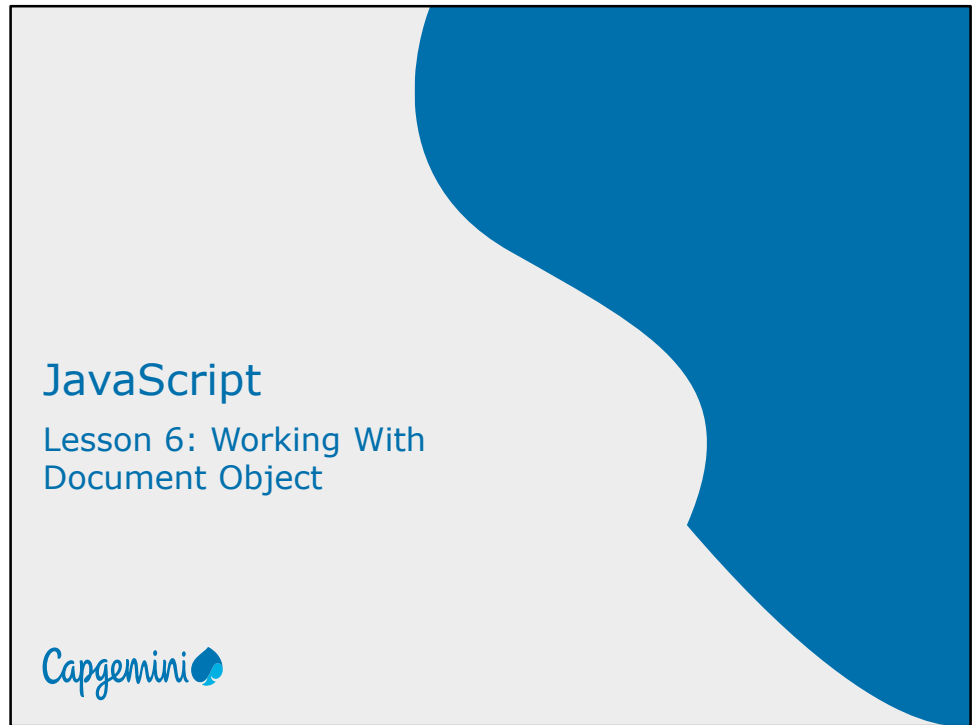


**Instructor Notes:**



## Instructor Notes:

### Lesson Objectives

- To understand the following topics:
  - Document Object and its properties and methods
  - Cookies object



## Instructor Notes:

### 6.1: Document Object

## Working With Document Object



- Container for all HTML HEAD and BODY objects associated within tags
- Provides access to page elements from your script
  - This includes form, link, anchor, as well as global Document properties such as background and foreground colors

*Document object* is part of the Window object. It is used to access all elements in a page. It provides access to the elements in an HTML page from within the script.

This includes the properties of every form, link and anchor (and, where applicable, any sub-elements), as well as global document properties such as background and foreground colors.

## Instructor Notes:

### 6.1: Document Object



## Document Object Properties

- `alinkColor`, `vlinkColor`, `bgColor`, `fgColor`, `linkColor`
- `anchors[]`
- `applets[]`
- `forms[]`
- `links[]`
- `title`

Property	Description
<code>alinkColor</code> <code>vlinkColor</code> <code>bgColor</code> <code>fgColor</code> <code>linkColor</code>	Get and set the properties of document – activated link, visited link, background color, foreground color (text) and hyperlink color.
<code>anchors[]</code> , <code>forms[]</code> , <code>links[]</code>	These properties retrieve array of values respectively as present in the document object
<code>title</code>	Gets the title of the document which occurs between the TITLE tags.

## Instructor Notes:

## 6.1: Document Object



## Document Object Methods

- write(), writeln()
- getElementsByTagName()
- getElementById()
- getElementsByName()
- getElementsByClassName()

Property	Description
write("string1", ...) writeln("string1", ..)	Both of these methods send text to a document for display in its window. The only difference between the two methods is that <i>document.writeln()</i> appends a carriage return to the end of the string it sends to the document (you must still write a   to insert a line break).
getElementById("#para1")	This method locates the element whose id has been passed. The text within this element can then be accessed using properties innerHTML or innerText
getElementsByTagName("p")	This method locates all the elements which match the tagname passed. Each element of this type of tag can then be accessed in an array like manner
getElementsByName()	This method locates all the elements which match the name passed. Same name to many elements is usually given for radio buttons.
getElementsByClass()	This method locates all the elements which match the class name passed.

## Instructor Notes:

### Demo

- `Link_Anchor_object.html`
- `Meta_information.html`
- `locate_element_by_id.html`
- `locate_elements_by_tagname.html`
- `locate_elements_by_name.html`
- `locate_element_by_class_name.html`



## Instructor Notes:

6.2: Working with Cookies



## Working with Cookies

- Text files that Web sites place in your computer to help your browsers remember specific information
- Used to store user preferences for content or personalized pages
- Following function sets cookie values (expiration date is optional):

```
function setCookie(name, value, expire)
{
    document.cookie = name + "=" + escape(value)
    + ((expire == null) ? "" : ("; expires=" + expire.toGMTString()))
}
```

### Using Cookies

*Cookies* are a mechanism for storing persistent data on the client in a file called *cookies.txt*. Because HyperText Transport Protocol (HTTP) is a stateless protocol, cookies provide a way to maintain information between client requests. This section discusses basic uses of cookies and illustrates with a simple example.

Each cookie is a small item of information with an optional expiration date and is added to the cookie file in the following format:

`name=value;expires=expDate;`

*Name* is the name of the datum being stored, and *value* is its value. If *name* and *value* contain any semicolon, comma, or blank (space) characters, you must use the *escape* and *unescape* functions to encode and decode them respectively.

*expDate* is the expiration date, in GMT date format:

`Wdy, DD-Mon-YY HH:MM:SS GMT`

Although it is slightly different from this format, the date string returned by the *Date* method *toGMTString* can be used to set cookie expiration dates.

The expiration date is an optional parameter indicating how long to maintain the cookie. If *expDate* is not specified, the cookie expires when the user exits the current browser session. Browser maintains and retrieves a cookie only if its expiration date has not yet passed.

## Instructor Notes:

### Limitations

Cookies have these limitations:

300 total cookies in the cookie file.

4 Kbytes per cookie, for the sum of both the cookie's name and value.

20 cookies per server or domain (completely specified hosts and domains are treated as separate entities and have a 20-cookie limitation for each, not combined).

Cookies can be associated with one or more directories. If your files are all in one directory, then you need not worry about this. If your files are in multiple directories, you may need to use an additional path parameter for each cookie.

### Using Cookies with JavaScript

The *document.cookie* property is a string that contains all names and values of Navigator cookies. Use this property to work with cookies in JavaScript.



## Instructor Notes:

### Demo

- Ch9\_ex1.cookie.html
- Demo\_Cookies.html



## Instructor Notes:

### Lab

- Lab 7 :
- Working with Document object.



## Instructor Notes:

### Summary



- JavaScript Document Object contains HTML elements contained in the <head> and <body> sections of a web page
- All the anchors are contained in anchor array.
- All the links are contained in link array
- Cookies are small text files stored on the site visitor's computer by their browser



#### Summary

In this chapter, you understood:

- DOM structure
- How to work with Document Object
- How to work with cookies

## Instructor Notes:

### Review Questions

➤ Question 1: The \_\_\_\_\_ is the container for all HTML HEAD and BODY objects.

- Option 1: Document
- Option 2: Object
- Option 3: Container

➤ Question 2: \_\_\_\_\_ property in document object retrieves an indexed array of anchors in a document.

