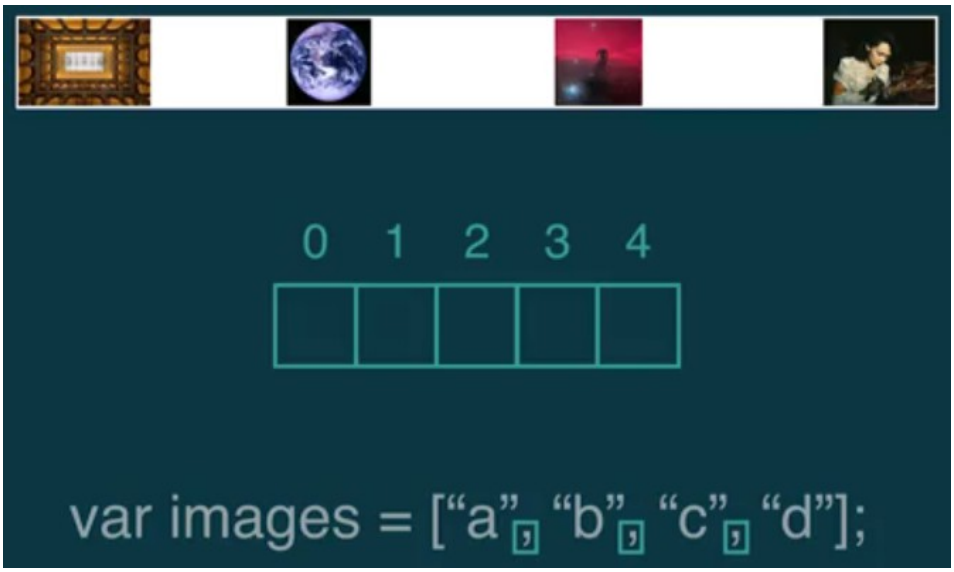
2.4.1.2: Storing objects in arrays and displaying them with a template

Now, you know the two most important ways of representing data in JavaScript.

- One is objects that we've looked at last week and
- now we've looked at also arrays which allows us to store large amounts of data by numbers.

And so now we need to look at how we take that data and put it into a template, the kind of templates we were working with last week. So let's very quickly recap. So this is what I showed you in the last video. It's an array, and the key thing about an array is it's a bunch of individual items with square brackets and separated by commas. That's a very simpler way. It's only got a few small text items in it.



Let's look at something a bit more complex, and in fact, something where an array contains, rather than individual variables, an array contains a bunch of objects, each one representing an image in that gallery. So here we have an example from the codes that you'll see.

```
var modal_template = Handlebars.compile(source);

    // define the data for the template
    // we define an objects which contains an
    // array of other objects. This array will be used
    // to create multiple images
    var data = {images:[
         {
                                                                    src:
"https://upload.wikimedia.org/wikipedia/commons/thumb/3/38/Shopping_Center_Magna
_Plaza_Amsterdam_2014.jpg/600px-Shopping_Center_Magna_Plaza_Amsterdam_2014.jpg",
           title: "Shopping Center Magna Plaza Amsterdam 2014",
           author: "Tuxyso",
           },
         {
                                                                    src:
"https://upload.wikimedia.org/wikipedia/commons/thumb/9/97/The_Earth_seen_from_A
pollo_17.jpg/600px-The_Earth_seen_from_Apollo_17.jpg",
           title:"The Earth seen from Apollo 17",
           author:"Ed g2s"
           },
         {
```

You want to understand, right, I've got a variable dana (**var** data), which is all my data. And you'll see it's actually an object and it contains a single member variable called images ({images:[])

- The reason it's an object is so that we can pass it into the template. But what I want you to look at is that this images member variable is an array {images:[

You can tell that because it's got square brackets here and inside an array is a bunch of objects. So each of these objects looks very much like the ones we were working with last week. It's got a source, a title, and an author and they've all got the same structure, same member variables. But they have different values for the author, that each one has a different image file, a different title and different author, exactly what we want. You should also notice that they're all separated by commas, and that's just how you create an array. So this is the kind of complex array we'll actually be working with. It has a fairly large number of objects, each which contains a bunch of complex data. So, that's a powerful technique. We've got quite a rich data structure now. We've got lots of data that we can include in our image gallery. But how do we actually get it onto the web page? Well, exactly the same technique we used last week, which is templates. But I need to show you something slightly different to see how they work with arrays.

This is a template that works with our array. It looks pretty similar to what we've seen before. It's a script with tight handlebars. I've included the dots there instead of the full title to fit it on the slide. I'll do that once in a while. And we've got template expressions, we've got the title and author, very much like we had the last time, but we've got a few new things.

- And the main thing that we're doing here is that this template is not designed just to display a single item, it's designed to display a lot of images.

So it's actually divided into two parts, so this alta div is designed to be done once. So this, you can think about all the stuff that is around all of the image gallery so you can put a title to your image gallery, maybe, up there, and you only want it to happen once. But most of the templates is in here which is, it happens for each data item. So we've now got a special template expression which has got this hash h or pound h, depending on how you'd like to pronounce that symbol. What that means is that it's its own mini-template inside the other templates and it happens for each image in the array, so with each item in the array called images. So every time it finds an item in the record images, it repeats this template.

And so all this code in here, this H1 with the title in H3, with the author is repeated for every single image. And the title and author we're looking at here, the title and author of the particular image we're looking at now. So, each time this mini bit of template is repeated, it'll have a new title, a new author. I have left out the source, but it'll also have a new source image.

Here's the full template code. You'll see it's pretty much the same as what I showed you. It's got this each images in here. The stuff that's common, that's done once, is pretty simple. It's just a bootstrap row is created for them and then for each image we've got some bootstrap formatting stuff. And basically, each one has its one image tag with a source, and a title, and an author. And as I said, these individual ones were repeated many, many times for each image that's coming out of the array called images. And so you will get an entire photo album with lots and lots of images, each with its unique image source, title, and author. And this is what it looks like. So instead of having just one individual image, we've got a whole sequence of images, each of whom has the same basic template structure, but each of which has its different data coming in from different items in that array. So we've now got a very powerful technique which will allow us to build websites with lots of bits of content, lots of individual parts. And this is something we're going to develop for the rest of this week, but also in the whole rest of the MOOC, as we move on to server side programming.

But, for today, I just wanna carry on in the next few videos with some detailed techniques we can use with this kind of array-based gallery.