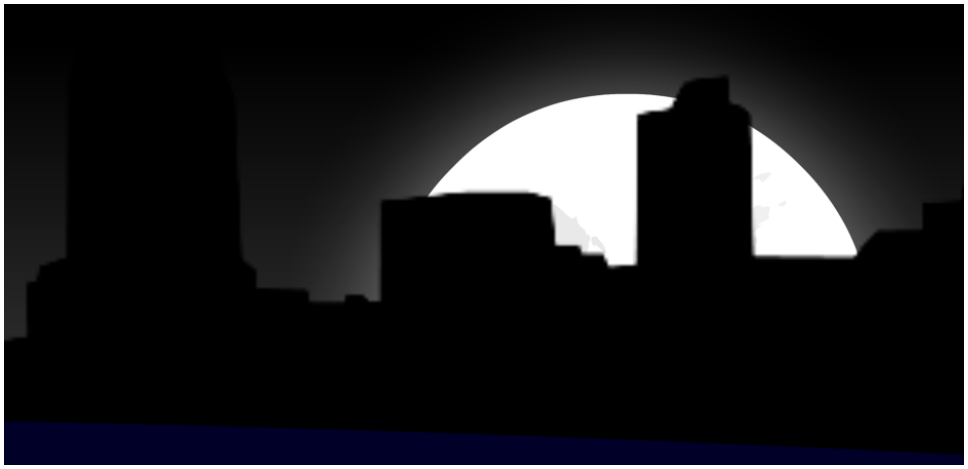
Related Tools

Note that you can also toggle the Camera off with the Camera icon at the bottom of the timeline.

Additionally, if you want to view Camera movement (or really any content) as you would when published… you can use the stage clipping icon above the stage to visually clip anything off-stage during authoring.

Fixing San Diego

If we test our animation, we can see that the City… especially when the Camera is really close to the buildings… things get a bit chunky and pixelated. Have a look at the figure below and see for yourself.

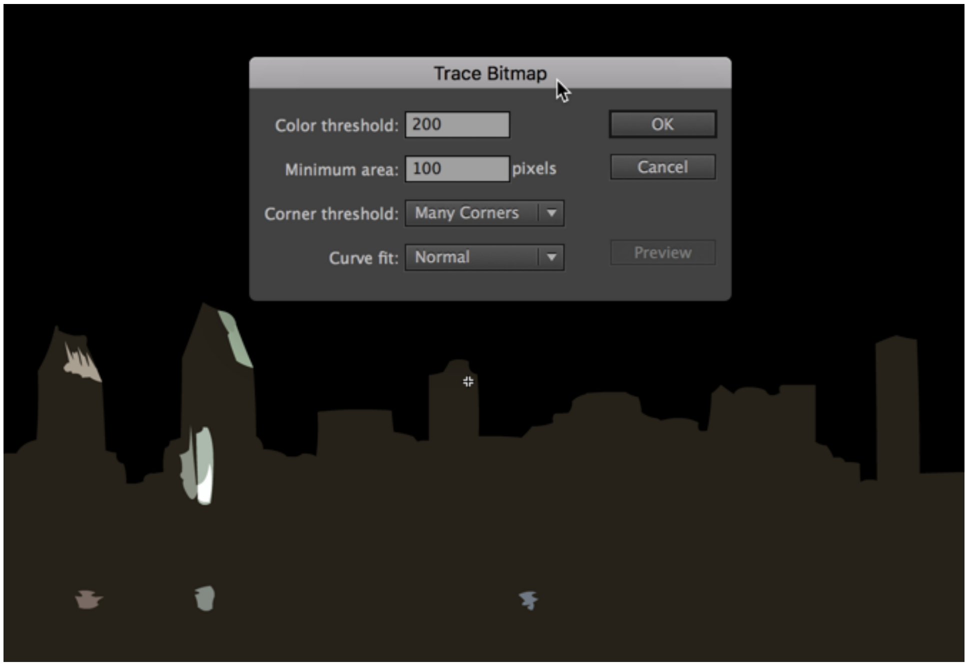


So what's going on? The Moon and Waters are perfectly crisp! Recall that the City instance is based upon an imported PNG. A pixel-based image format. Everything else was created within Animate CC as vectors.

If only we could convert the City image to vector… oh yeah, we can!

- Open the Library and locate the City Graphic symbol.
- Double-click into the symbol to enter edit mode.
- Now, select the bitmap image of the cityscape and choose Modify > Bitmap > Trace Bitmap from the application menu.

- In the dialog that appears, set the color **threshold** to 200 and the **minimum area** to 100. Go ahead and click the Preview button to see how it will look.



Color threshold: 200

Minimum area: 100 pixels

Corner threshold: Many Corners

Curve fit: Normal

Play with the settings a little more if you like and hit OK when finished. Your bitmap image is now a true vector with crisp, clean lines – infinitely scalable.

SEQUENCE 3: Enabling Interactivity

While animation is certainly a valid use for Animate CC, the application is great for implementing interactivity as well. In this sequence, we'll wire in some user interaction using JavaScript and the Actions panel to control the Timeline playback.

ActionScript and JavaScript

For nearly 20 years, the programming language used within Animate (Flash Professional) projects has been ActionScript. So why the current shift to JavaScript? Well, it isn't exactly as simple as that.

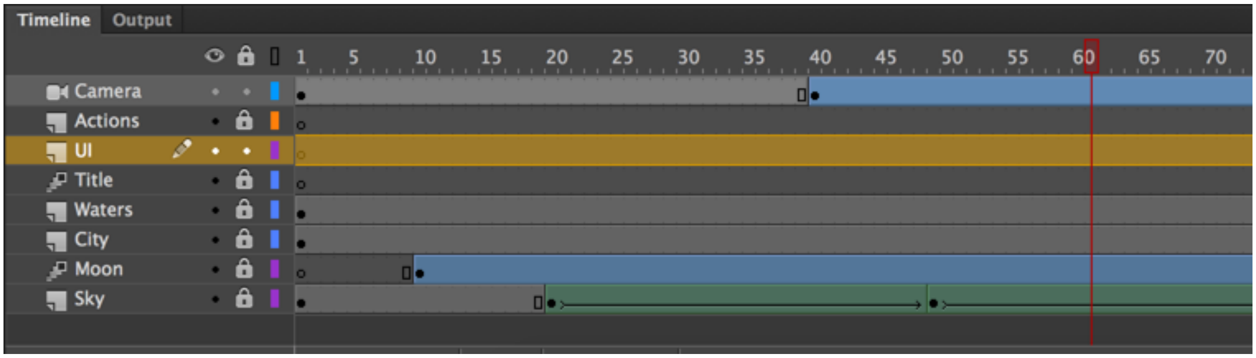
For one thing, Animate is not a JavaScript-only application. You use JavaScript in projects which require it, such as HTML5 Canvas and WebGL… but you can still use ActionScript on projects which are to be published for Flash Player or AIR, including AIR for Android and iOS – and Adobe continues to update these runtimes.

It's important to note that Animate is not a JavaScript and web-centric application – but is rather a platform and language agnostic application for working across a variety of targets.

Build your content in Animate CC and use it across all target platforms! More about this in Sequence 4…

Prepare the Timeline for Interactivity

We'll need to create two additional layers for our interactive elements. One is specifically to hold our code – name this "Actions". The other will hold any user interface elements we will visually place on the stage. Name this one "UI".



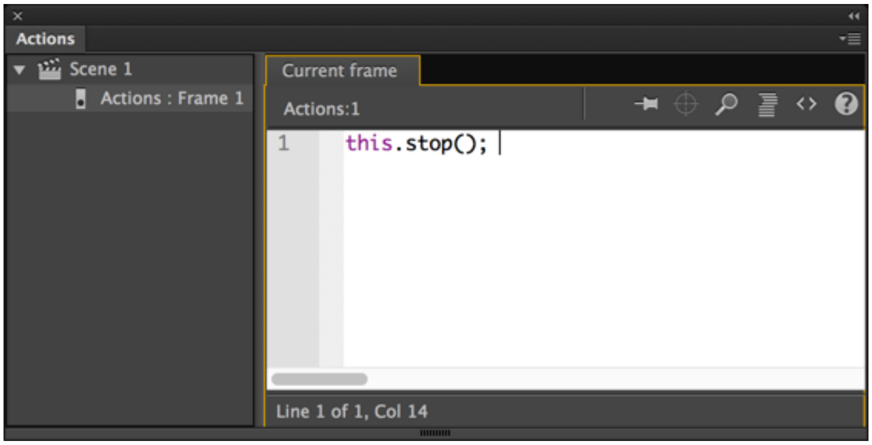
Both of these will appear directly beneath the Camera layer as the Camera is always on top of the timeline.

Stopping the Playhead

If you go ahead and test the project at this point by using Control > Test from the application menu, you'll see how great everything has come together… but the animation loops continuously… let's stop it on both the first and final frames.

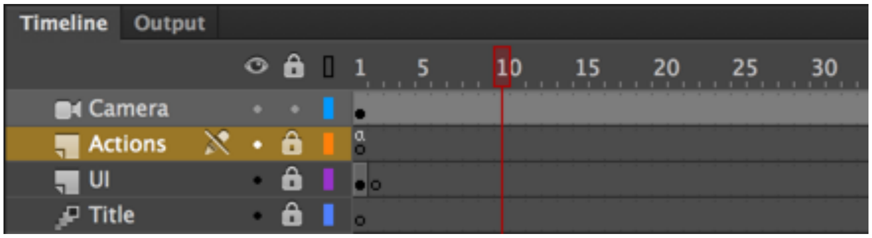
1. Select the keyframe which already exists on frame 1 of the Actions layer and open the Actions panel from the application menu: Window > Actions. Now type the following code into the script pane:

```
this.stop();
```



```
this.stop();
```

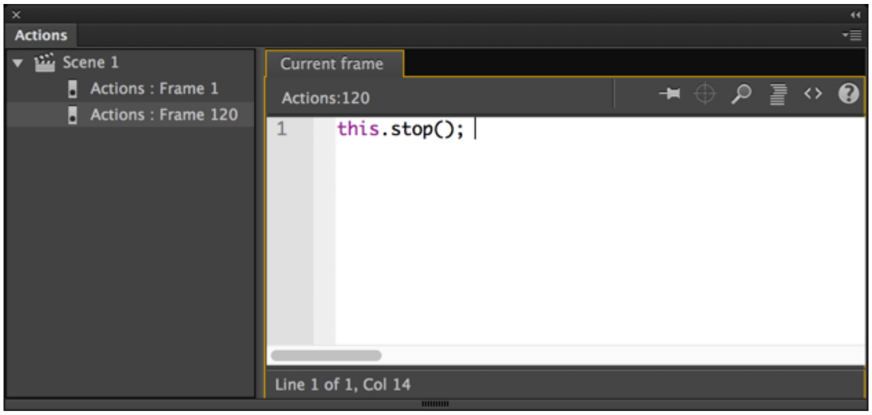
2. Notice that the keyframe now has a small a character within it, indicating there is code attached to that frame.



3. Now we will move on to the final frame. First, we must insert a new keyframe on frame 120 to hold the code that stops the playhead from looping. Move the playhead to frame 120 and choose Insert > Timeline > Keyframe from the application menu. A new, empty keyframe now exists on that frame.

4. Select the newly created keyframe and open the Actions panel again (if closed). Now type the stop code into the script pane just like last time:

```
this.stop();
```

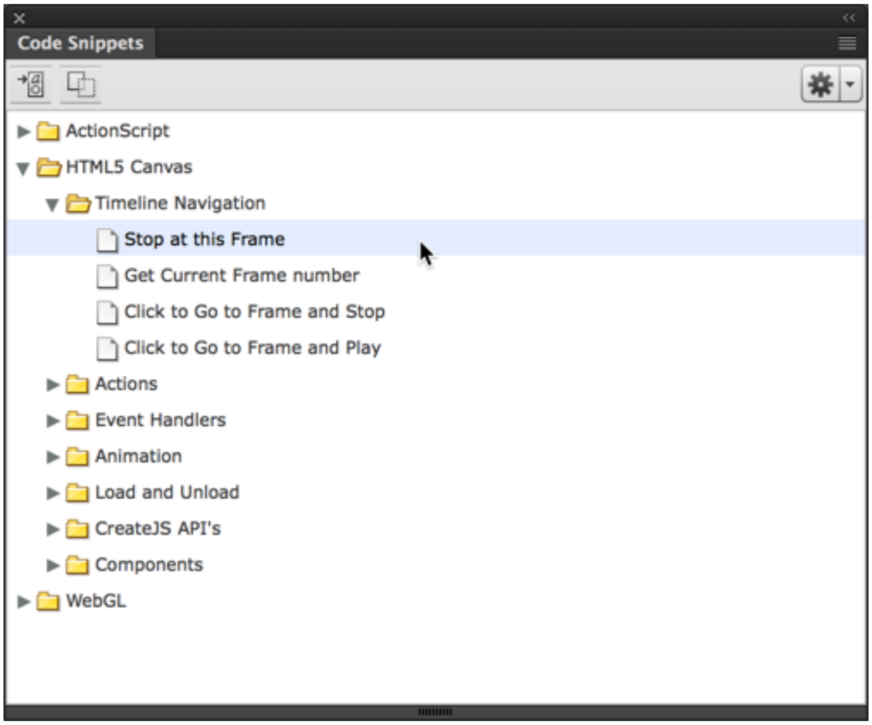


```
this.stop();
```

5. That's all it takes to stop the playhead on the selected frames. Test the movie and see for yourself – everything is frozen at frame 1!

Using Code Snippets

The Code Snippets panel is a great way for those who are new to working with code to learn how to do some fundamental things within Animate projects. Access this panel from within the Actions panel or by choosing Window > Code Snippets from the application menu.



Code Snippets

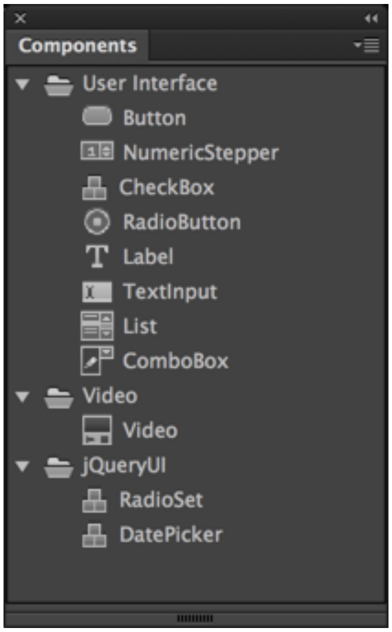
- ActionScript
- HTML5 Canvas
  - Timeline Navigation
    - Stop at this Frame
    - Get Current Frame number
    - Click to Go to Frame and Stop
    - Click to Go to Frame and Play
  - Actions
  - Event Handlers
  - Animation
  - Load and Unload
  - CreateJS API's
  - Components
- WebGL

The snippets included are divided into 3 main categories: ActionScript, HTML5 Canvas, and WebGL. These correspond to whichever target platform you are working with.

HTML5 Components

Yes! If using HTML5 Canvas a target document type, you can now use components.

If you open the Components panel using Window > Components from the application menu, you will see a list of components for use with the active target platform. Since we are using HTML5 Canvas as our target document type – we see components for use with HTML5 Canvas.



To use any of these components, we can drag them onto the stage and then, with the component instance selected, modify the component properties through the Properties panel just as we do with regular symbols – except components will always expose special properties for us to manipulate.

Building UI Elements

Before we write any more code, let's go ahead and build some visual elements which will inform the viewer how to begin and then replay our animation. We will create these on the UI layer – so be sure that layer is selected and unlocked.

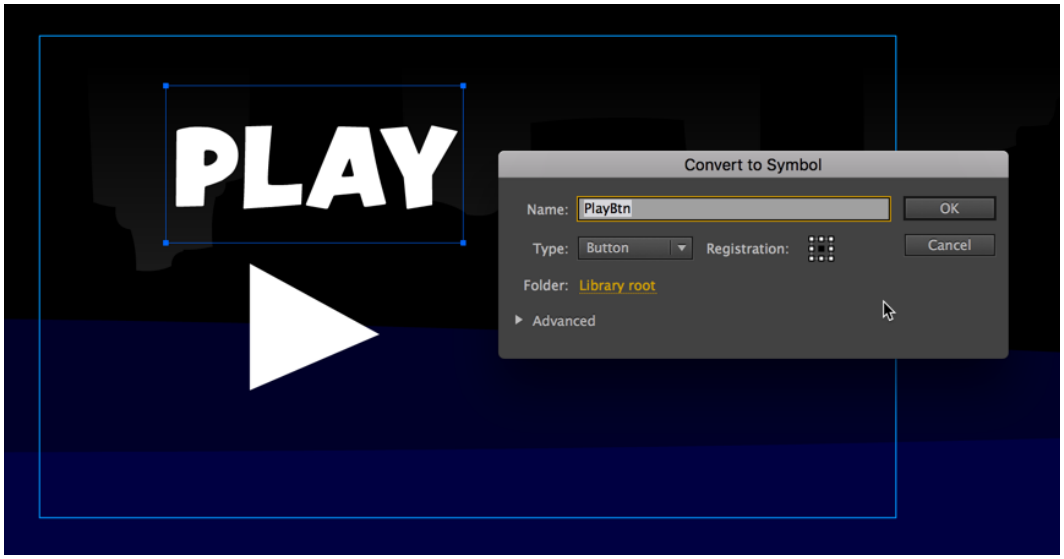
We will create a clickable button to play the animation and then one to replay the animation. After these are created, we will write the code to handle the desired actions.

- The most obvious thing to do for a play button is to have the word "PLAY" with a triangular play button icon. Select the Text tool and choose a font and set your text properties – be sure you are using Static text. I've made mine big and white so it stands out against the dark background.

- Click on the stage and type in the text value of **"PLAY".** You will now have a very clear message for the user to read.

- Choose the Pen tool and create a triangular shape beneath the text which appears like a play button icon. You could also use a rectangle and then manipulate the various anchor points to create this triangular shape. I have made this the same color as my text.

- Select both objects and choose Modify > Convert to Symbol from the application menu. Choose to create a Button symbol and name it "PlayBtn". Hit OK and the button is created for you.



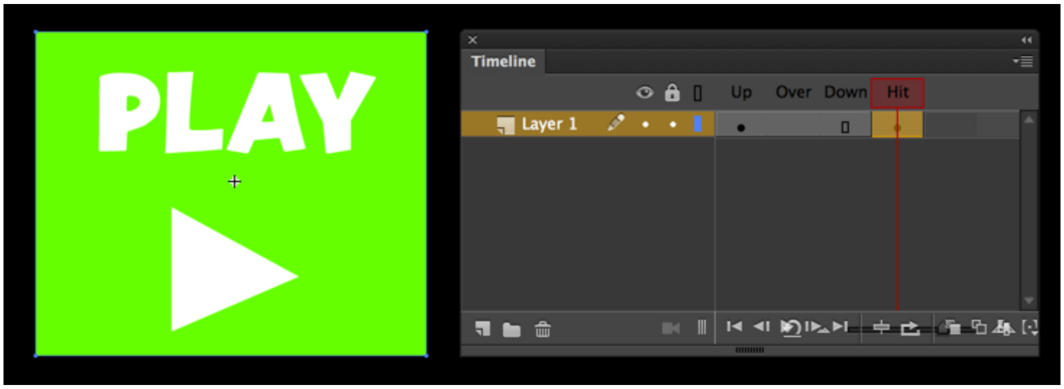
Name: PlayBtn

Type: Button

Folder: Library root

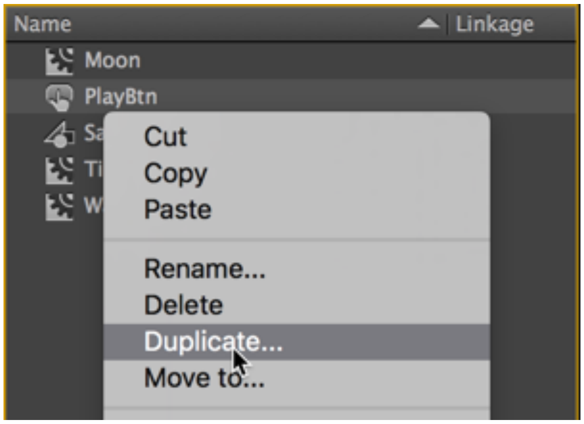
- Edit the new button by double-clicking upon the instance which exists on frame 1 of the UI layer.

- Button symbols in Animate CC have their own unique timeline consisting of various button states, including a Hit state which defines the clickable area. Select the Hit state of the symbol and choose Insert > Timeline > Keyframe to create a new keyframe and then draw a simple rectangle across the visual assets to create a solid hit state.

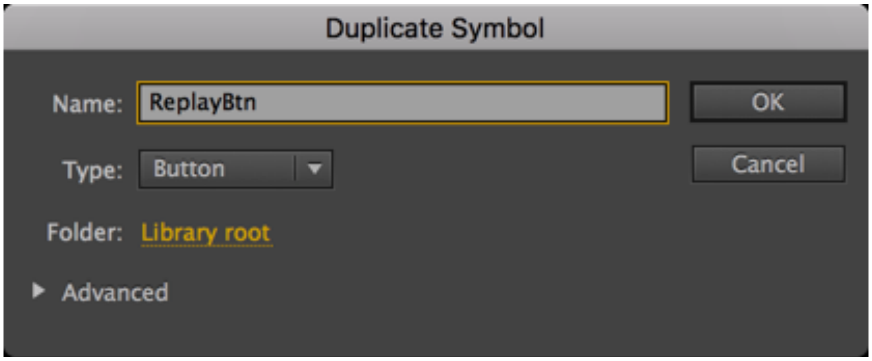


The Hit state is not visible to the user – so don't worry about how it appears here.

- With our play button created, let's duplicate it and change it a bit to create a replay button. Open the Library and right-click on the PlayBtn symbol. Choose Duplicate from the menu that appears.



- You will be presented with a dialog asking what to name the new symbol. Enter "ReplayBtn" and hit OK.

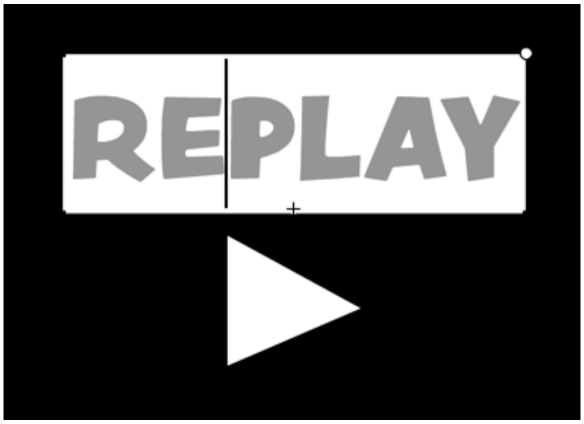

Duplicate Symbol

Name: ReplayBtn

Type: Button

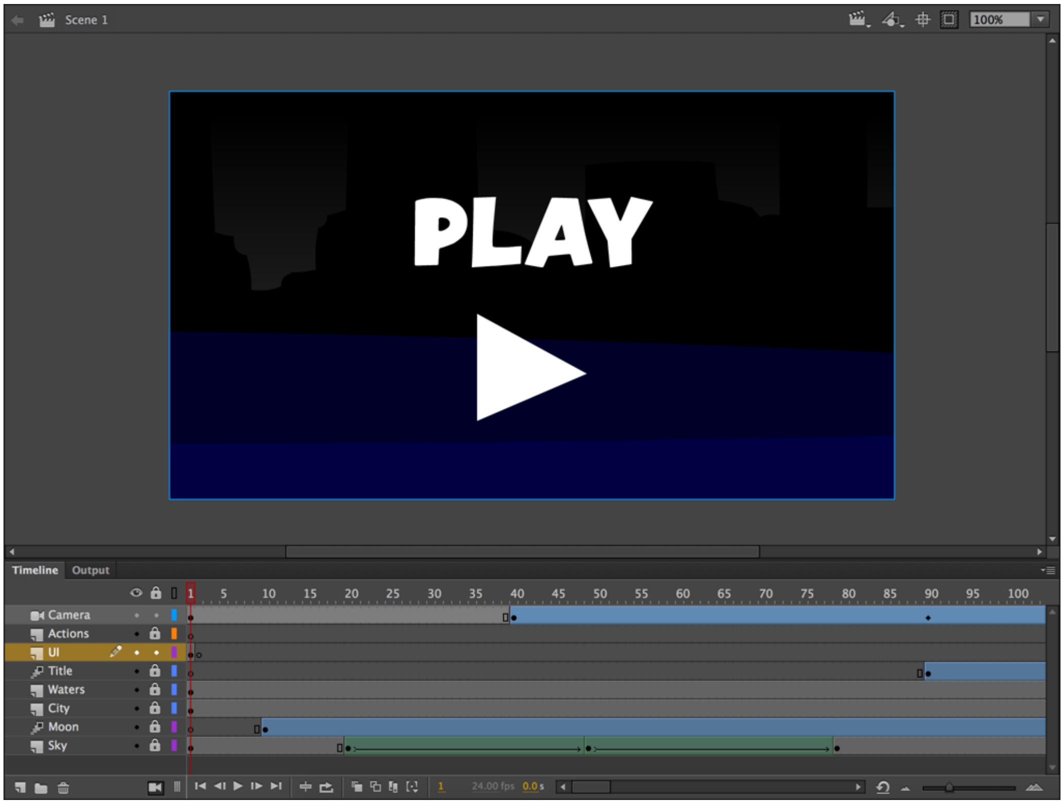
Folder: Library root

- You'll see the new symbol in the Library. Double-click upon it to edit the text. Change the text from "PLAY" to "REPLAY".



- You may also consider making the hit shape bigger as well – since the word "replay" takes up more room. You may also need to reposition elements if they are no longer as you'd like them to appear.
- Exit out of the symbol by clicking "Scene 1" above the stage. This brings us back to the main stage. Here, we can resize and reposition the PlayBtn instance on frame 1 until it appears just as we like. I've made mine pretty large and have centered it upon the screen.
- You will only want this button to appear on the first frame and then go away once someone clicks it. Select frame 2 of the UI layer and choose Insert > Timeline > Blank Keyframe from the application menu. This inserts a new, empty keyframe one frame 2 – ensuring that our play button only exists on

frame 1.

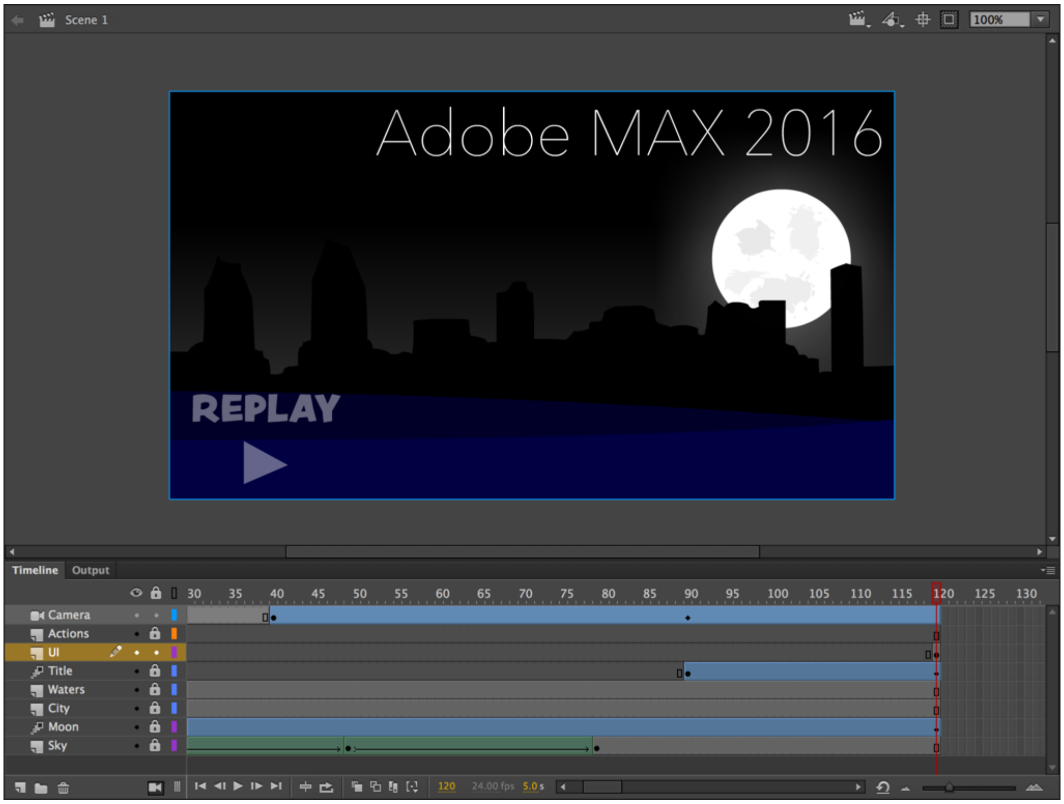


- Move the playhead forward to the final frame – frame 120. Insert another blank keyframe on this frame within the UI layer. This is where we will create an instance of our replay button from the library.

- Being sure that frame 120 is selected, locate ReplayBtn in the Library and drag an instance of this symbol onto the stage.

- You can then modify the size and location of the instance to suit your needs. I've made mine smallish and tucked it into the lower left corner of the stage. Additionally, I've brought its Alpha property down to 40% through the Color

Effect section of the **Properties** panel.

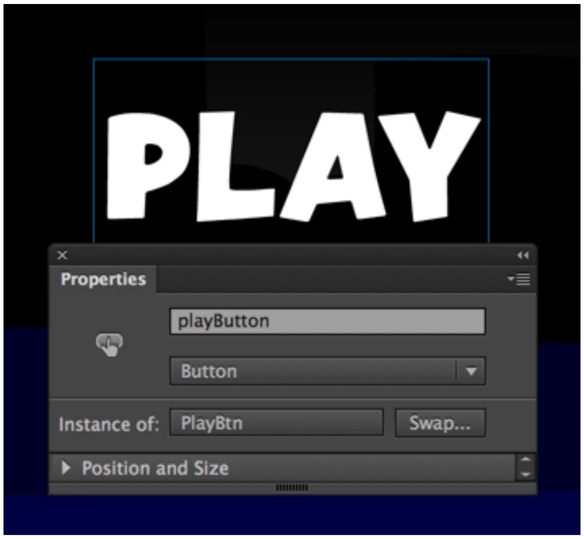


Play and Replay Functions

If we go ahead and test our animation by choosing **Control > Test** from the application menu, the animation will display in the browser but will be stopped at the first frame since we included a stop action on that frame.

The play button will be visible but clicking on it does nothing until we add a bit more JavaScript in order to provide interactivity through user interaction. Let's write some code to listen for and react to click events on these Button symbol instances.

- Select the PlayBtn instance on frame 1 and note that the Properties panel allows us to provide an **instance name** value. Enter "playButton". An instance name allows us to refer to this instance through JavaScript code.

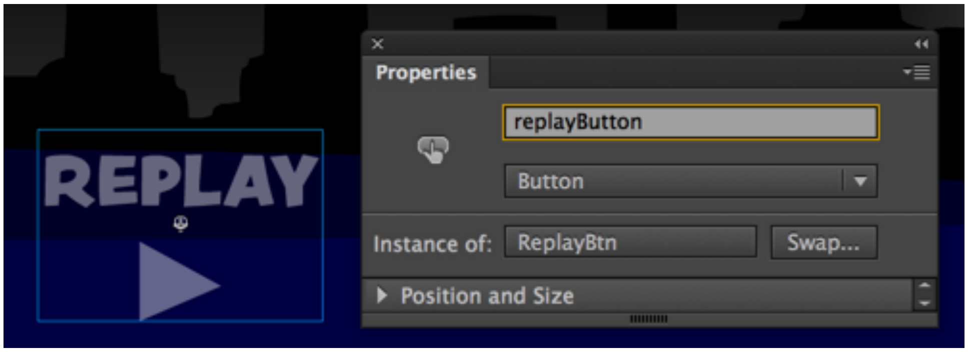


playButton

Button

Instance of: PlayBtn

- We are going to now do something very similar to our other button. Move the playhead to frame 120 and select the ReplayBtn instance.
- With the instance now selected, refer to the **Properties** panel once again and this time enter “replayButton” as the **instance name**. If you like, you can now lock down the UI layer.



replayButton

Button

Instance of: ReplayBtn

Now that each of our Button symbol instances has a unique instance name, the next thing we must do is write the **JavaScript** code to listen for and respond to user initiated events.

- If you’ve closed the **Actions** panel, you can open it once again by selecting **Window > Actions** from the application menu.

- Along the left side is displayed a list of the current frames that hold code – frame 1 and frame 120. Clicking on either of these will display the associated code within the script pane. Choose the entry for frame 1.

- You can see that we only have our stop action on this frame right now. Add an empty line or two below this by hitting enter, and type in the following:

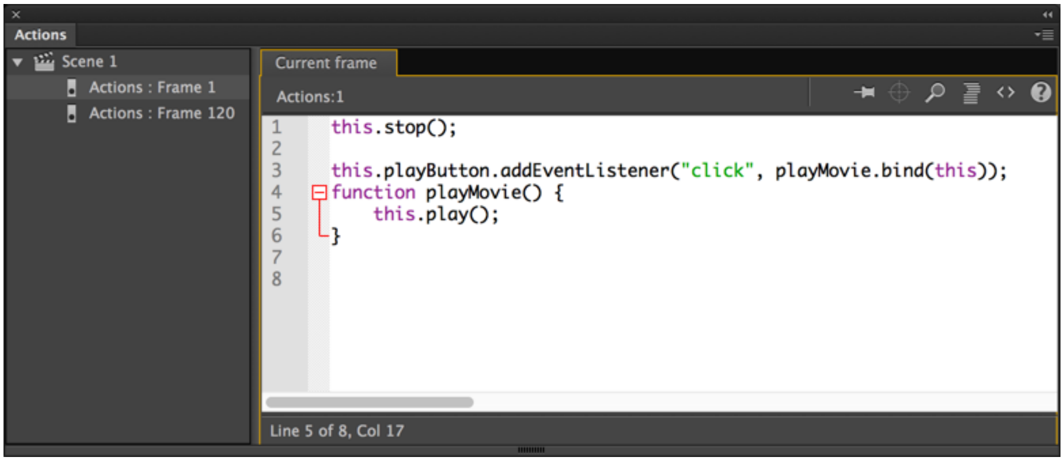
```
this.playButton.addEventListener("click",
playMovie.bind(this));

function playMovie() {

    this.play();

}
```

- So what does this code do? We refer to our button instance by instance name using this.playButton – the command, addEventLister tells the button instance to listen for a "click" event type and if an event is detected, to run a function named playMovie. That function is declared immediately after, and invokes the command this.play() – which tells the playhead to start playing the animation!



```
this.stop();

this.playButton.addEventListener("click", playMovie.bind(this));
function playMovie() {
    this.play();
}
```

Line 5 of 8, Col 17

"this" can refer to many different things in JavaScript, depending upon scope. When using "this" on the main timeline, outside of any function, it refers to the current scope and you can then refer to the instance names of objects on the stage as well.

- Now click in the entry for frame 120. Again, you should see only the stop action in this frame. Again, we will add in some code:

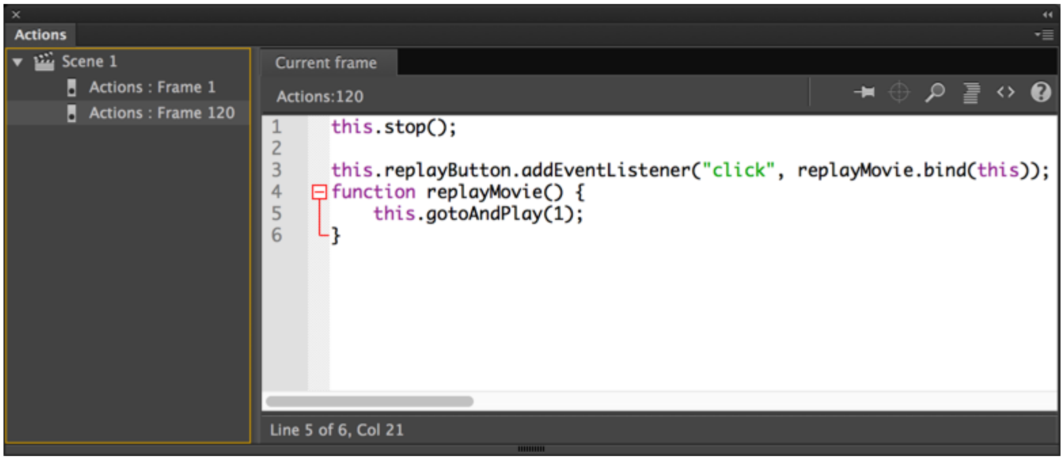
```
this.replayButton.addEventListener("click",
replayMovie.bind(this));

function replayMovie() {

    this.gotoAndPlay(1);

}
```

- The first line of code is basically the same as what we just saw, except it is targeting the instance of replayButton on frame 120. The replayMovie() function tells the playhead to go back to frame 1 (which in JavaScript is actually the second frame on the timeline) and begin playing from there – essentially replaying the animation.



```
this.stop();

this.replayButton.addEventListener("click", replayMovie.bind(this));
function replayMovie() {
	this.gotoAndPlay(1);
}
```

Curious about using .bind(this) on our event handler function calls? What this does is essentially bind the present scope through to the function when it is invoked. That way, “this” in our function body retains the scope of “this” on our root.

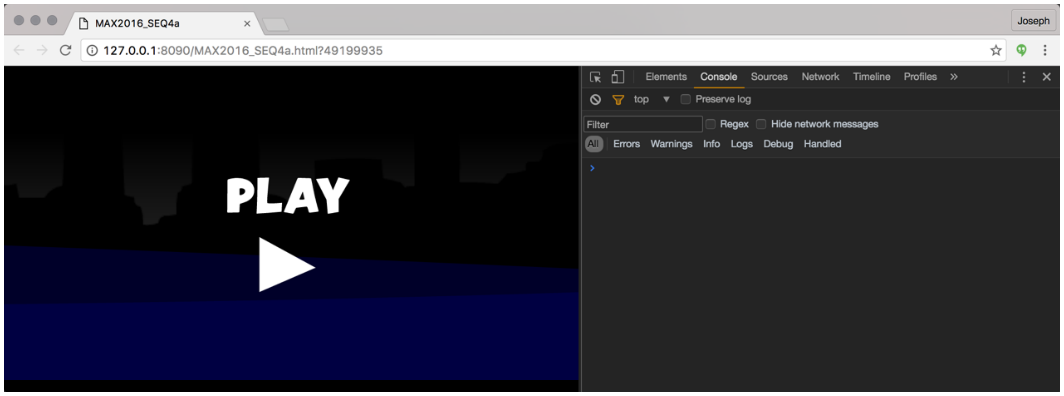
- With that, our code is complete and we have a fully interactive animation that will run in any modern web browser. Go ahead and test the movie – you can see how smoothly everything functions.

HTML5 Canvas Problems

Sometimes, if your interactive elements do nothing or your animation is not running correctly, it is because of JavaScript errors. To see whether your HTML5 Canvas project is throwing errors, you can refer to the browser console.

Let's look at how to access the console from Google Chrome. Many other browsers have similar features.

- Run your project by choosing Control > Test from the application menu.
- In the Chrome web browser, right-click anywhere within the viewport and choose Inspect.
- In the tooling area that appears, click on the Console tab. Any errors and warning and such will be displayed within.



Look! No errors!

SEQUENCE 4: Publishing and Exporting

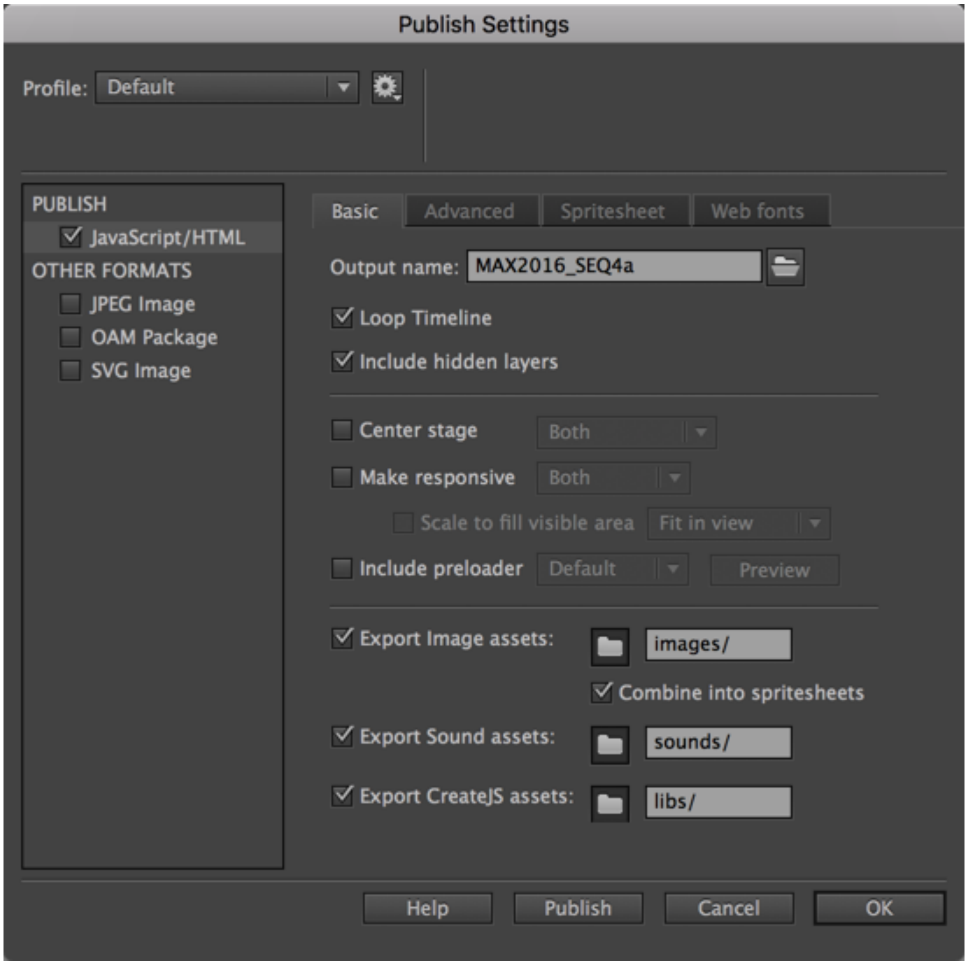
Our project is basically completed. We have a variety of files that can be published, exported, and converted to many different file types and platforms.

Okay! We can publish for playback! Let's look at some of our options.

HTML5 Canvas

Content created in Animate can now to be published directly to a number of modern HTML5 formats. The primary document type which allows this publish target is HTML5 Canvas and this is what we have used throughout this workbook.

When creating content using this publish target, Animate will leverage the CreateJS JavaScript libraries to output an entire animation or interactive project for the HTML5 canvas element. You can even write JavaScript in the Actions panel!



Publish Settings

Profile: Default

PUBLISH

☑ JavaScript/HTML

OTHER FORMATS

□ JPEG Image

□ OAM Package

□ SVG Image

Basic | Advanced | Spritesheet | Web fonts

Output name: MAX2016_SEQ4a

☑ Loop Timeline

☑ Include hidden layers

□ Center stage: Both

□ Make responsive: Both

□ Scale to fill visible area: Fit in view

□ Include preloader: Default | Preview

☑ Export Image assets: images/

☑ Combine into spritesheets

☑ Export Sound assets: sounds/

☑ Export CreateJS assets: libs/

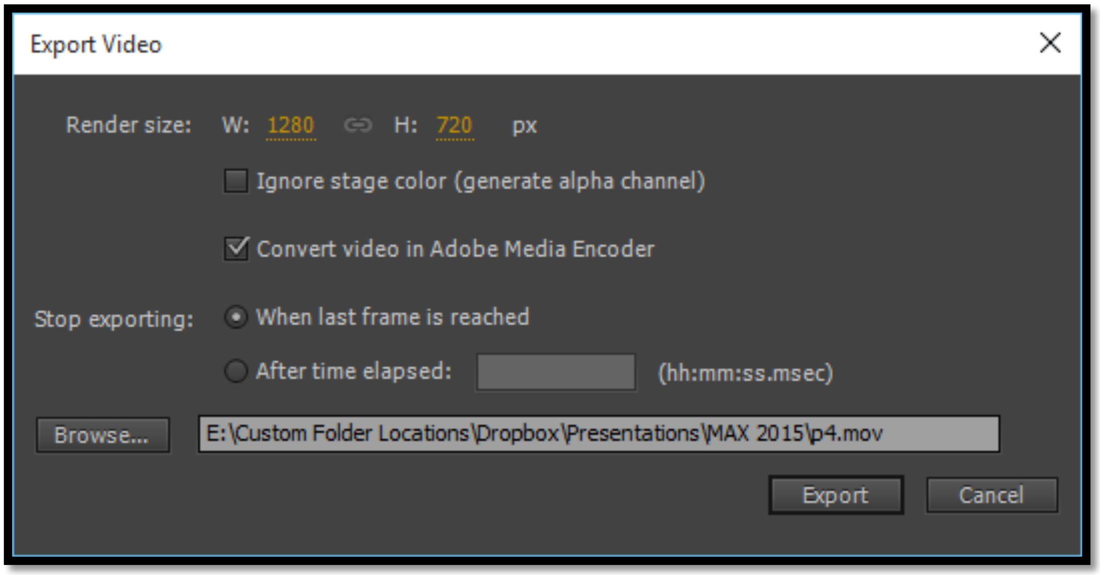
Publish Settings can be easily accessed from the Properties panel and include a LOT of different options. You can actually perform a true publish through the File > Publish option in the application menu.

HD Video

If we render our project to a web-friendly video format like MP4 – we don't have to worry about the limitations of certain platforms. Just make an awesome little video!

One of the major uses of Animate is to animate and produce content for television and video… so we are good! The fact that an ActionScript 3.0 based document type is generally the best choice for video simply due to the amount of extra tools and filters that are available should not be overlooked.

Choose File > Export > Export Video and then make some decisions around where to export the video and at what resolution. Animate does a frame by frame rendering of the content… so what ends up in the video is precisely what was authored. It even takes into account dynamic animation through the use of code!



Export Video

Render size: W: 1280 H: 720 px

□ Ignore stage color (generate alpha channel)

☑ Convert video in Adobe Media Encoder

Stop exporting: ● When last frame is reached

○ After time elapsed: (hh:mm:ss.msec)

E:\Custom Folder Locations\Dropbox\Presentations\MAX 2015\p4.mov

Once the video is rendered, we can fire up Adobe Media Encoder to render it to any format desired. You can also just import the video to an Adobe Premiere Pro or After Effects project.

Since video cannot be interactive, you'll want to remove the Actions and UI layers of our project before your render it as video.

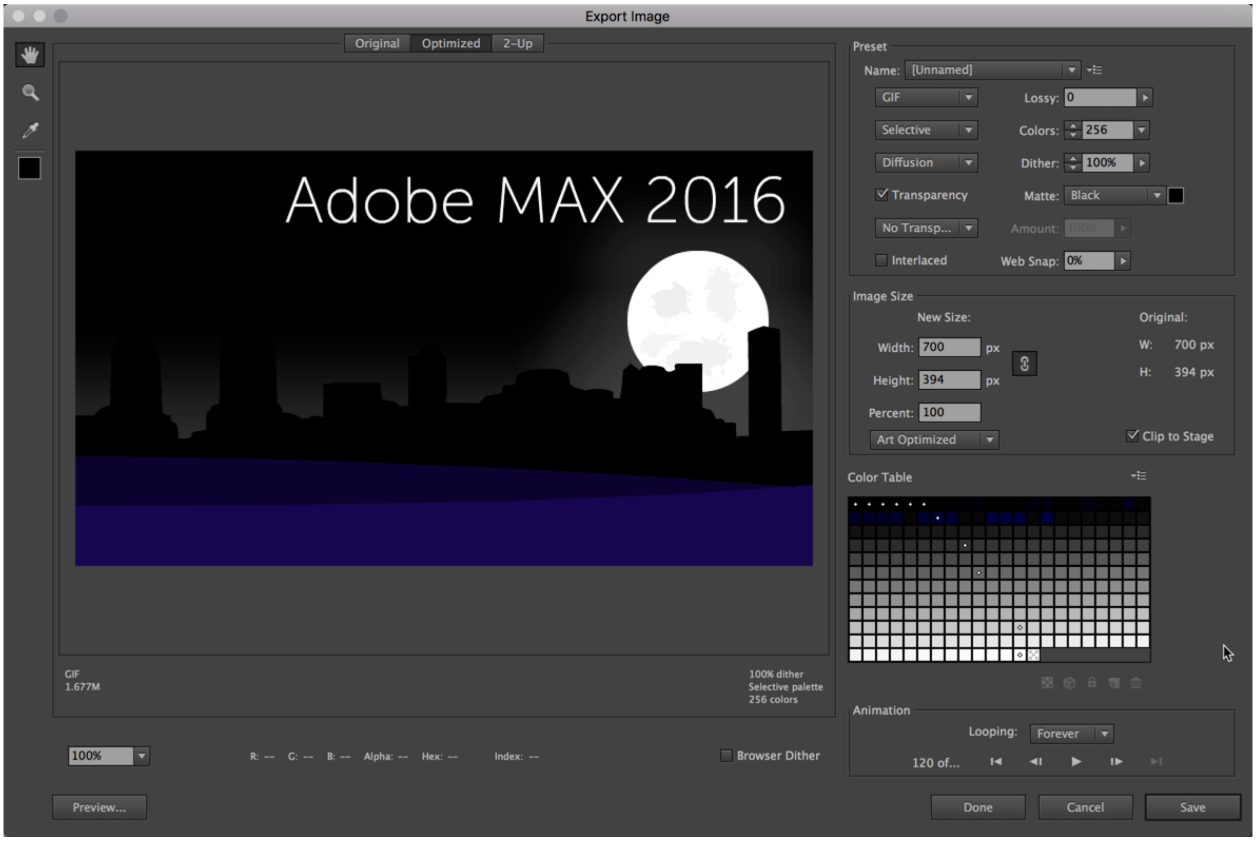
Animated GIF

Select File > Export > Export Animated GIF and a dialog will appear. You can apply individual preset, image, size, color, and animation settings to the optimized and 2-UP views and use the various preset options to set various parameters.

- Select Matte to modify and blend the background color across the edges of the asset.
- Select Transparent or non-transparent to work with the color picker and the transparent option to control the transparency of the asset.
- Set the Image size, color and animation options.

The file size values are displayed below the screen and clicking the Preview button allows you to preview the animation and navigate to a specific frame. Additionally, you may choose Select Clip to Stage in the image size panel to set the clipping boundary to the stage.

When ready, select Save to save the animated GIF!



Preset
Name: [Unnamed]
GIF — Lossy: 0
Selective — Colors: 256
Diffusion — Dither: 100%
☑ Transparency — Matte: Black
No Transp... — Amount: 100%
☐ Interlaced — Web Snap: 0%

Image Size
New Size: Width: 700 px, Height: 394 px, Percent: 100
Original: W: 700 px, H: 394 px
Art Optimized
☑ Clip to Stage

GIF
1.677M

100% dither
Selective palette
256 colors

100%
R: -- G: -- B: -- Alpha: -- Hex: -- Index: --
☐ Browser Dither

Animation
Looping: Forever
120 of...

When you save an optimized file using the Export Animated GIF option, you can generate an HTML file for the image. This file contains all the necessary information to display your image in a web browser.

Since a GIF file cannot be interactive, you'll want to remove the Actions and UI layers of our project before your render it as video.

Flash Player

Animate CC has deep roots in ActionScript and the Adobe Flash Runtimes: Flash Player and AIR. This should come as no surprise… for the majority of its lifetime – this was really the only supported platform!

As a result, choosing any of the AS3-based document types allows for the fullest set of creative tooling and the use of a deeply mature programming language – ActionScript 3.0.

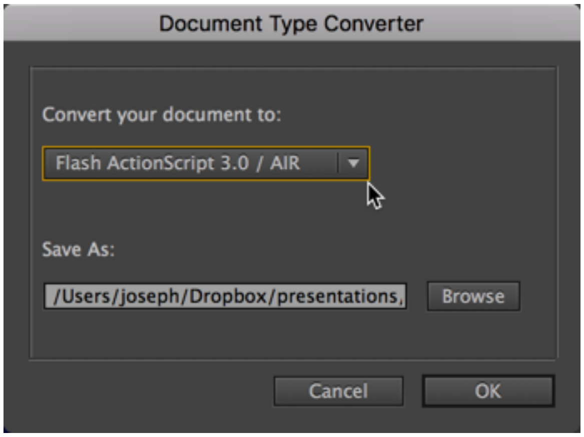
A great example from this workshop project is the Camera. Using the Camera in an ActionScript 3.0 document gives you access to Tints and Color Effects!

Here are some statistics from last year around Flash Player that may surprise you:

- The standalone Flash Player runtime is updated over 1 billion times per month from Adobe.
- Integrated Flash Player is built into Google Chrome, Internet Explorer 11, and Microsoft Edge.
- Preinstalled with Microsoft Windows 10.
- Every ChromeBook includes Flash Player. Huge for educational content.
- 9 out of 10 Facebook games runs on Flash Player.
- New builds for Linux!

- Add this all up – Flash Player is (still) everywhere!

To convert our HTML5 Canvas document to ActionScript 3.0, choose Commands > Convert to Other Document Formats from the application menu. In the dialog that appears, choose "Flash ActionScript 3.0 / AIR" as your new document type and hit OK.



Convert your document to: Flash ActionScript 3.0 / AIR

Save As: /Users/joseph/Dropbox/presentations

Once created, have a look at any warnings that appear in the Output panel. Any existing JavaScript will be commented out – and will need to be replaced with ActionScript, as Animate CC does not convert any code in the process.

WebGL

The secondary native HTML5 document type involves the use of WebGL (Web Graphics Library) technology. Interestingly enough, WebGL also makes use of the HTML5 canvas element… but in a different way.

WebGL content is actually GPU-accelerated, meaning that it makes direct use of the hardware GPU (Graphical Processing Unit) instead of sharing rendering tasks with everything else running on the CPU (Central Processing Unit). This makes for a much more effective and powerful rendering target.

While WebGL can be made interactive with JavaScript, the code syntax varies quite a bit from what we've seen using HTML5 Canvas.

You can convert to this document type the same way as detailed above.

Publishing for iOS and Android with AIR

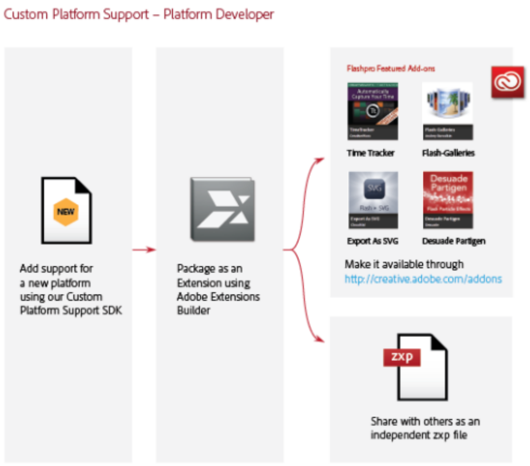
Additionally, we can publish directly to iOS and Android by leveraging Adobe AIR! AIR is hugely popular for building real apps across both desktop and mobile targeting Windows, macOS, Linux, iOS, and Android.

- AIR Application Installs:
  8 BILLION (double since November 2015!)
- Mobile AIR applications:
  500k+ (more than doubled since November 2015!)
- Soon to support both Android TV and Apple TV!

While targeting Flash Player, Video, Canvas, or WebGL… we could just convert the document and just tweak things here or there… creating a native mobile app is a bit of a different story. We'll probably want to rearrange the various elements on the Stage to accommodate the device resolution, as mobile apps are generally full screen experiences. Additionally, we'd most likely want to create a true app experience around such content, not have a simple animation be the entire app.

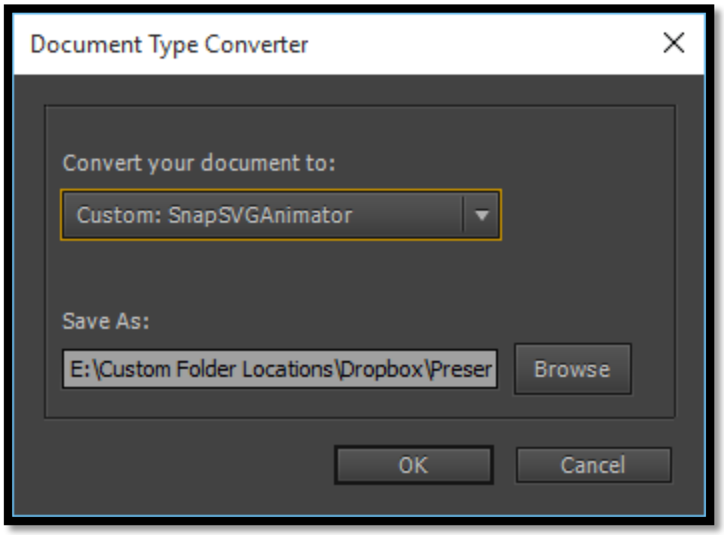
Regardless – Adobe AIR is a great target for desktop and mobile app development that you can consider when working in Animate CC!

Using Custom Publish Targets


Custom Platform Support – Platform Developer
NEW
Add support for a new platform using our Custom Platform Support SDK
Package as an Extension using Adobe Extensions Builder
Flashpro Featured Add-ons
Time Tracker
Flash-Galleries
Export As SVG
Desuade Partigen
Make it available through http://creative.adobe.com/addons
zxp
Share with others as an independent zxp file

As we have seen, Animate supports creating rich assets and animations using native document types such as ActionScript 3.0, HTML5 Canvas, and WebGL. The Custom Platform Support feature helps to extend the power of Animate CC to support platforms that are not native to Animate.

Using the Custom Platforms SDK, a developer can integrate his or her own platform as a real FLA based target. By installing a platform support plugin for a new document type, Animate users can create their assets using the rich feature set of Animate and publish it in the output format of the custom platform!



Document Type Converter

Convert your document to: Custom: SnapSVGAnimator

Save As: E:\Custom Folder Locations\Dropbox\Preser

Existing custom platform examples include:

- Snap.svg Animator

- GAF Publisher
- OpenFL
- Away3D

Additional Resources

To keep up on all things Animate, here are a few resources:

- **Adobe Animate Team Blog**
  http://blogs.adobe.com/animate/
- **Adobe AIR and Adobe Flash Player Team Blog**
  http://blogs.adobe.com/flashplayer/
- **Adobe Animate CC Learn and Support**
  https://helpx.adobe.com/animate.html

To contact Joseph Labrecque, note the following links:

- **Website**
  http://josephlabrecque.com/
- **Twitter Profile**
  https://twitter.com/JosephLabrecque
- **Blog - In Flagrante Delicto!**
  http://inflagrantedelicto.memoryspiral.com/
- **Lynda.com Author Page**
  http://www.lynda.com/JosephLabrecque

- **Amazon.com Author Profile**
  http://amazon.com/author/josephlabrecque

THANK YOU – HAVE FUN WITH ADOBE ANIMATE!