



Jennifer Gupta

eLearning JavaScript upgrades

JavaScript can be used with Storyline & Captivate to create some awesome content. Most of these are taken from tutorials by other people that I've tested and ensured that the code is localized. Then I just added directions and code to make it useful for our purposes. Below, I link to their pages rather than separating out each bit of code by who wrote it.

Storyline has some awesome articles & community projects on this subject for you to explore:

Check them out - [Storyline Article - Best Practices with JS](#) ---- [Community challenge](#)

Contributors: Code writers:

[NZ Malik](#)

[Richard from Bridgehill Learning Solutions](#)

[Learning Dojo](#)

Articulate Global

[Little Man Project](#)

[eLearning Locker](#)

Note: [Google Translate](#) and Chat GPT have their own courses.

☰ Custom Mute / UnMute button

☰ Alt Text to Tool Tips for Accessibility

☰ Turn off or modify player tooltips

☰ NotePad in Storyline

☰ Light / Dark Mode

☰ Dynamic caption font/size change

☰ Moving Captions

☰ Transparent background on Captions

☰ Add Confetti

OTHER RESOURCES (UNTESTED)

☰ More ideas with JavaScript (Storyline)

☰ GreenSock

☰ Ideas with JavaScript (Captivate)

☰ FlexPerts

☰ Learning Javascript

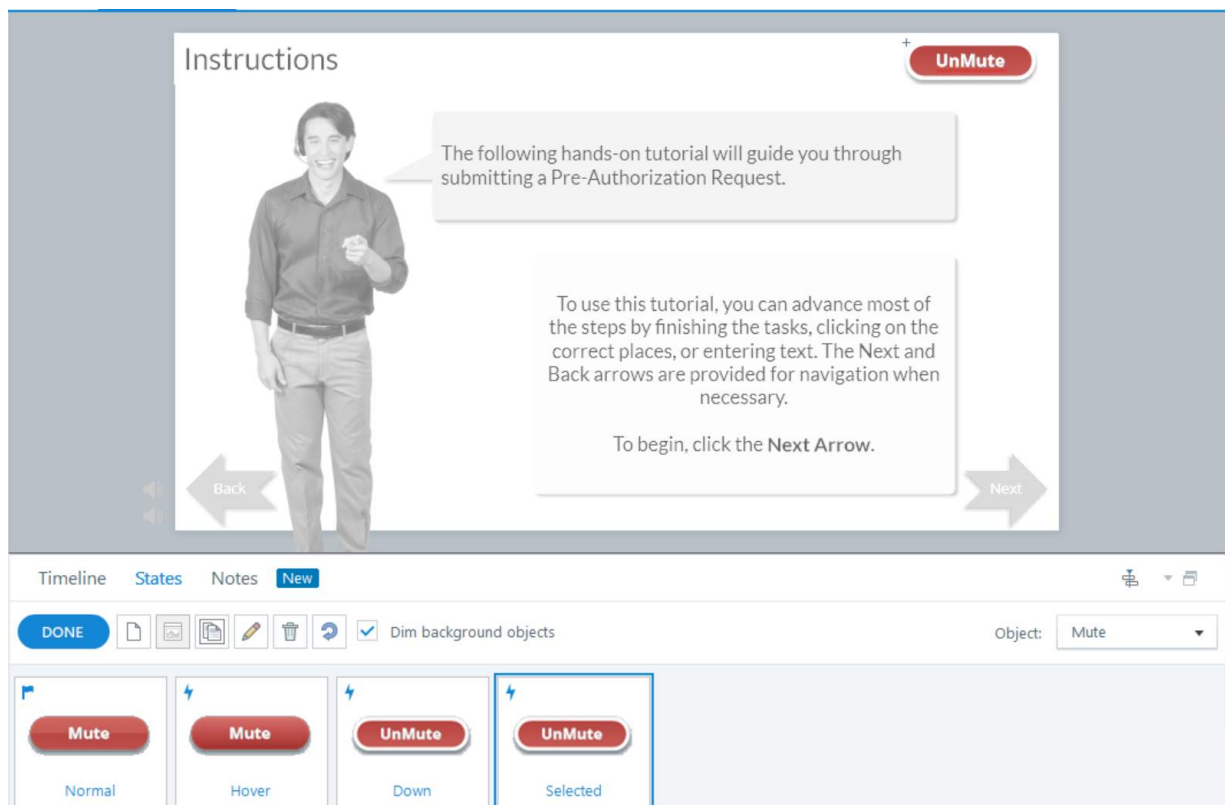
Custom Mute / UnMute button



Jennifer Gupta

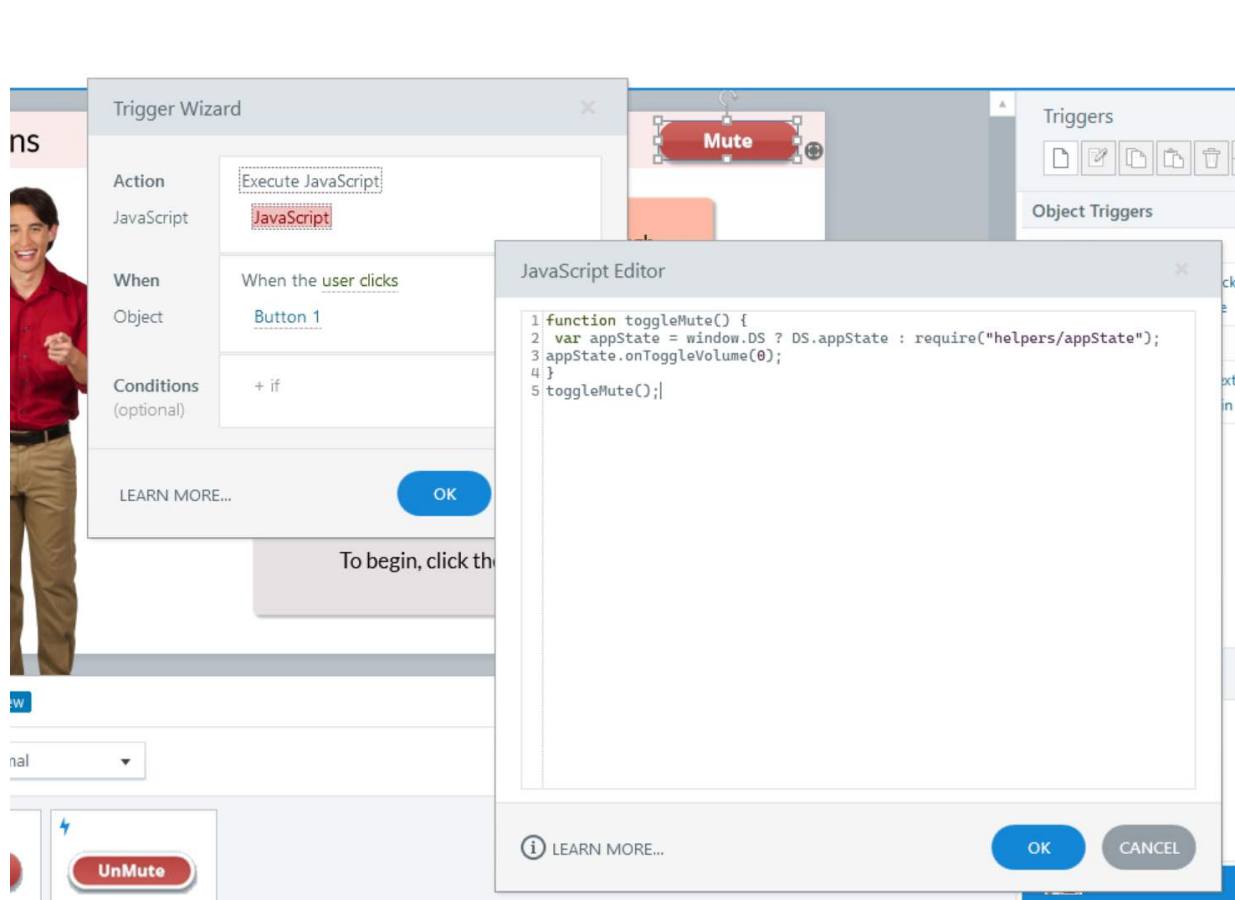
Add a Custom Mute/Unmute Button to Your Articulate Storyline Courses!

Ever need a quick way to have a learner mute/ un mute audio in your course? Use some JS to create this easy mute/unmute button.



Create your button

Ensure that it has a selected state.



Add your trigger

Create a trigger to execute JavaScript when the user clicks your button. Copy in the code from below.

```
function toggleMute() {  
  var appState = window.DS ? DS.appState : require("helpers/appState");  
  appState.onToggleVolume(0);  
}
```

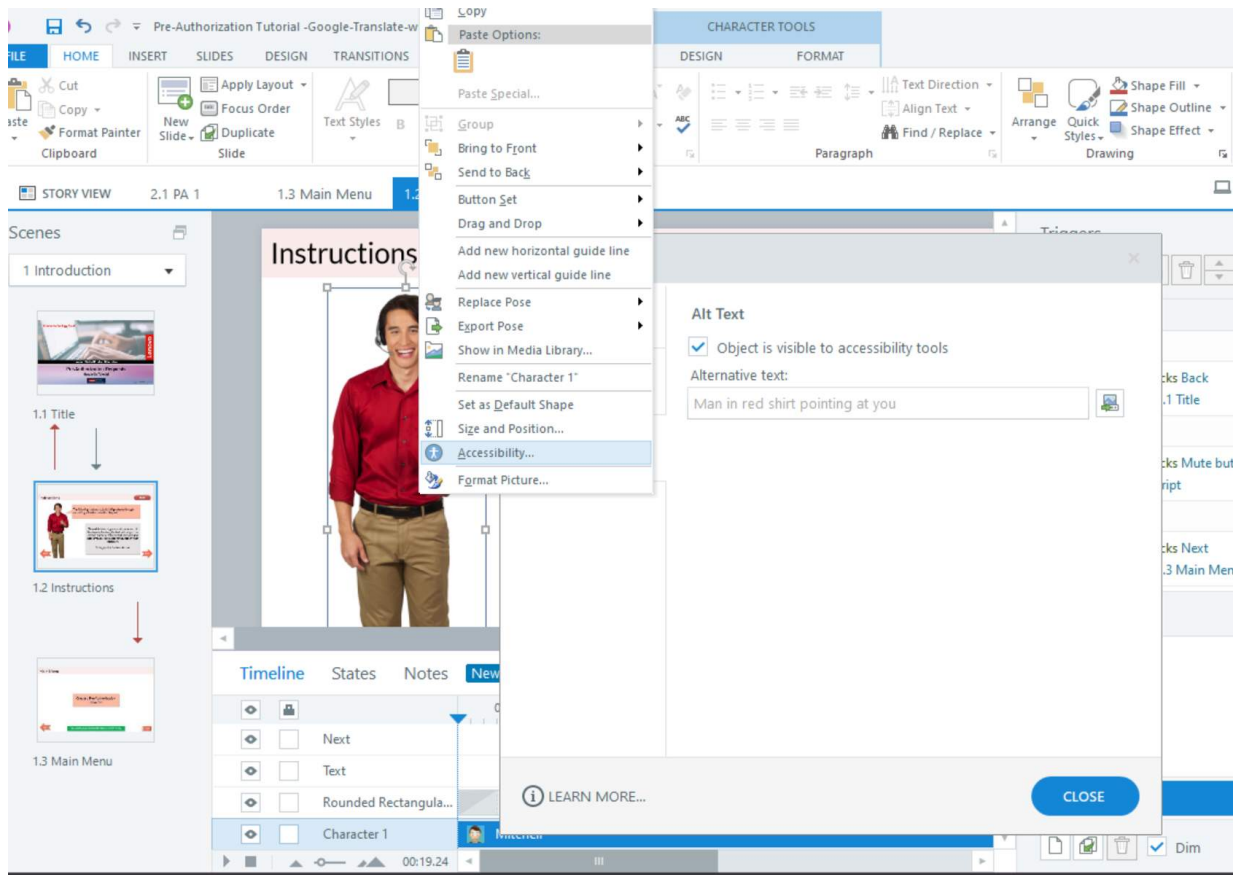
```
}  
toggleMute();
```

Alt Text to Tool Tips for Accessibility



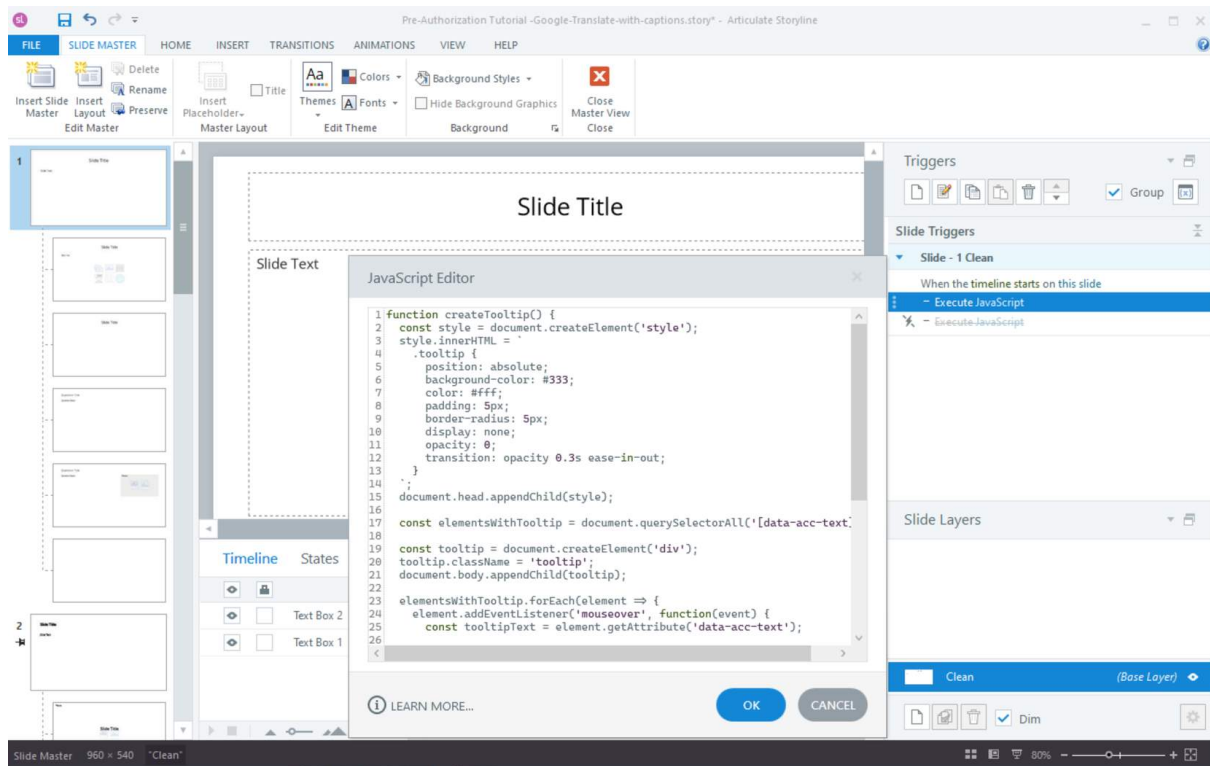
On the Master Slide

This solution uses the Accessibility Alt Text tags and turns them into tooltips. For each things you want to have a tooltip on - right click on it, click the Accessibility option and then enable the Alternative Text.



Finding a way

Where the road leads, I will go. Along the stark desert, across the wide plains, into the deep forests I will follow the call of the world and embrace its ferocious beauty.



Then, Add the below code on the Master Slide. Execute upon timeline start.

```
function createTooltip() {
  const style = document.createElement('style');
  style.innerHTML = `
    .tooltip {
      position: absolute;
      background-color: #333;
      color: #fff;
      padding: 5px;
      border-radius: 5px;
      display: none;
      opacity: 0;
      transition: opacity 0.3s ease-in-out;
    }
  `;
  document.head.appendChild(style);

  const elementsWithTooltip = document.querySelectorAll('[data-acc-text]');

  const tooltip = document.createElement('div');
```

```
tooltip.className = 'tooltip';
document.body.appendChild(tooltip);

elementsWithTooltip.forEach(element => {
  element.addEventListener('mouseover', function(event) {
    const tooltipText = element.getAttribute('data-acc-text');

    tooltip.style.left = (event.clientX + 10) + 'px';
    tooltip.style.top = (event.clientY + 10) + 'px';

    tooltip.textContent = tooltipText;

    tooltip.style.opacity = 1;
    tooltip.style.display = 'block';
  });

  element.addEventListener('mouseout', function() {
    tooltip.style.opacity = 0;
    setTimeout(() => {
      tooltip.style.display = 'none';
    }, 300);
  });
});

createTooltip();
```

Pre-Authorization Request Tutorial-w-captions

Instructions

Mute



The following hands-on tutorial will guide you through submitting a Pre-Authorization Request.

Man in red shirt pointing at you

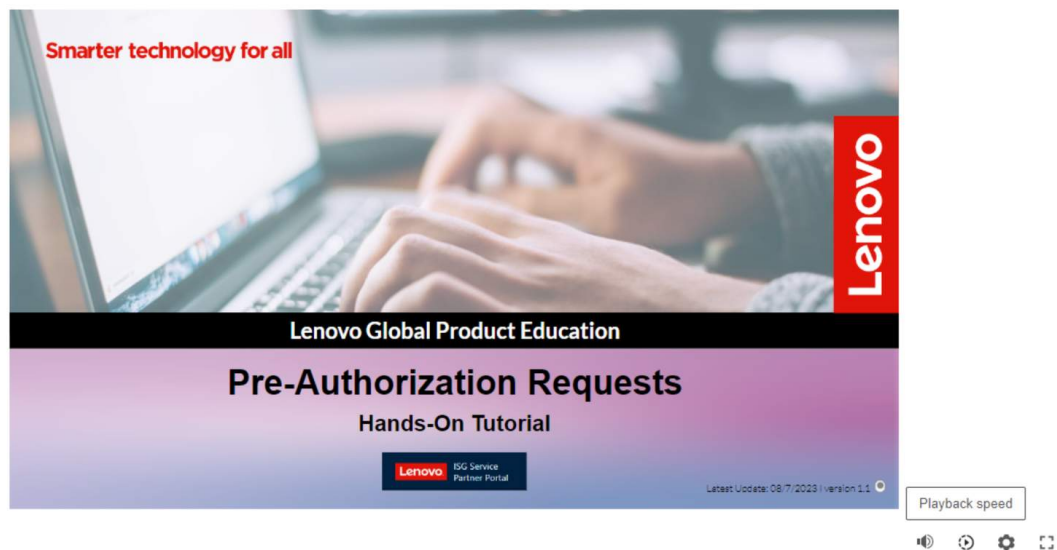
The Alternate Text (Alt Txt) now appears when the learners over their mouse over the object.

Turn off or modify player tooltips



Jennifer Gupta

Pre-Authorization Request Tutorial-w-captions



Player Tooltips

Alas, they come in 1 variety.. On and boxed in. With some JavaScript, we can modify them, or Turn them off entirely!

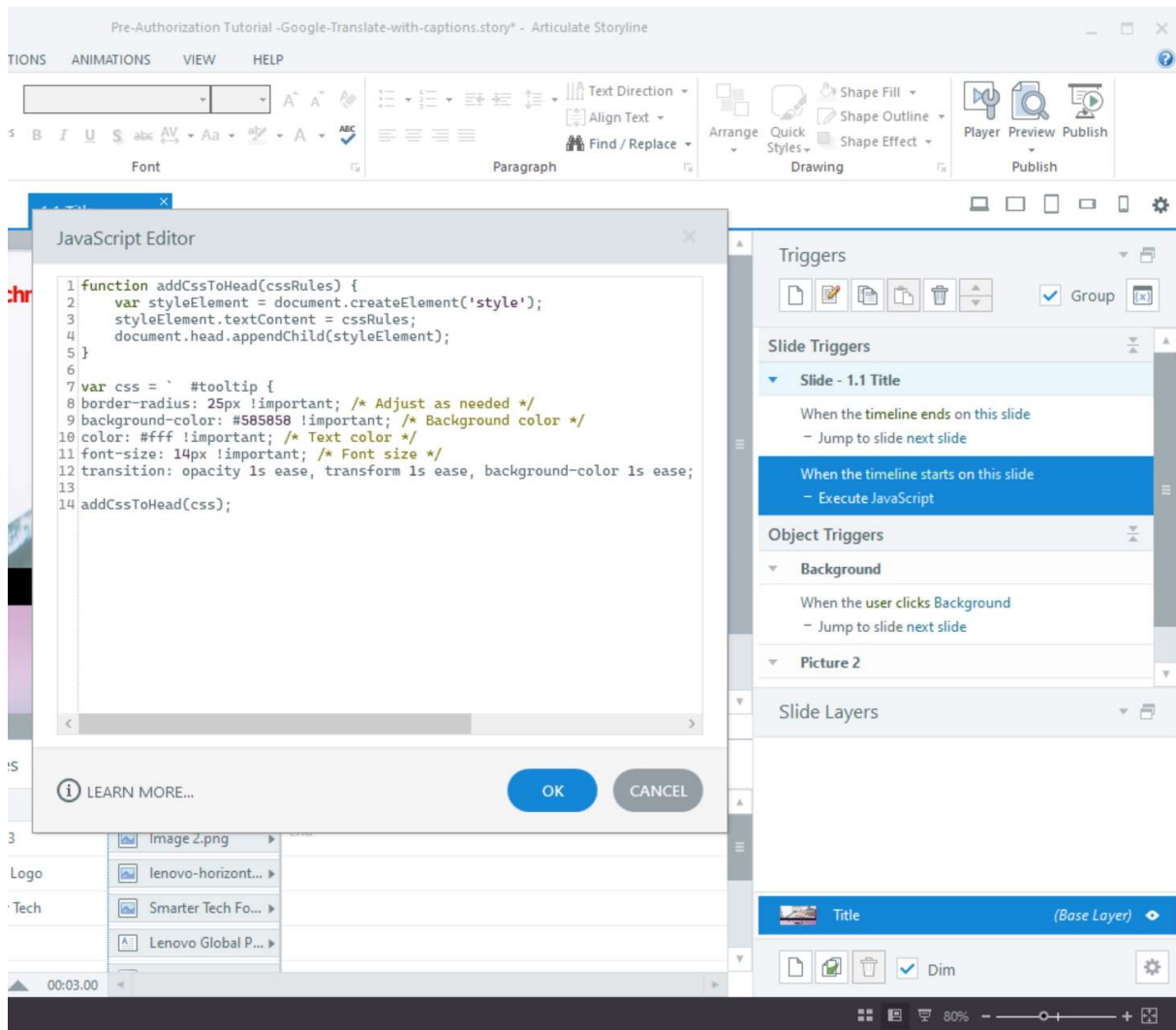
Lets start with turning them off. On your first slide - go ahead and create a trigger that executes JavaScript when the timeline starts. Then, paste the code below into it.

```
function addCssToHead(cssRules) {  
    var styleElement = document.createElement('style');  
    styleElement.textContent = cssRules;  
    document.head.appendChild(styleElement);  
}  
  
var css = ` #tooltip{  
    opacity: 0 !important;  
} `;  
addCssToHead(css);
```



Modify

Now, you can choose to modify them instead. Perhaps making them brand compatible. Let's see how this is done.



Create trigger

Create your trigger: Execute JavaScript when the timeline starts on this slide on the first slide of your Storyline file.

```
function addCssToHead(cssRules) {
    var styleElement = document.createElement('style');
    styleElement.textContent = cssRules;
    document.head.appendChild(styleElement);
}

var css = ` #tooltip {
border-radius: 25px !important; /* Adjust as needed */
```

```
background-color: #585858 !important; /* Background color */
color: #fff !important; /* Text color */
font-size: 14px !important; /* Font size */
transition: opacity 1s ease, transform 1s ease, background-color 1s ease;

addCssToHead(css);
```

Understand the Code

The above code has a few places you can change: I have made them bold and red so that they stand out. Change these things in your code and then copy it into the JS trigger.

```
border-radius: 25px !important; /* Adjust as needed */
```

```
background-color: #585858 !important; /* Background color */
```

```
color: #fff !important; /* Text color */
```

```
font-size: 14px !important; /* Font size */
```

NotePad in Storyline



Add a Notepad to your courses

Now you can easily add in a Notepad to your courses and let your learners take notes!
Notes are downloadable in many formats.

I'll show you 2 ways to do this.

The first method is super easy, Step 1: Add this .story file (already setup for you) Feel free to modify the Notepad.story file to suit your color and branding needs.










Notepad.story

158.1 KB





Triggers

      Group 

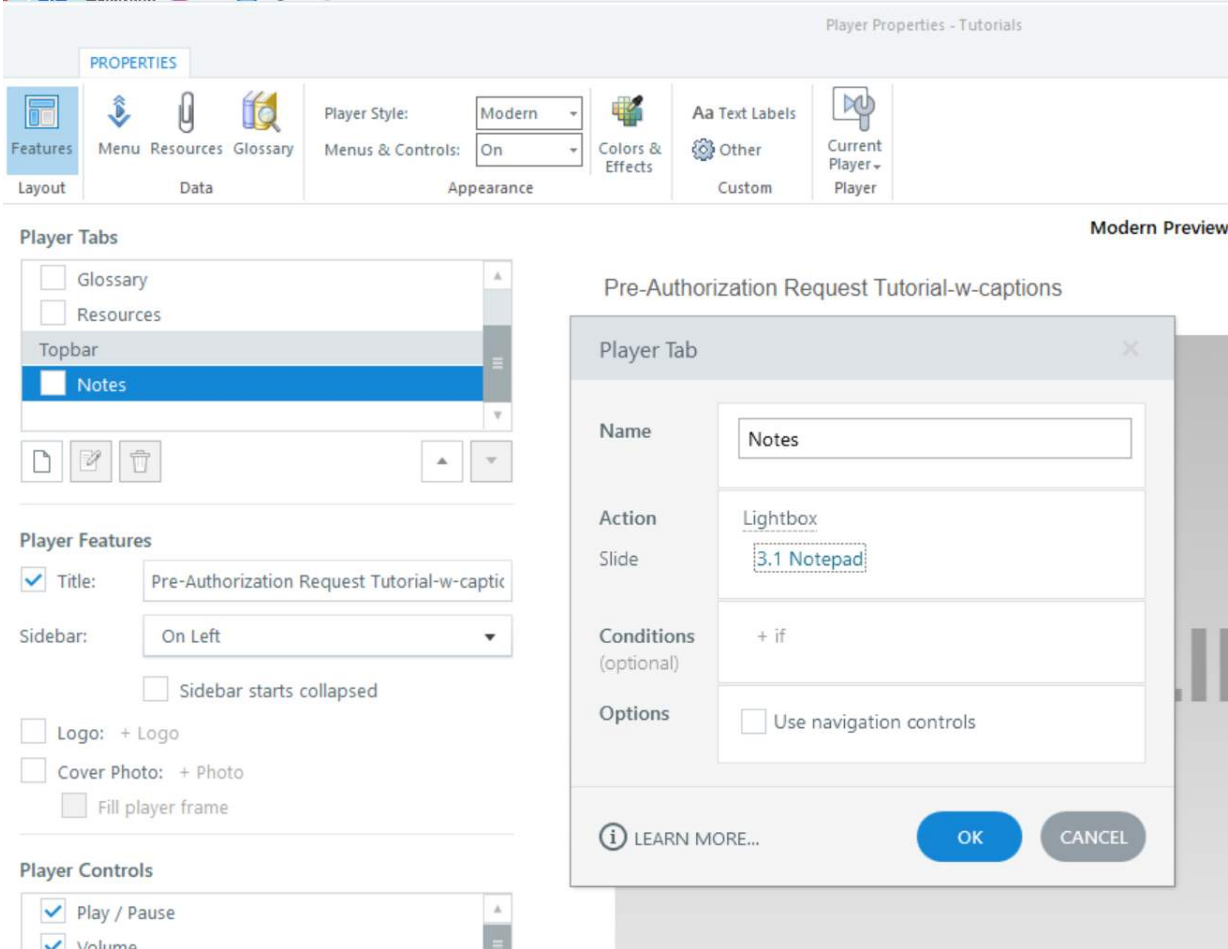
- ▼ **Slide - 3.1 Notepad**
 - When the **timeline starts** on this slide
 - Execute JavaScript
- Object Triggers**
- ▼ **ClearContentButton**
 - When the **user clicks** ClearContentButton
 - Execute JavaScript
- ▼ **DocSaveButton**
 - When the **user clicks** DocSaveButton
 - Execute JavaScript
- ▼ **PdfSaveButton**
 - When the **user clicks** PdfSaveButton
 - Execute JavaScript
- ▼ **Text Entry**
 - When Text Entry loses focus
 - Set **Title** equal to the typed value
- ▼ **Text Entry 1**
 - When Text Entry 1 loses focus
 - Set **Editor** equal to the typed value
- ▼ **TxtSaveButton**
 - When the **user clicks** TxtSaveButton

Slide Layers

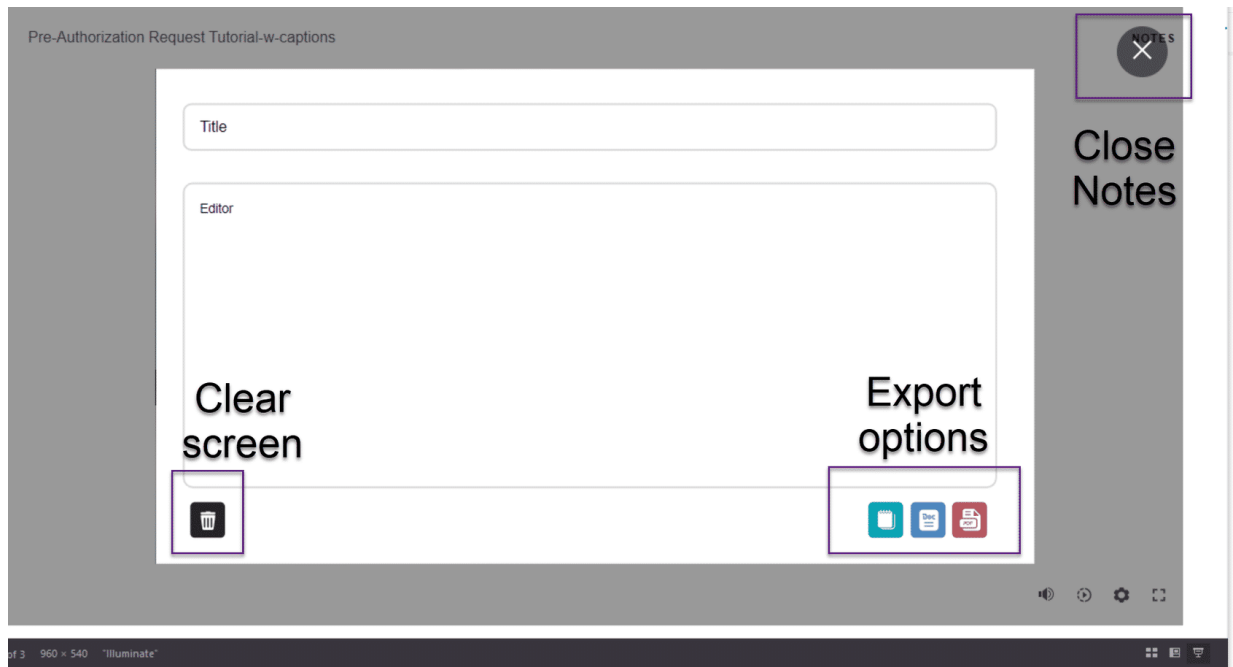
 

Windows taskbar: 80% zoom, window management icons, and system tray icons.

Don't mess with the JavaScript triggers already setup for you. This will break the code and make the notepad functions not work.

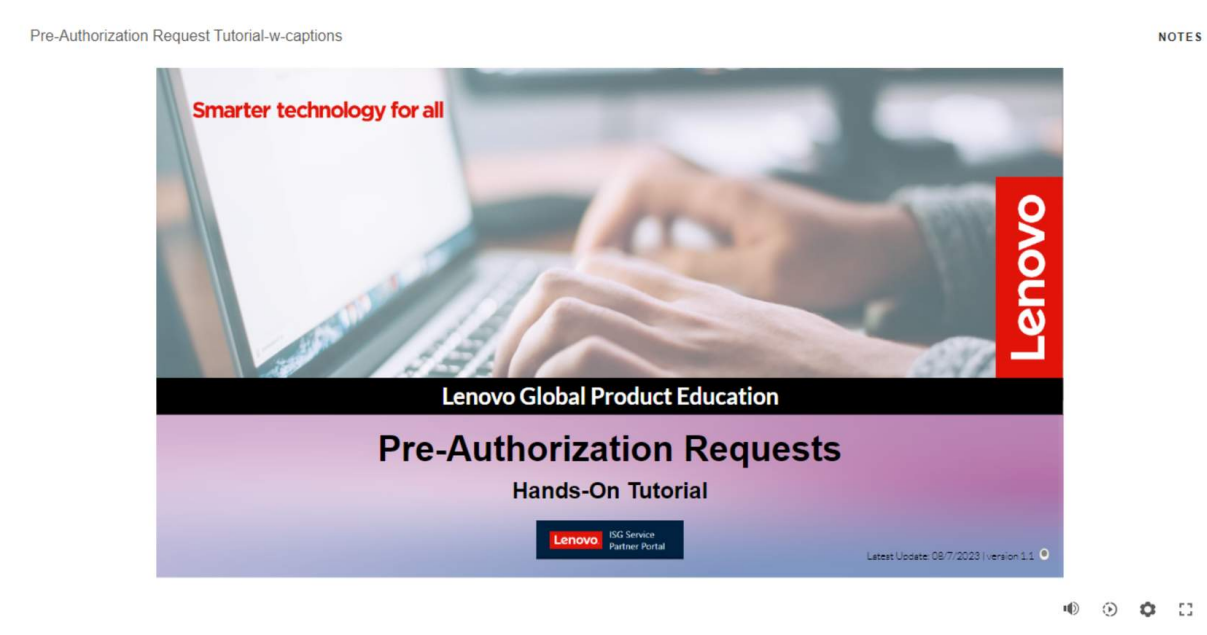


In your player - setup a new tab and lightbox the imported slide.



There are a few export options pre-programmed in. Export to PDF, Text file or Doc file.

A clear screen button is provided, and as always, you can use the X button to close the lightbox.



The notepad will now appear so that all learners have it on hand.

Light / Dark Mode



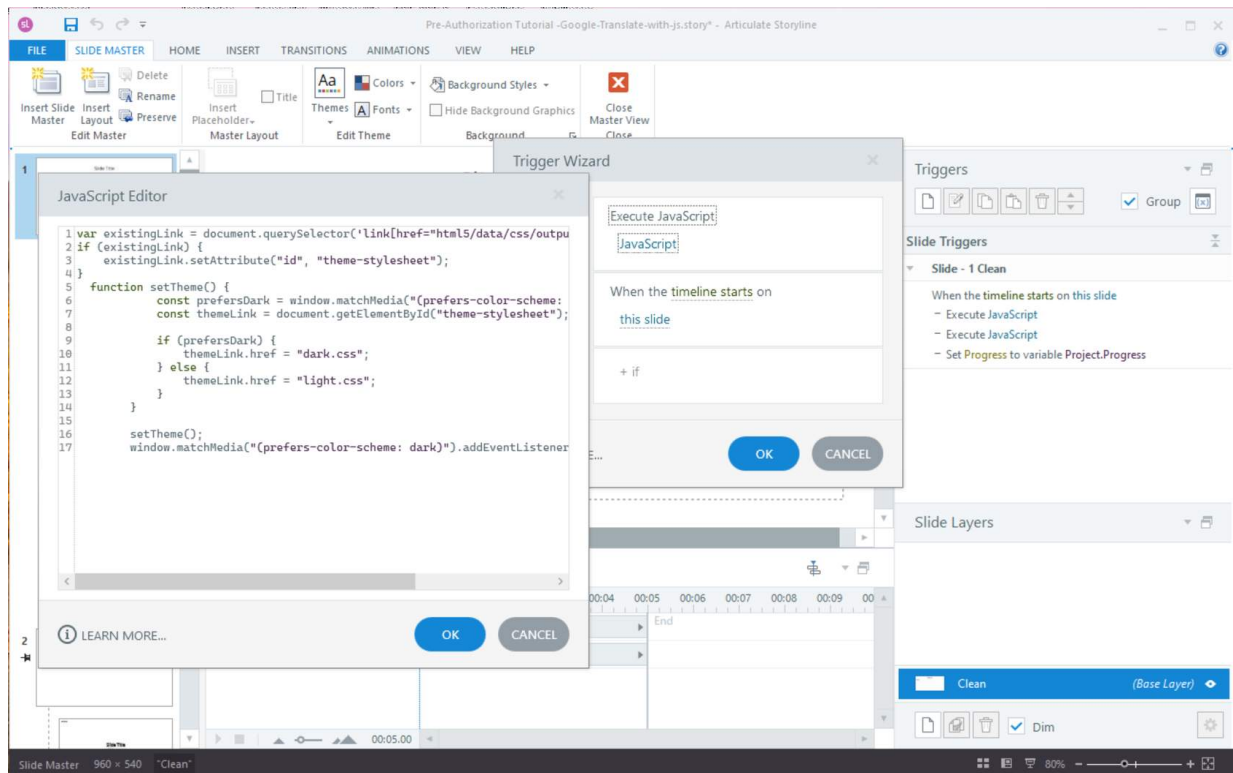
We can create Light/ Dark modes for our learners!

This is a bit more labor intensive the first time you set this up, BUT you can do it for each template, and then the implementation is quite simple.

There are a few parts to this solution. They are:

- 1) JS on the Master Slide
- 2) Create Player Tab & assign a JS to it.
- 3) Create and publish (to web) your light mode content. Rename & save the css file.
- 4) Create and publish (to web) your dark mode content. Rename & save the css file.
- 5) Publish your file as you normally would. Extract zip file, copy in your light and dark .css files to the correct place.
- 6) Rezip & share.

Once you have done this for the first time on any template you can skip steps 3 & 4 and just copy in your previously saved dark & light .css files.



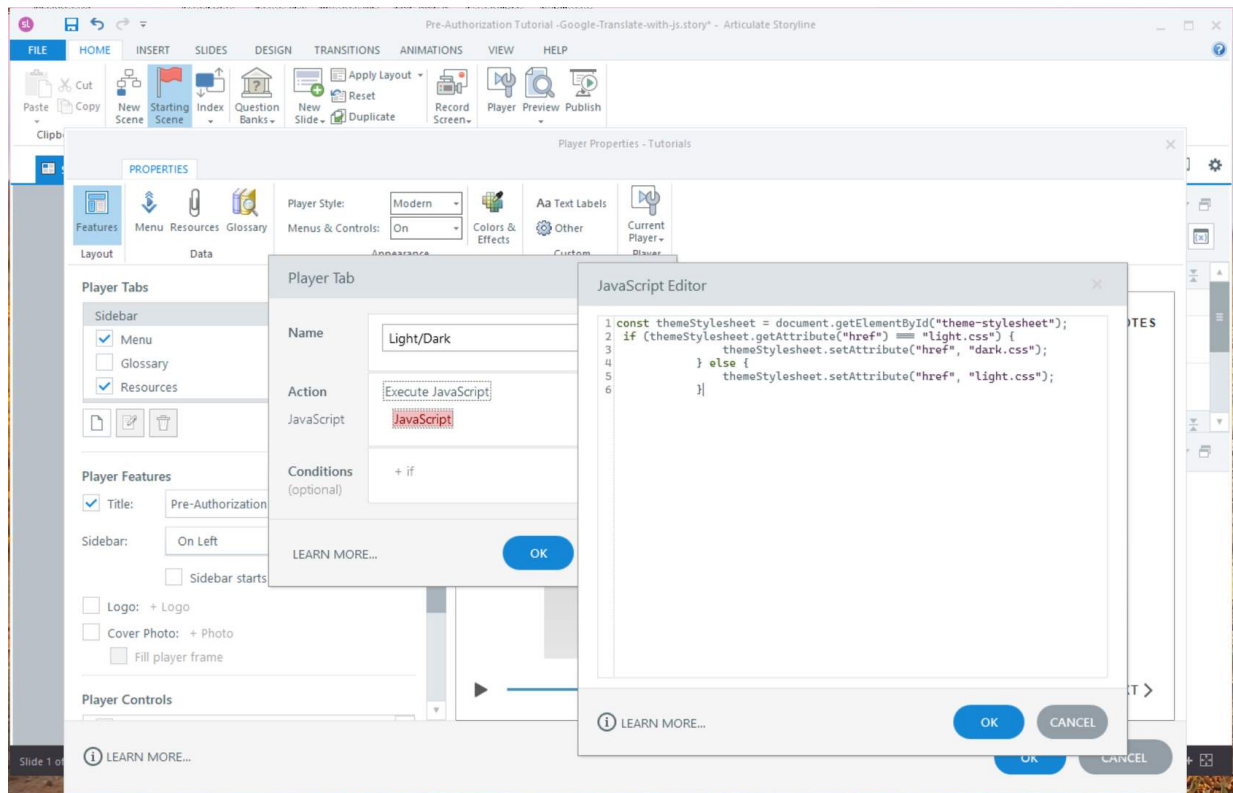
Step 1: Setup a JavaScript trigger that executes when the timeline starts on the MASTER SLIDE:

Copy in the code shown below:

```
var existingLink = document.querySelector('link[href="html5/data/css/output
if (existingLink) {
    existingLink.setAttribute("id", "theme-stylesheet");
}
function setTheme() {
    const prefersDark = window.matchMedia("(prefers-color-scheme: <
    const themeLink = document.getElementById("theme-stylesheet");

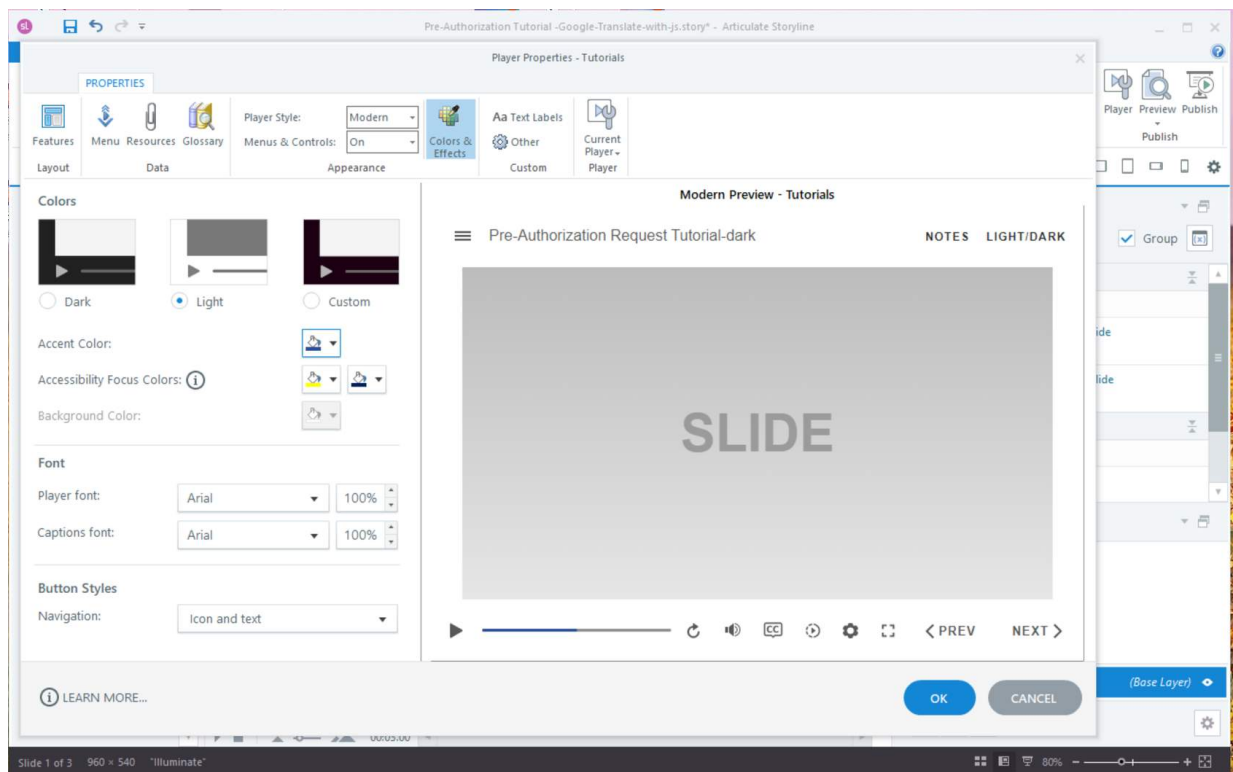
    if (prefersDark) {
        themeLink.href = "dark.css";
    } else {
        themeLink.href = "light.css";
    }
}
```

```
setTheme();  
window.matchMedia("(prefers-color-scheme: dark)").addEventListener
```



Step 2: Open your player. Create a new tab (I called mine Light/Dark. Execute a JavaScript code when it's clicked. (Paste the code from below)

```
const themeStylesheet = document.getElementById("theme-stylesheet");  
if (themeStylesheet.getAttribute("href") === "light.css") {  
  themeStylesheet.setAttribute("href", "dark.css");  
} else {  
  themeStylesheet.setAttribute("href", "light.css");  
}
```

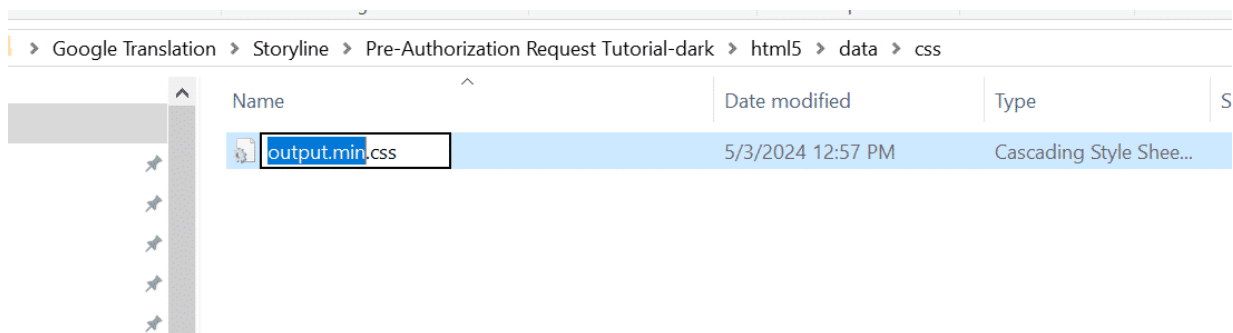


Step 3: Open your Player. Select Colors & Effects

Select the Light radio button

Setup your colors/fonts, etc.

Click OK to save it.

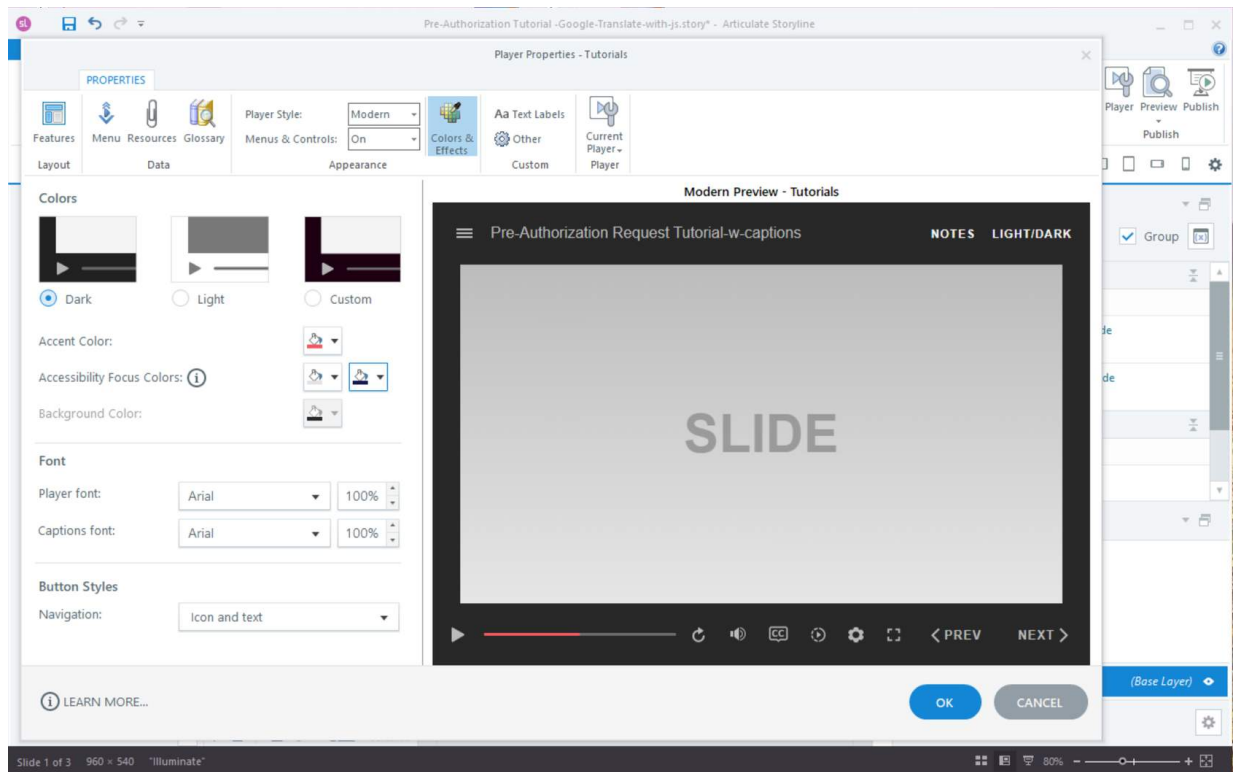


Publish your course as a web file

Go into the folder structure.

Get into the html5/data/css folder

Rename output.min.css to light.css

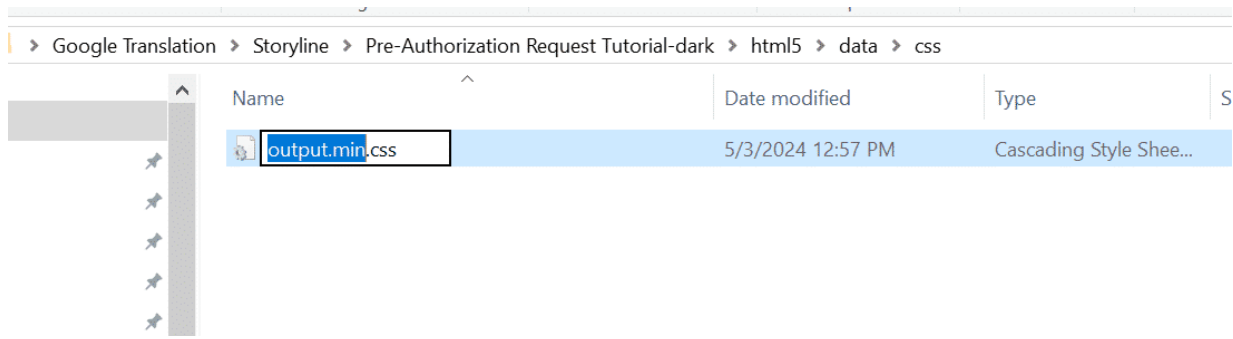


Step 4: Open your Player. Select Colors & Effects

Select the Dark radio button

Setup your colors/fonts, etc.

Click OK to save it.



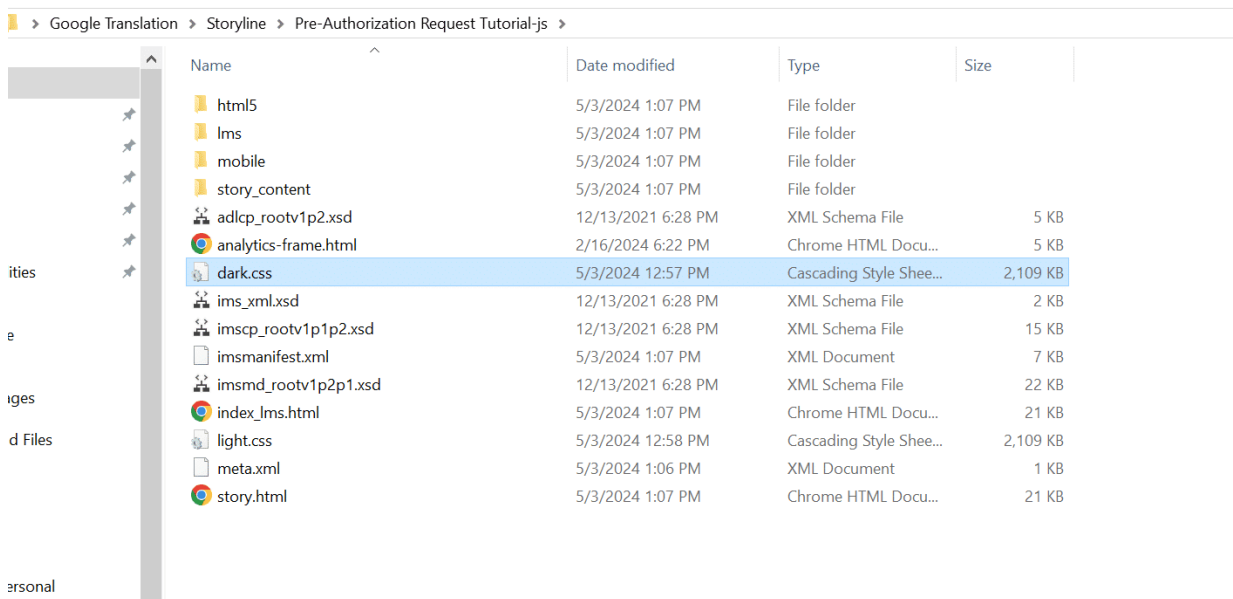
Publish your course as a web file

Go into the folder structure.

Get into the html5/data/css folder

Rename output.min.css to dark.css

Publish your whole course as normal

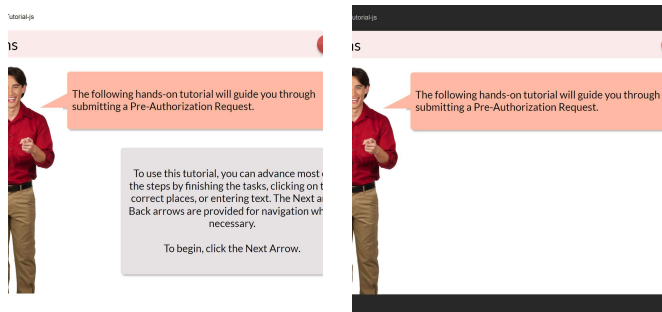


Now. In your main directory, where your story.html file is, copy in BOTH your dark.css and your light.css files.

Test!

Double click your story.html file and make sure it works!

When all is well - Zip up the directory and deploy!



Dynamic caption font/size change




Jennifer Gupta

What if you want to give your learners options to control the captions on their own? Enter some nifty JavaScript code and you can give it to them!

Nayraqat Autorizaci3n mayiwi Yatichawi-w-captions

Amuyt'ayawinaka




Aka ampamamp lurat yatichawix m3 Pre-Autorizaci3n Solicitud uñt'ayañ tuqitw irpapx3tam.

Aka yatichawi apnaqañatakixa, jilpacha lurawinakaruxa nayraru sartayañatakixa lurawinaka tukuyasa, chiqapa chiqanaku ch'iqt'aña, jan ukaxa qillqaña qillqt'aña. Jutir ukat Qhipax flechanakax wakiski ukhax sarañatakix churatawa.

Qalltañatakixa, **Jutir Flecha** siski ukaruw ch'iqt'añama .

- Aymara
- Afrikaans
- Albanian
- Amharic
- Arabic
- Armenian
- Assamese
- Aymara
- Azerbaijani
- Bambara**
- Basque
- Belarusian
- Bengali
- Bhojpuri
- Bosnian
- Bulgarian
- Catalan
- Cebuano
- Chichewa
- Chinese (Simplified)
- Chinese (Traditional)

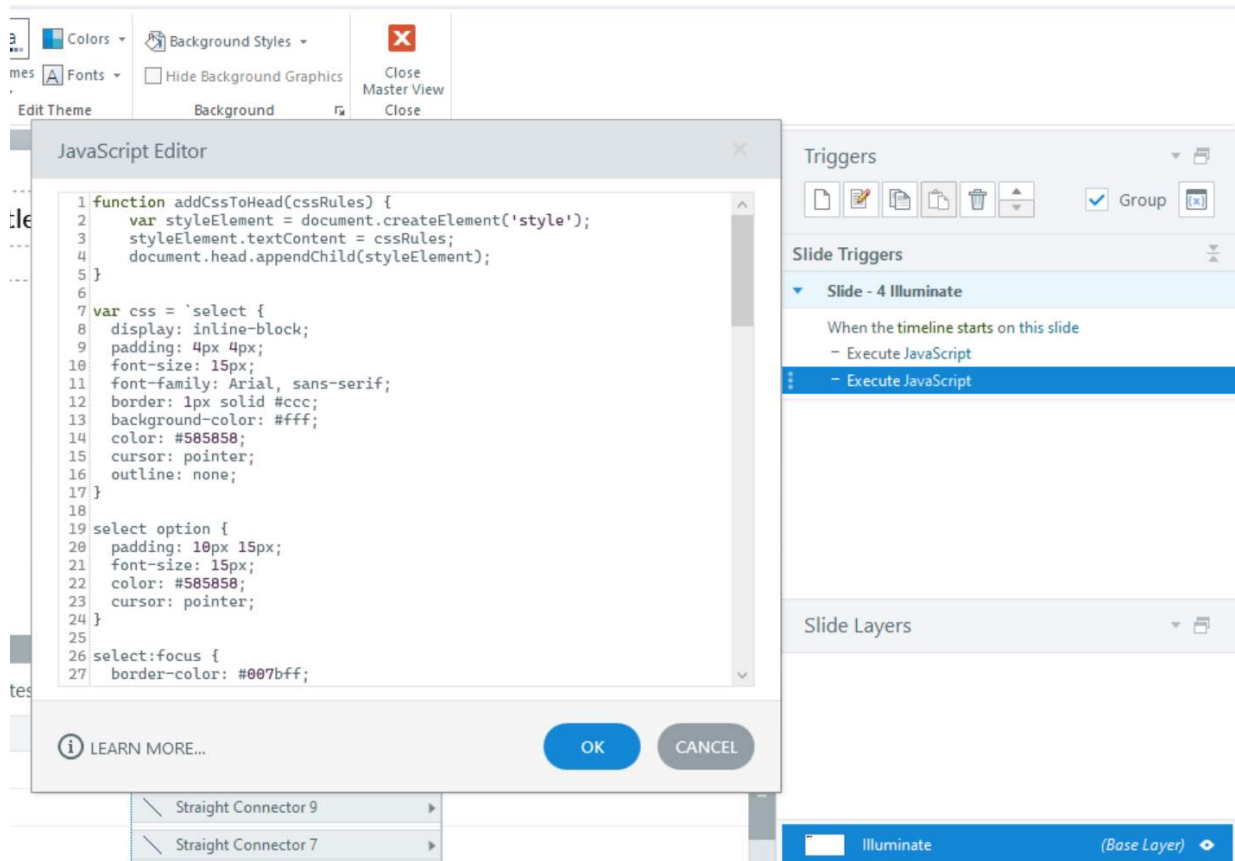


What if you could use provide your learners with the ability to change their font size, and preferred font family? Now you can!

The screenshot shows the 'Player Properties - Tutorials' interface. At the top, there is a 'PROPERTIES' tab and a navigation bar with icons for Features, Menu, Resources, Glossary, Player Style (Modern), Menus & Controls (On), Colors & Effects, Aa Text Labels, Other, and Current Player. Below this, the 'Player Tabs' section includes a 'Sidebar' with checkboxes for Menu, Glossary, and Resources, and a 'Topbar' with icons for document, edit, and trash. The 'Player Features' section has a checked 'Title' field with the text 'Pre-Authorization Request Tutorial-w-capti...', a 'Sidebar' dropdown set to 'On Left', and checkboxes for 'Sidebar starts collapsed', 'Logo', 'Cover Photo', and 'Fill player frame'. The 'Player Controls' section has checkboxes for 'Play / Pause', 'Volume', 'Captions', 'Playback speed', and 'Accessibility controls'. The 'Accessibility controls' checkbox is checked and highlighted with a purple box, with a tooltip 'Accessibility controls' appearing below it. To the right of the settings is a large grey rectangular area representing the player content.

First things first

Ensure that Accessibility Controls are turned ON in your Player. The JavaScript code modifies this panel, so it needs to be visible.



Add code to Master Slide

Execute the (below) JavaScript code on your MASTER SLIDE when the timeline starts on this slide.

```
function addCssToHead(cssRules) {
    var styleElement = document.createElement('style');
    styleElement.textContent = cssRules;
    document.head.appendChild(styleElement);
}

var css = `select {
    display: inline-block;
    padding: 4px 4px;
    font-size: 15px;
    font-family: Arial, sans-serif;
```



```
border: 1px solid #ccc;
background-color: #fff;
color: #585858;
cursor: pointer;
outline: none;
}

select option {
padding: 10px 15px;
font-size: 15px;
color: #585858;
cursor: pointer;
}

select:focus {
border-color: #007bff;
}

`;

addCssToHead(css);

var fontFamilyLabel = document.createElement("label");
fontFamilyLabel.textContent = "CC Font Family";
fontFamilyLabel.classList.add("switch-label");

var fontFamilySelect = document.createElement("select");
fontFamilySelect.id = "fontFamily";
var fontFamilies = [
    "Arial",
    "Helvetica",
    "Times New Roman",
    "Verdana",
    "Georgia",
    "Palatino",
    "Garamond",
    "Courier New",
    "Brush Script MT"
];
fontFamilies.forEach(function(fontFamily) {
    var option = document.createElement("option");
    option.value = fontFamily;
    option.textContent = fontFamily;
    fontFamilySelect.appendChild(option);
});

var fontSizeLabel = document.createElement("label");
```

```

fontSizeLabel.textContent = "CC Font Size";
fontSizeLabel.classList.add("switch-label");

var fontSizeSelect = document.createElement("select");
fontSizeSelect.id = "fontSize";
var fontSizes = [8, 10, 12, 14, 16, 18, 20, 24, 28, 32, 36, 40, 48, 56];
fontSizes.forEach(function(fontSize) {
    var option = document.createElement("option");
    option.value = fontSize;
    option.textContent = fontSize;
    fontSizeSelect.appendChild(option);
});
fontSizeSelect.value = "14";
var shortcutsSwitch = document.querySelector('[data-ref="shortcutsSwitch");

shortcutsSwitch.insertAdjacentHTML('afterend', '<br><br>');
shortcutsSwitch.insertAdjacentElement('afterend', fontFamilySelect);
shortcutsSwitch.insertAdjacentElement('afterend', fontFamilyLabel);
shortcutsSwitch.insertAdjacentHTML('afterend', '<br><br>');
shortcutsSwitch.insertAdjacentElement('afterend', fontSizeSelect);
shortcutsSwitch.insertAdjacentElement('afterend', fontSizeLabel);

fontFamilySelect.addEventListener("change", applyFont);
fontSizeSelect.addEventListener("change", applyFont);

function applyFont() {
var selectedFont = document.getElementById("fontFamily").value;
var selectedSize = document.getElementById("fontSize").value;

var styleElement = document.createElement("style");
styleElement.type = "text/css";

var cssRule = ".caption { font-family: " + selectedFont + " !important";

if (styleElement.styleSheet) {
    styleElement.styleSheet.cssText = cssRule;
} else {
    styleElement.appendChild(document.createTextNode(cssRule));
}

var existingStyleElement = document.getElementById("fontStyle");
if (existingStyleElement) {
    existingStyleElement.parentNode.removeChild(existingStyleElement);
}
}

```

```
styleElement.id = "fontStyle";  
  
document.head.appendChild(styleElement);  
}
```

Moving Captions



Jennifer Gupta

What if your learners could **MOVE** the captions where ever **THEY** wanted? Well... there's a JS for that too!

The screenshot shows a video player interface for a tutorial titled "Pre-Authorization Request Tutorial-w-captions". The video content features a man in a red shirt and khaki pants. There are several text boxes overlaid on the video: a pink "Instructions" header, an orange callout box with text, a grey box with detailed instructions, and a dark grey box with a tip. The video player includes a "Select Language" dropdown menu in the top right, a "Hide captions (Ctrl+Alt+C)" button in the bottom right, and standard accessibility icons (CC, a person icon, a gear, and a speech bubble icon) at the bottom right.

Pre-Authorization Request Tutorial-w-captions

Select Language

Instructions

The following hands-on tutorial will guide you through submitting a Pre-Authorization Request.

To use this tutorial, you can advance most of the steps by finishing the tasks, clicking on the correct places, or entering text. The Next and Back arrows are provided for navigation when necessary.

To begin, click the **Next Arrow**.

Advance most of the steps by finishing the tasks, clicking in the correct places,

Hide captions (Ctrl+Alt+C)

CC [person icon] [gear icon] [speech bubble icon]

Simply add this code to the Master Slide (Execute JavaScript when timeline begins on this slide).

```
const addCssToHead = (cssRules) => {  
  const styleElement = document.createElement('style');  
  styleElement.textContent = cssRules;  
  document.head.appendChild(styleElement);  
};  
  
const css = `  
.caption-container {  
  pointer-events: auto !important;  
}  
.caption{  
  cursor: grab;  
}  
`;  
addCssToHead(css);  
  
document.querySelectorAll('.caption-container').forEach(caption => {  
  let offsetX, offsetY, posX, posY;  
  
  const handleMouseMove = (e) => {  
    e.preventDefault();  
    let clientX, clientY;  
    if (e.clientX) {  
      clientX = e.clientX;  
      clientY = e.clientY;  
    } else if (e.touches && e.touches.length === 1) {  
      clientX = e.touches[0].clientX;  
      clientY = e.touches[0].clientY;  
    }  
    const dx = clientX - offsetX;  
    const dy = clientY - offsetY;  
    caption.style.transform = `translate(${posX + dx}px, ${posY + dy}px)`;  
  };  
  
  const handleMouseUp = () => {  
    document.removeEventListener('mousemove', handleMouseMove);  
    document.removeEventListener('mouseup', handleMouseUp);  
    caption.style.cursor = 'grab';  
  };  
});
```

```
};

caption.addEventListener('mousedown', (e) => {
  e.preventDefault();
  caption.style.cursor = 'grabbing';
  offsetX = e.clientX;
  offsetY = e.clientY;
  const rect = caption.getBoundingClientRect();
  posX = rect.left;
  posY = rect.top;
  document.addEventListener('mousemove', handleMouseMove);
  document.addEventListener('mouseup', handleMouseUp);
});

// Touch events for mobile devices
caption.addEventListener('touchstart', (e) => {
  e.preventDefault();
  caption.style.cursor = 'grabbing';
  offsetX = e.touches[0].clientX;
  offsetY = e.touches[0].clientY;
  const rect = caption.getBoundingClientRect();
  posX = rect.left;
  posY = rect.top;
  document.addEventListener('touchmove', handleMouseMove);
  document.addEventListener('touchend', handleMouseUp);
});
});
```

Transparent background on Captions



Jennifer Gupta

Ok, but what if you want to make the captions transparent (as in, no background color on them)?

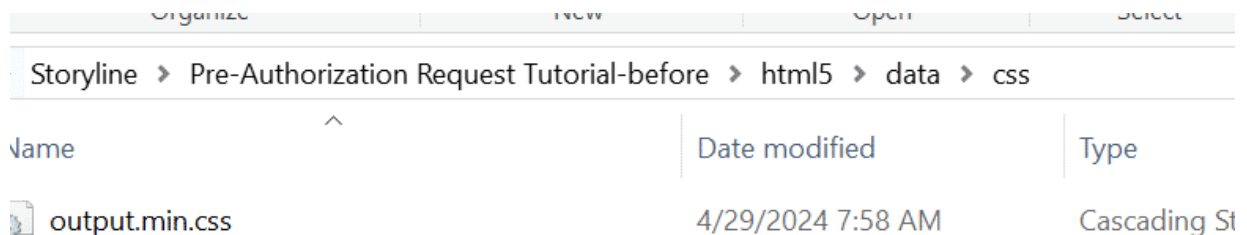
Well, then the JavaScript has to go into the output file, but it IS doable.

How do you do this? Publish your storyline file. Then, navigate to `html5/data/css/output.min.css`

Open this file with (Notepad) or another code editor.

Append the below code to the bottom of the page.

Save the file. (ReZip and use as normal)





```
.caption p {  
display: inline-block;  
pointer-events: none;  
padding: 10px;  
text-shadow: none;  
color: #ffffff;  
font-size: 18px;  
font-family: Play;  
background: 100%  
margin: 0 0 10px 0;  
padding: 10px;  
text-align: center;  
line-height: 1.5;  
}
```

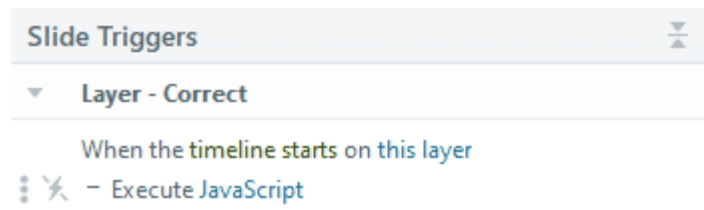
Add Confetti



Jennifer Gupta

Choose your style of confetti

First, select when you want confetti to appear. Is this on a page load, like when they reach the course exit/passed page? Or on a layer load, like when a correct layer loads, or on a button trigger? Wherever that is, you are going to need to create a trigger there, a trigger that executes JavaScript when that thing happens...



Make your trigger

This trigger executes the JavaScript when the timeline starts on the Correct layer.

Choose your style of confetti

Next, you need to choose the style of confetti you want. Visit Canvas Confetti's website and try them out!

Canvas Confetti

TAKE ME THERE!

Canvas Confetti

Basic Cannon

The default mode... just your regular basic average blast of confetti. But it's still a little cool, right?

Run ▶

```
1 confetti({
2   particleCount: 100,
3   spread: 70,
4   origin: { y: 0.6 }
5 });
```

Random Direction

Go crazy with some randomness. Shoot a random amount of confetti in random directions. (Go ahead... you know you want to click that button more than once.)

Run ▶

```
1 function randomInRange(min, max) {
2   return Math.random() * (max - min) + min;
3 }
4
5 confetti({
6   angle: randomInRange(55, 125),
7   spread: randomInRange(50, 70),
8   particleCount: randomInRange(50, 100),
9   origin: { y: 0.6 }
10 });
```

Realistic Look

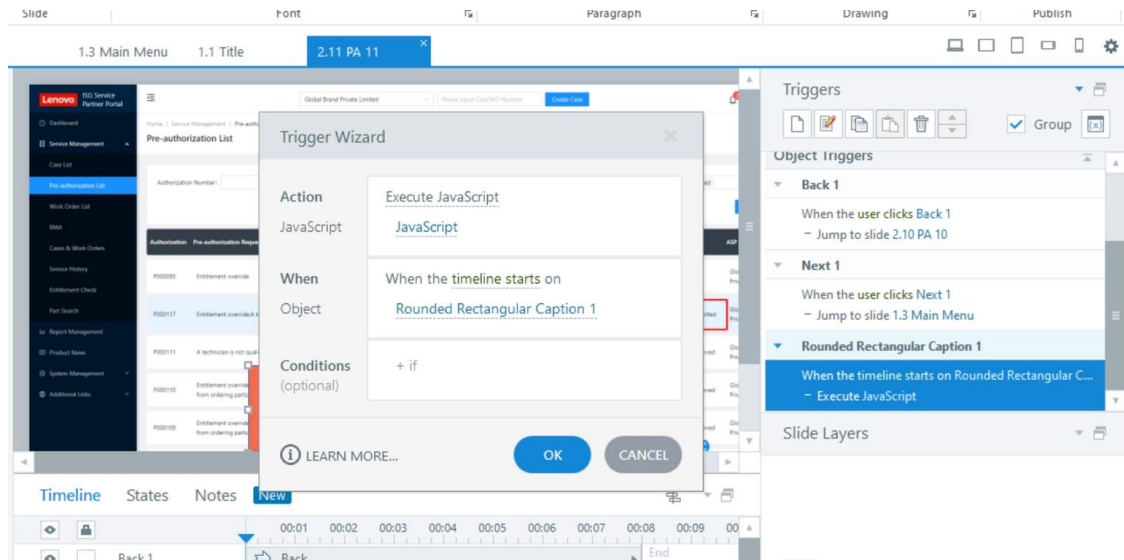
If you happened to get curious and changed the particle count to 400 or so, you

Scroll through the page. Click on any Run button to see the confetti in action.

Copy the code

Once you've selected the confetti you like, copy the code that sits below the Run button.



You'll add this into the JavaScript code section from above.



Now, copy in the javascript file into the correct directory

Publish your course with the JS trigger on it, then:

Download the js file (below) and add it into the /lib directory of your published file.

	confetti.browser.min.js 7.5 KB	
---	--	---

Html Editing

Go into your story.html file and add a line of code just before the </head> tag. You can open this file in NotePad or any code editor.



```
<script src="lib/confetti.browser.min.js" defer ></script>
```

Save the file

Save the file.

Test it to make sure it works, then zip and deploy!

Pre-Authorization Tutorial -Google-Translate-with-js.story (Preview)* - Articulate Storyline

FILE PREVIEW

Close Preview Close Select Replay Edit Slide Inspect Preview

Close Preview Close

Pre-Authorization Request Tutorial-js

NOTES LIGHT DARK

Lenovo ESG Service Partner Portal

Global Brand Private Limited | Please Input Case/MSI Number | [Link to Case](#)

Pre-authorization List

Authorization Number: Machine Identifier: ASP Name:

Search

Authorization	Pre-authorization Request	Submitted Date	Created By	Machine Identifier	Status	ASP Location	Action
100010	Entitlement exceeded technician is not qualified	2023-09-09 13:42:27	lgupta@csd@global.com	H4125005	Approved	Global Brand Private Limited	
100011	A technician is not qualified ASP Account is restricted to Labor Only	2023-07-13 22:21:52	shah23274@chicout.net	K239181	Approved	Global Brand Private Limited	
100012	Entitlement exceeded with ordering party				Approved	Global Brand Private Limited	
100013	Entitlement exceeded with ordering party				Approved	Global Brand Private Limited	

Congratulations! You have successfully created and submitted a Pre-Authorization Request!

Back Next

Slide 11 of 11 960 x 540 "Clean"

More ideas with JavaScript (Storyline)



Project 99

They are trying to add 99 ways to use JavaScript in Articulate products.

[GO TO PROJECT 99](#)

Create a typing Effect in Storyline

This one is really cool.. Use it for computer entry, phone simulations, etc!

[BRIDGEHILLS](#)

Remove x (to close) from Lightboxes

Ever wanted to remove the standard X to close in a Storyline Lightbox? Now you can! Don't forget to create triggers/buttons/etc of your own.

[BRIDGEHILLS](#)

Capture and Print Learner Responses

[BRIDGEHILLS](#)

Learning Dojo

Detect the Time

[YOUTUBE](#)

Export Data to Google Sheets

[LITTLE MAN PROJECT](#)

Target Specific objects in Storyline

[YOUTUBE](#)

Get Current Date

[VIMEO](#)

Motion examples

[YOUTUBE](#)

ELB Learning

JavaScript tricks in Storyline

[ELB LEARNING](#)

Location

Decent list of code snippets for use in Storyline!

[LEARNING DOJO JS CODE S...](#)

Lesson 11 of 14

GreenSock

JG Jennifer Gupta

GreenSock

Repository of JavaScript Animations

TAKE ME THERE

Learning Dojo

Enhance your courses with GreenSock

LEARNING DOJO

Our favorite Learning Dojo guy has created an amazing tool! Check out how to use it in the video below.

Custom Storyline Animations Made Easy

Learning Dojo



Watch on

Learning Dojo

Enhance your courses with GreenSock

[GSAP GENERATOR](#)

Ideas with JavaScript (Captive)



Create Notes & export

YOUTUBE

Manipulate the playbar

YOUTUBE

Location 3

This location can be a URL, another lesson, or an email address. You can type a description here.

GO TO LOCATION 3

FlexPerts



Flexperts

This newcomer to the field is making headway with creating some storyline generators that generate code for you to work with. They have started with a tilt animation, check it out!

Flexperts

Storyline Generators

[GO TO LOCATION 1](#)

Lesson 14 of 14

Learning Javascript



Want to learn it?

I'm going to start compiling some information on ways to learn JS so we can write our own!

CodeWithNatalia

Learn JS with Natalia - she's compiled all her links on one place

[LINKEDIN](#)