**DwBP - 8 - Implementing the Overlay Plugin Using Code**

The next use case we want to take care of is displaying an overlay that uses information from the media info object. We've worked with the media info object in an earlier video, and we're going to take that knowledge and apply it in a little different way.

When we use a plug-in in code like we are here, there's a few things we have to do. The first one, we check to see if there is a CSS file that's associated with the plug-in. The overlay plug-in there is and so we include that one in the head.

The next thing we want to make sure we have the JavaScript file. And that is in there. And then we write some code around using this specific plug-in. In this case, we're using the overlay plug-in. So again, we need to make sure we have the CSS, we have the JavaScript, and then somewhere in code we call the actual plug-in itself.

Let's look at this page what we have, make sure we're comfortable with the code. And we just talked about that. That's importing our CSS. Here again, we're using a style with the video-js class to size the player. Here is the video itself. Notice I've added an ID.

Here's the script associated with the player. Here is the overlay plug-in's JavaScript file. And now we have a little code to look at here. Lanes 33 and 34 are code we've looked at a few times now. This is just basically getting ready to interact with the player. We have the ready method there and we get a handle to our player with my player.

If you remember back to using the media info property, we had to wait for the load start event to be dispatched before we could get to the data. And so that's what we're doing here. We're waiting for the load start. So on line 36, you see we're waiting for the load start. And then we're going through and I'm assigning a variable here to this groups variable called myplayer.mediainfo.tags.

What that is, that's the object, in this it's an array, that contains the tags you have possibly entered for your video. So let's go look at that. I'm going to go to Studio. I'm going to go to media. Go in and click on the video. I'm going to scroll down here, and notice the tags I've entered are wildlife and tigers. And you can add to that in this interface, but right now the two I have are wildlife and tigers.

So basically, I want to use that information and it's stored in array, so I grab this array and I put it in groups. And then what I'm going to do is I'm going to use JavaScript's join method. And what that does is that simply takes an array and joins the array elements and separate them by a comma, but there's no comma after the last one. Of course, you could write a for loop to do the same thing, but I'm as lazy as all the other developers. If someone's already done the work for me, that's what I'm going to do.

Now here is where I actually call the overlay method. Well, how would I possibly know what to do? Well how you would know is you could go out, and I pointed out the documents on the plug-ins and you could find the one you want. So here's the overlay. And then you could look through, there will be an example here. And we can see there's an overlay up there. And in this case, when I pause it appears and disappears. Another overlay appears down here. So those are overlays. There will be an example.

And then it'll show you how to actually implement the overlay in Studio. And also, then it'll go through and show you how to implement the overlay using code. And so I'm going to go through and use some of the information that's in here, and then you see all the properties for the overlay. So that's how you'd know. You could go to the docs and get the appropriate doc for the overlay.

I'm going to define only a single overlay, and I want it to display some HTML here for me. So I simply put that in quotes. And you see I define a span and some text that says, for similar videos, see the groups. And then I put a BR in there to bring it to the next line. I do a strong. And here's where then I display that variable I created to hold the tags.

I told it I want it to start displaying on play. And then I want it to end after four seconds. So for start and end, you can either use events or you can use numbers, which has to do with where the current time is. And notice then the align is on the bottom left.

So this is an example where I'm using the code, and I'm using a code because I want to use some information that's available to me on the page. Let's go take a look at this working. Here I am at all the files.

I'm going to click Task 6 Solution. It's ready to go. And hopefully now when I click Play we'll see the overlay appear. And it does.

I'm not a designer so forgive me for the look. And there after four seconds it disappears.

Let's change that a little bit. Again, we have the code pen available. You can see there's the link. There's the video tag itself. And there's the scripts, one for the video and one for the overlay.

Here's the CSS I've added just to resize it. I don't care about that very much either. And then here's the JavaScript. So here, instead of "start on play, and end on--" let's change this to where we also end on an event.

So what I'm going to do is I'm going to say end on pause. I'll save that. And let's see what happens. I'm going to pull this up. I'll play this. There it is. There. Now, you can see it's both starting and stopping on an event. Let's change it once more.

Let's just say I want it in the bottom. And I'm going to start it at two seconds and have it end at eight seconds. There's two seconds. It's starting. I'll come down here so we can see, hopefully, 7, 8, and there it disappears.

Great. So now we've accomplished our two tasks we wanted to, we had use cases for. We wanted to play IMA3 ads, and we see that was all a Studio-based task and we simply used the UI to use that plug in. And in the next task, we wanted to use an overlay, the overlay plug in, that uses info from the mediainfo object. And we've done that also.

Now one last quick word. If I wanted to, I could encapsulate this logic of the plug in and build my own custom plug in and then apply it to other players. So we can encapsulate the logic we build in a plug in and almost make it just like a standard plug in that Brightcove supplies. Again, there are docs on how to do that.

Well, that concludes about plug ins. The next video, which is the last video of the series, we're going to use play list. We can define a play list and then the videos will play and the user can choose which video they want to be watching at a certain time from the playlist. Hopefully, we'll see in the next video. Thank you.