

## **UNIT 4 : JAVASCRIPT OBJECTS**

---

### 4.1 Creating object :

(By object literal, By creating instance of Object,  
By using an object constructor)

### 4.2 Date object :

4.2.1 Date constructor: Date(), Date(milliseconds),  
Date(dateString), Date(year, month, day, hours,  
minutes, seconds, milliseconds)

4.2.2 Date Methods: getDate(),  
getDay(),getMonth(), getHours(),  
setDate, setMonth(),setDay(), toString()

---

## 4.3 Document Object Model (DOM):

4.3.1 DOM concepts

4.3.2 DOM properties

4.3.3 DOM methods :

write(),

writeln(),

getElementById(),

getElementsByName()

---

## 4.1 Creating object

---

- ❑ By object literal
  - ❑ By creating instance of Object
  - ❑ By using an object constructor
-

# By object literal

---

- ❑ The object literal is an easy way of creating an object.

- ❑ It is creating {...} brackets.

- ❑ Syntax:

```
var object_nam={  
    property1:value1,  
    property2:value2,....  
    propertyn:valuen  
};
```

---

# Example

---

```
<html>
  <head>
    <script>
      var student={name:"ram",
                    location:"Ayodhya",
                    Age:10000};
      document.write(student.name);
    </script>
  </head>
</html>
```

---

# By creating instance of Object

---

- ❑ The second way to create an object is using new keyword with object().
  - ❑ The new operator is used to create an instance of an object.
  - ❑ Dot . and [] can be used to specify properties and methods.
-

---

## □ Syntax:

- `var object_name= new Object();`
  - `Object_name.`  
`property_name=property_value; // specify`  
`property using dot`
  - `Object_name["property_name"]=property_value;`  
`// specify property using []`
-

# Example

---

```
<html>
  <head>
    <script>
      var student = new Object();
      student.name="Ram";
      student["location"]="Ayodhya";
      document.write(student.name+"<br>");
      document.write(student["location"]);
    </script>
  </head>
</html>
```

---



# By using an object constructor

---

- ❑ First create a function with parameters then assign value to each parameters using this keyword.
  - ❑ This keyword refers to the current object.
-

---

## □ Syntax:

```
function function_name(parameter1,  
parameter2,...., parametern)  
{  
    this.parameter1=value1;  
    this.parameter2=value2;  
    ...  
    this.parametern=valuen;  
}  
object_name= new function_name(value1,  
value2,....valuen);
```

---

# Example:

---

```
<html>
  <head>
    <script>
function stud(name,location,age)
{
    this.name=name;
    this.location=location;
    this.age=age;
}
student = new stud("Ram","Ayodhya",1000);
document.write(student.name+"<br>");
document.writeln(student.location+"<br>");
document.writeln(student.age);
    </script>
  </head>
</html>
```

## 4.2 Date object :

---

### 4.2.1 Date constructor:

`Date()`,

`Date(milliseconds)`,

`Date(dateString)`,

`Date(year, month, day, hours,  
minutes, seconds, milliseconds)`

---

---

## 4.2.2 Date Methods:

getDate(),  
getDay(),  
getMonth(),  
getHours(),  
setDate,  
setMonth(),  
setDay(),  
toString()

---

# Date constructor:

---

- ❑ The purpose of date object is to work with date(days, months, years) and time (milliseconds, seconds, minutes, hours)
  - ❑ Date object is created using **new Date()**
  - ❑ Syntax :
  - ❑ `var dateObj=new Date();`
-

# Date()

---

- ❑ This is the default constructor with no argument.
  - ❑ It initializes the Date object with the current date and time.
-

# Example:

---

```
<html>
  <head>
    <script>
      var dateObj= new Date();
      document.write(dateObj);
    </script>
  </head>
</html>
```

---



# Date(milliseconds)

---

- ❑ This method is with integer parameter.
  - ❑ It represents the date in milliseconds since January 1, 1970
  - ❑ Example:
    - `dateObj = new Date(999999);`
-

# Example:

---

```
<html>
  <head>
    <script>
      dateObj= new Date(99999);
      document.write(dateObj);
    </script>
  </head>
</html>
```

---

## Date(dateString)

---

- ❑ A dateString parameter is string value representing a date, in the format accepted by the Date.parse() method.
-

# Example:

---

```
<html>
  <head>
    <script>
      var str="8/25/2021";
      dateObj= new Date(str);
      document.write(dateObj);
    </script>
  </head>
</html>
```

---

# Date(year, month, day [, hours, minutes, seconds, milliseconds])

---

Parameter	Description
Year	Specify the year
Month	Specify the month, range from 0 to 11 (January to December)
Day	Specify the day of the month. Its default value is 1
Hours	Specify the hours of the day in 24 hour format. Default value is 0
Minutes	Specify minutes of the time
Seconds	Specify second of the time
Milliseconds	Specify the millisecond of the time

---

# Example:

---

```
<html>
  <head>
    <script>
      bdate= new Date(1988,8,25,5,5,5,0);
      document.write(bdate);
    </script>
  </head>
</html>
```

---

## 4.2.2 Date Methods:

---

- ☐ getDate()
  - ☐ getDay()
  - ☐ getMonth()
  - ☐ getHours()
  - ☐ setDate
  - ☐ setMonth()
  - ☐ setDay()
  - ☐ toString()
-

# getDate()

---

- ❑ It returns the day (from 1 to 31) of the month.
-



## getDay()

---

- ❑ It returns the day of the week
  - ❑ Value form 0 to 6 (Sunday to Saturday)
-

# getMonth()

---

- ❑ It returns the month
  - ❑ The value is from 0 to 11 (January to December)
-

## getHours()

---

- ❑ It returns the hours
  - ❑ Value is from 0 to 23
-

# Example

---

```
<html>
  <head>
    <script>
      var bdate= new Date();
      document.write("Day of Month :"+bdate.getDate()+"<br>");
      document.write("Day :"+bdate.getDay()+"<br>");
      document.write("Month :"+bdate.getMonth()+"<br>");
      document.write("Hour :"+bdate.getHours()+"<br>");
    </script>
  </head>
</html>
```

---

# setDate()

---

- ❑ It sets the day of given date object.
-

## setMonth()

---

- ❑ It sets the day of the month of given date object.
-

# setDay()

---

- ❑ It sets the day of the month of given date object.
-

## toString()

---

- ❑ It converts date object to string value
-



# Example

---

```
<html>
  <head>
    <script>
      var bdate= new Date();
      bdate.setDate(10);
      bdate.setMonth(10);
      bdate.toString();
      document.write("Day of Month :"+bdate.getDate()+"<br>");
      document.write("Day :"+bdate.getDay()+"<br>");
      document.write("Month :"+bdate.getMonth()+"<br>");
      document.write("To String :"+bdate.toString()+"<br>");
    </script>
  </head>
</html>
```

---

## 4.3 Document Object Model (DOM):

---

4.3.1 DOM concepts

4.3.2 DOM properties

4.3.3 DOM methods :

write(),

writeln(),

getElementById(),

getElementsByName()

---

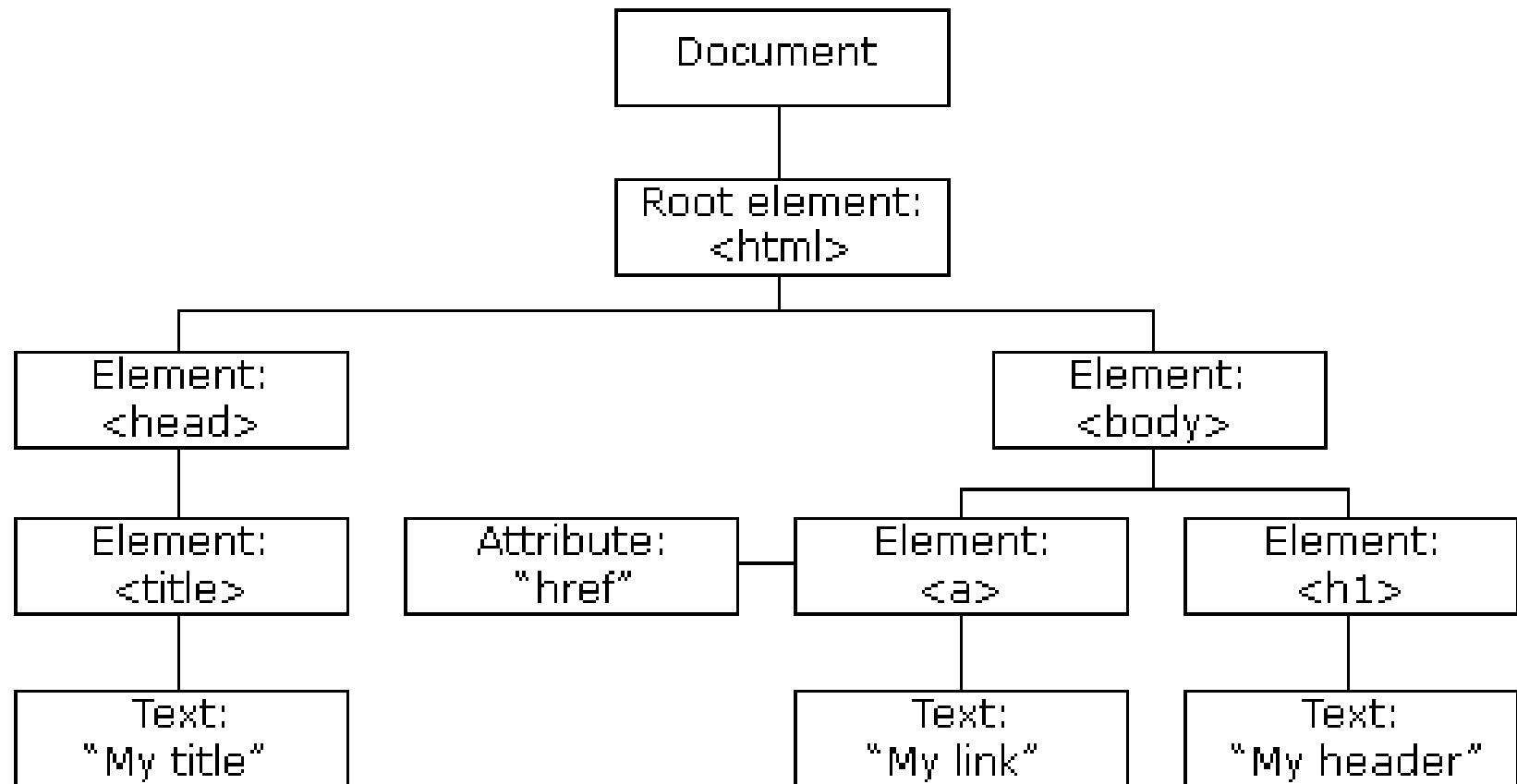
## 4.3.1 DOM concepts

---

- ❑ The HTML DOM is a standard **object** model and **programming interface** for HTML.
  - ❑ It defines:
    - The HTML elements as **objects**
    - The **properties** of all HTML elements
    - The **methods** to access all HTML elements
    - The **events** for all HTML elements
-

# The HTML DOM Tree of Objects

---



## 4.3.2 DOM properties

---

- ❑ DOM properties are accessed using the “document” object.
  - ❑ Syntax:
    - `document.property_name`
-

Properties	Description
links	refers to all <a> and <area> elements that have a “href” attributes
anchors	refers to all <a> elements that have “name” attributes
images	refers all <img> elements
doctype	Specify the document’s doctype
scripts	refers to all <script> elements
head	refers to all <head> elements
forms	refers to all <form> elements
cookie	returns all name/value pairs of cookies in the document
domain	refers domain name of the server
lastModified	get the date and time of the last updated document
readyState	get the status of the document
title	refers to <title> of the document
URL	specify the URL of the Document

### 4.3.3 DOM methods :

---

- ☐ write()
  - ☐ writeln()
  - ☐ getElementById()
  - ☐ getElementsByTagName()
-

# write()

---

- ❑ Write some text directly to the HTML document.
  - ❑ Syntax:
    - `document.write(exp1, exp2, exp3, ...)`
-



# Example:

---

```
<script>  
document.write("VNSGU");  
</script>
```

---

# writeln()

---

- ❑ The `writeln()` method is identical to the [`document.write\(\)`](#) method, with the addition of writing a newline character after each statement.
-

## Example:

---

```
<script>  
document.writeln("VNSGU");  
document.writeln("SURAT");  
</script>
```

---

# getElementById()

---

- ❑ The `getElementById()` method returns the element that has the ID attribute with the specified value.
  - ❑ Syntax:
    - `document.getElementById(elementID)`
-

# Example:

---

```
<html>
<head>
<script>
function test()
{
    var s=document.getElementById("txt").value;
    document.write(s);
}
</script>
</head>
<body>
Enter Your Name :<input type="text" id="txt"> <br>
<input type="button" value="Click Here" onclick="test();">
</body>
</html>
```

---

# getElementsByName()

---

- ❑ The `getElementsByName()` method returns a collection of all elements in the document with the specified name.
  - ❑ The [HTMLCollection](#) object represents a collection of nodes. The nodes can be accessed by index numbers. The index starts at 0.
  - ❑ Syntax:
    - `document.getElementsByName(name)`
-

# Example:

---

```
<html>
<head>
<script>
function test()
{
    var s1=document.getElementsByName("txt")[0].value;
    var s2=document.getElementsByName("txt")[1].value;
    //document.write(s1+" "+s2);
    alert(s1+" "+s2);
}
</script>
</head>
<body>
Enter Your First Name :<input type="text" name="txt"> <br>
Enter Your Last Name :<input type="text" name="txt"> <br>
<input type="button" value="Click Here" onclick="test();">
</body>
</html>
```

---