Lecture 10: JavaScript Events

ICT1153

Buddhika Gayashani Jayaneththi
Department of ICT
Faculty of Technology
University of Ruhuna

Events

- HTML events are "**things**" that happen to HTML elements.
- When JavaScript is used in HTML pages, JavaScript can "**react**" on these events.
- Examples of events:
  - A mouse click
  - A web page or image loading
  - HTML input field change
  - Key press
  - Selecting an input box in an HTML form
  - Submitting a HTML form

Event Handler

- Event handler is something that you write into a web page which responds whenever a specific type of Event happens.
- HTML allows **event handler attributes, with JavaScript code**, to be added to HTML elements.

Example :

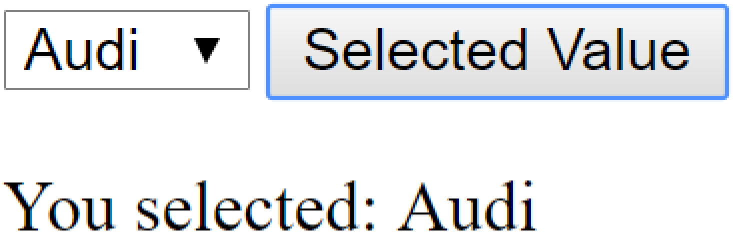
```
<input type="button"  name="click1"  value="Click me!”
onclick= “event_handler_code”>
```

Common HTML Events

Event	Description
onchange	An HTML element has been changed
onclick	The user clicks an HTML element
onmouseover	The user moves the mouse over an HTML element
onmousemove	The mouse pointer is moving while it is over an element
onmouseout	The user moves the mouse away from an HTML element
onkeydown	The user pushes a keyboard key
onload	The browser has finished loading the page

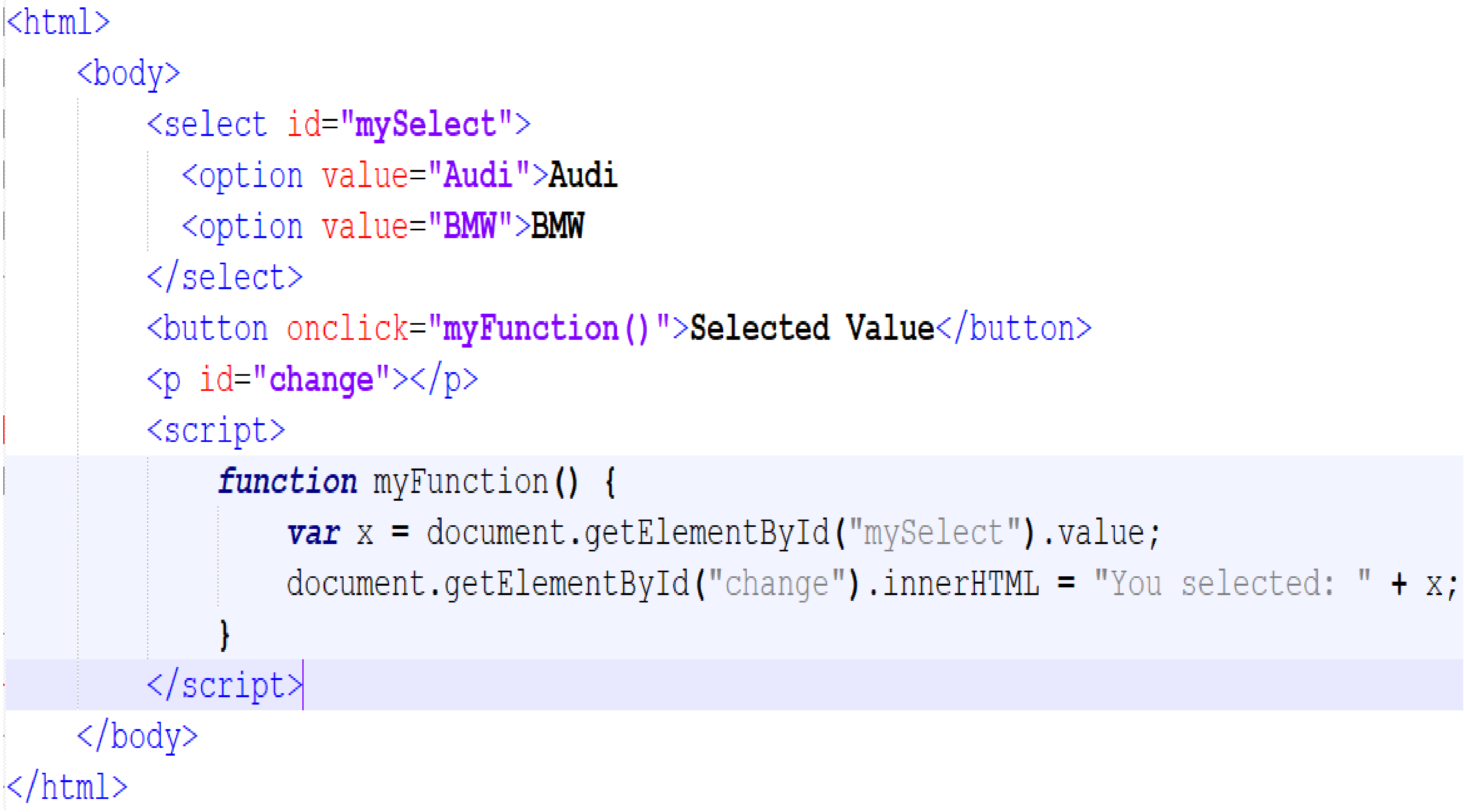
Exercise

- Create a web page consists with a select box (options: Audi, BMW)
- When the user clicks on the button after selecting an option, the page should display "You selected: ***option***" as a paragraph text as given below
- Use javascript to perform the function (you may use id values for elements where appropriate)



You selected: Audi

Answer



```
<html>
    <body>
        <select id="mySelect">
          <option value="Audi">Audi
          <option value="BMW">BMW
        </select>
        <button onclick="myFunction()">Selected Value</button>
        <p id="change"></p>
        <script>
            function myFunction() {
                var x = document.getElementById("mySelect").value;
                document.getElementById("change").innerHTML = "You selected: " + x;
            }
        </script>
    </body>
</html>
```

Object Based Programming

Object Based Programing

- JavaScript is an OOP language.
- It allows you to define your own objects and make your own variable types.
- Also there are built-in objects in JavaScript.
- **Object**: A real world entity which has *state* and *behavior*
  - Ex:

Object : Bicycle
States: price, brand, current speed
Behavior: changing gear, applying brakes

Object : Dog
States: name, color, breed, hungry
Behavior: barking, fetching, wagging tail

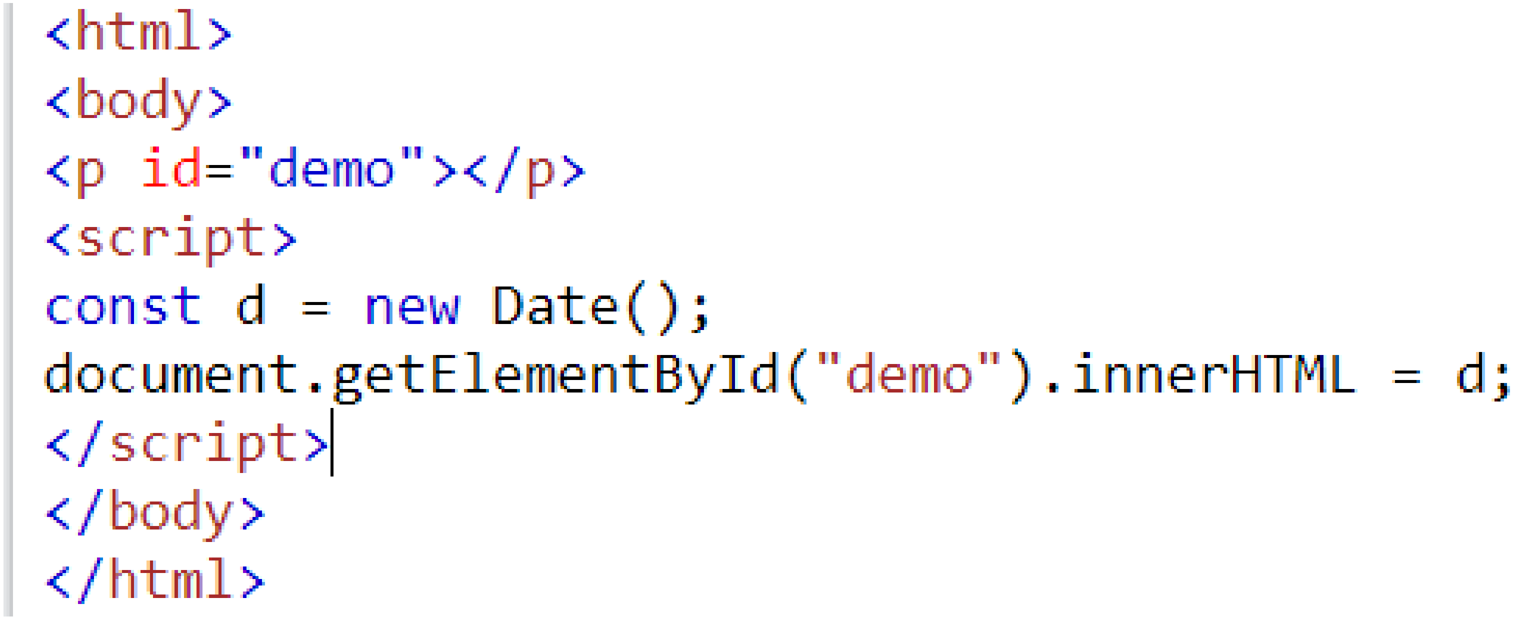
Date Object

- Date object is useful when working with dates.
- A date consists of a year, a month, a day, an hour, a minute, a second, and milliseconds.
- Date objects are created with the **new Date()** constructor.
- There are 4 ways of initiating a date:
  - new Date()
  - new Date(*milliseconds*)
  - new Date(*date string*)
  - new Date(*year, month, day, hours, minutes, seconds, milliseconds*)

4 ways to create a Date Object

new Date()

- Creates a new date object with the **current date and time.**

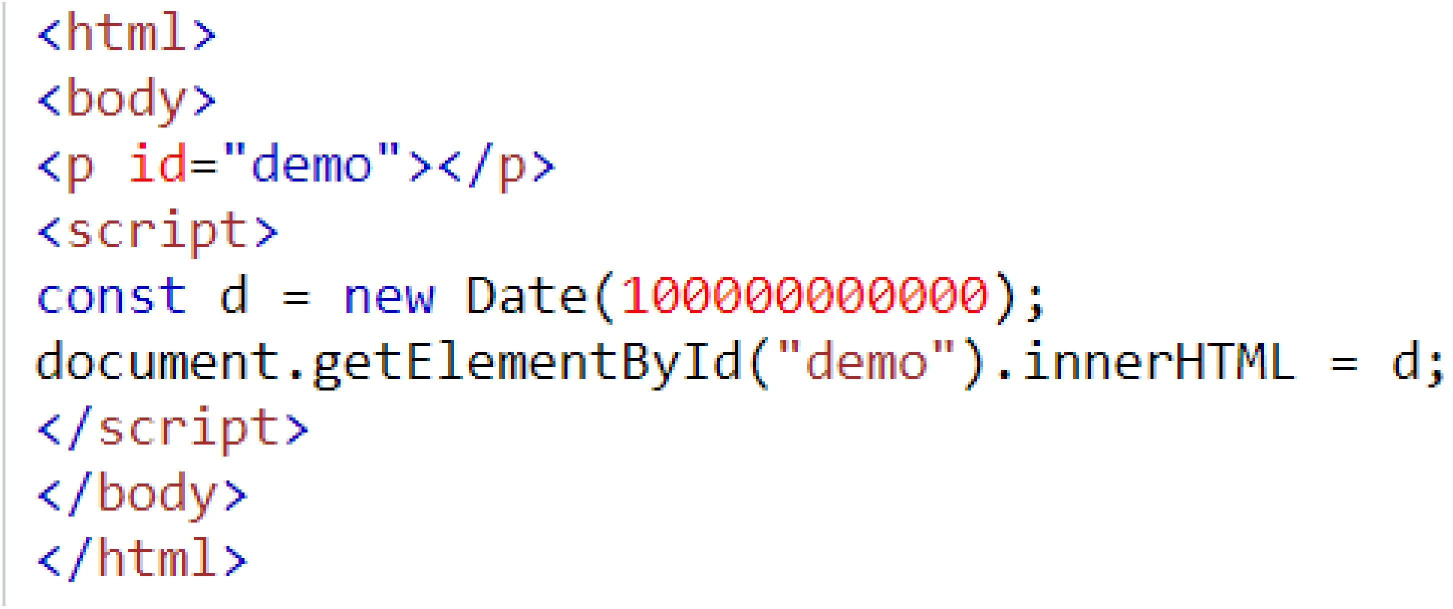


```
<html>
<body>
<p id="demo"></p>
<script>
const d = new Date();
document.getElementById("demo").innerHTML = d;
</script>
</body>
</html>
```

4 ways to create a Date Object

new Date(*milliseconds*)

- Creates a new date object as **zero time plus milliseconds.**
- 01 January 1970 **plus** 100 000 000 000 milliseconds is approximately 03 March 1973.

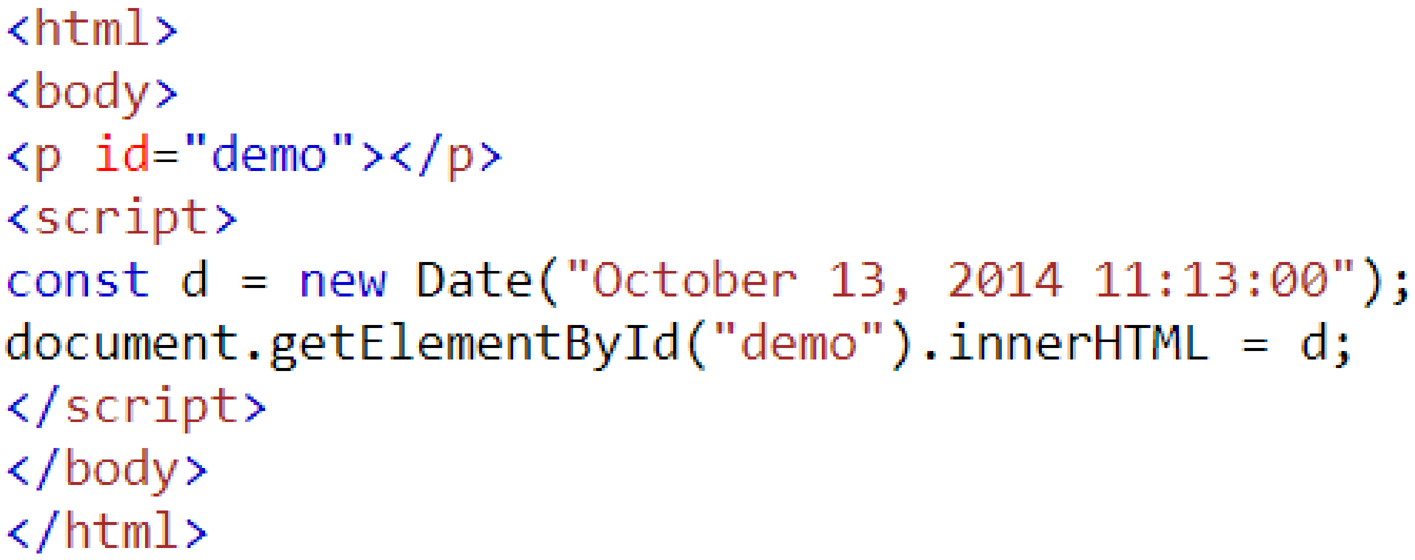


```
<html>
<body>
<p id="demo"></p>
<script>
const d = new Date(100000000000);
document.getElementById("demo").innerHTML = d;
</script>
</body>
</html>
```

4 ways to create a Date Object

new Date(*date string*)

- Creates a new date object from a **date string.**



```
<html>
<body>
<p id="demo"></p>
<script>
const d = new Date("October 13, 2014 11:13:00");
document.getElementById("demo").innerHTML = d;
</script>
</body>
</html>
```

4 ways to create a Date Object

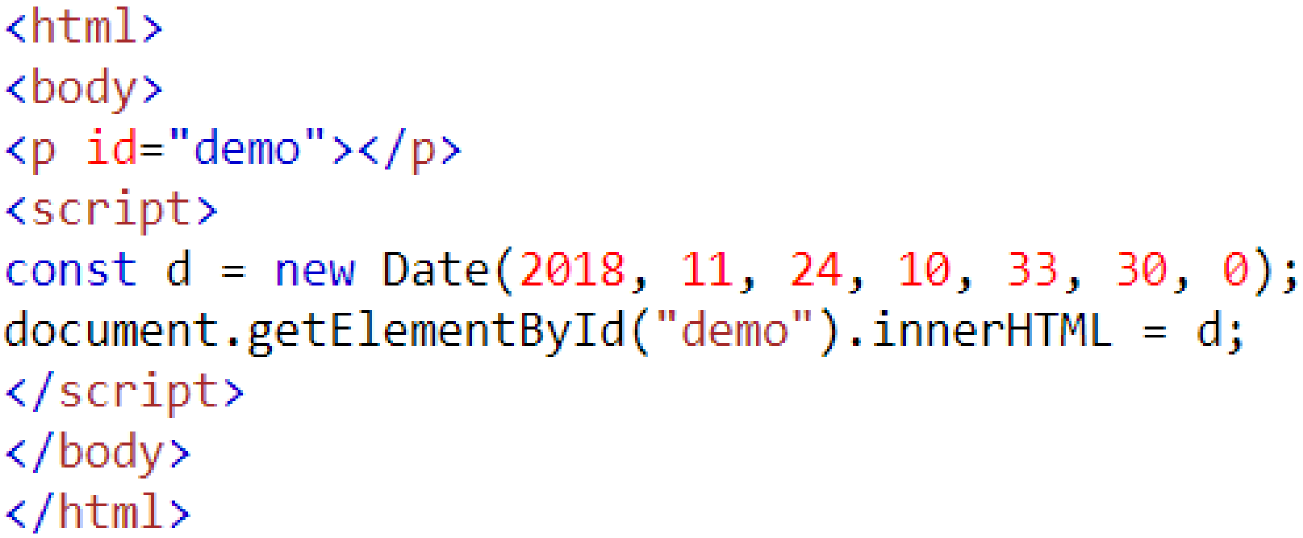
new Date(*year, month, day, hours, minutes, seconds, milliseconds*)

- Creates a new date object with a **specified date and time.**
- 7 numbers specify year, month, day, hour, minute, second, and millisecond (in that order)**.**

- **Note:** JavaScript counts months from **0** to **11**:

January = 0.

December = 11.



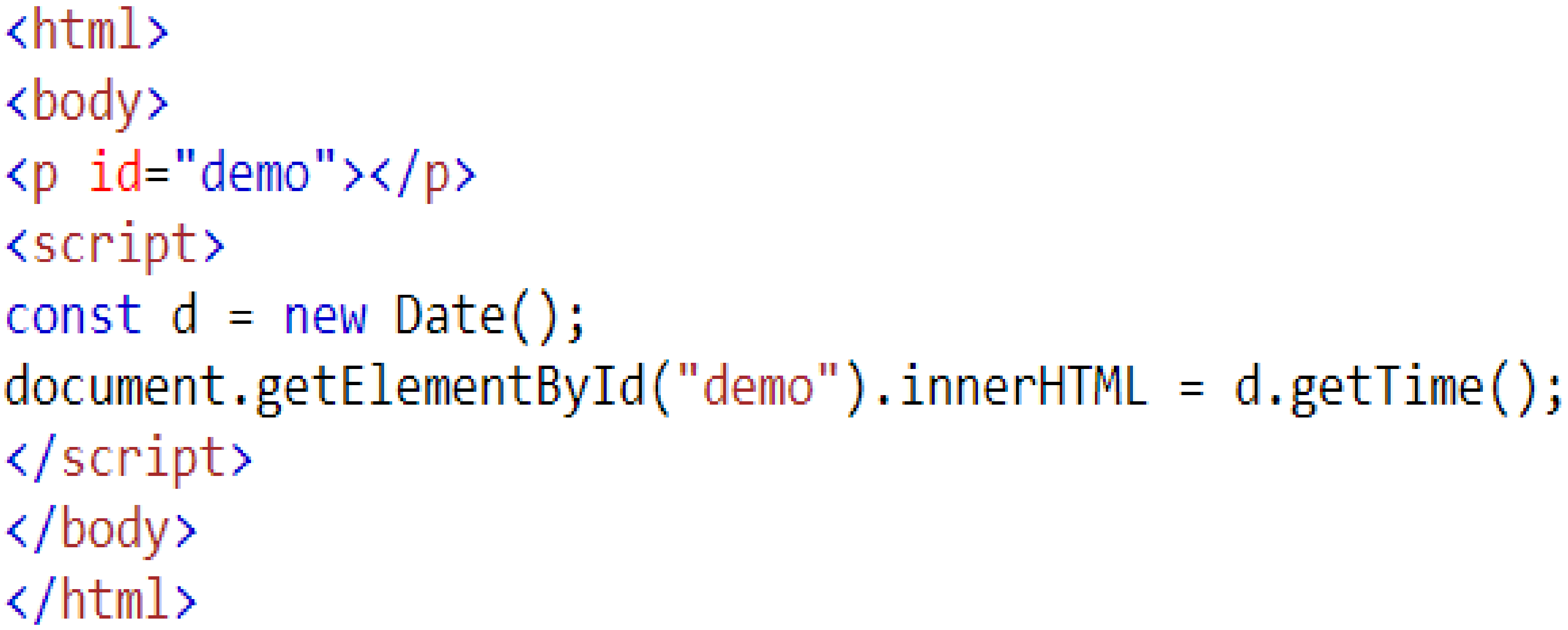
```
<html>
<body>
<p id="demo"></p>
<script>
const d = new Date(2018, 11, 24, 10, 33, 30, 0);
document.getElementById("demo").innerHTML = d;
</script>
</body>
</html>
```

Date Methods

- Date methods are to get and set date values.
- Date get methods are:

Method	Description
getDate()	Get the day as a number (1-31)
getDay()	Get the weekday as a number (0-6)
getFullYear()	Get the four digit year (yyyy)
getHours()	Get the hour (0-23)
getMinutes()	Get the minutes (0-59)
getMonth()	Get the month (0-11)
getSeconds()	Get the seconds (0-59)
getTime()	Get the time (milliseconds since January 1, 1970)

Date Get Methods Example



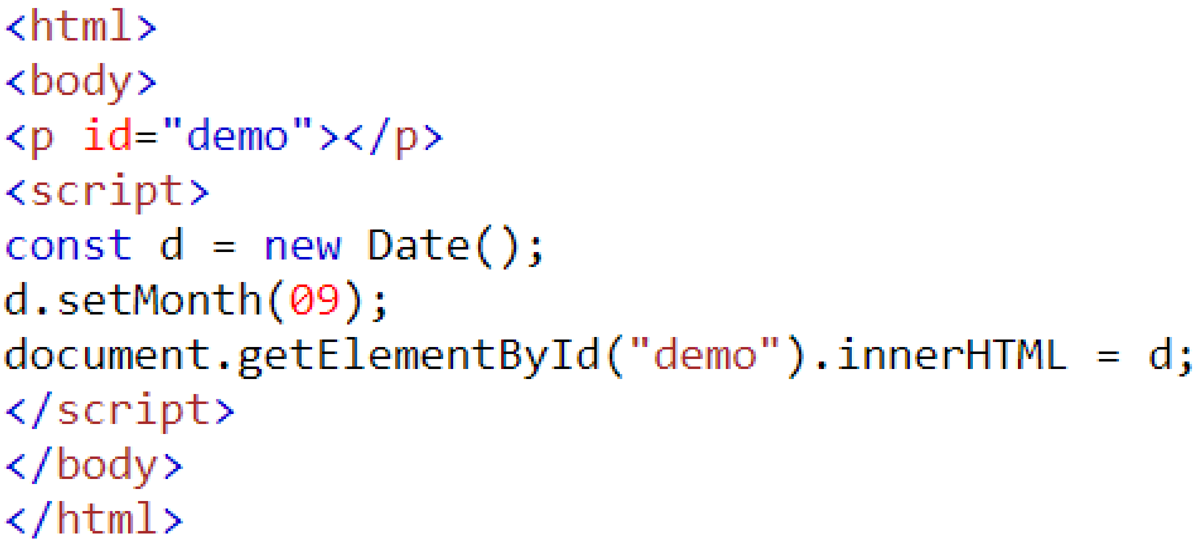
```
<html>
<body>
<p id="demo"></p>
<script>
const d = new Date();
document.getElementById("demo").innerHTML = d.getTime();
</script>
</body>
</html>
```

Date Methods

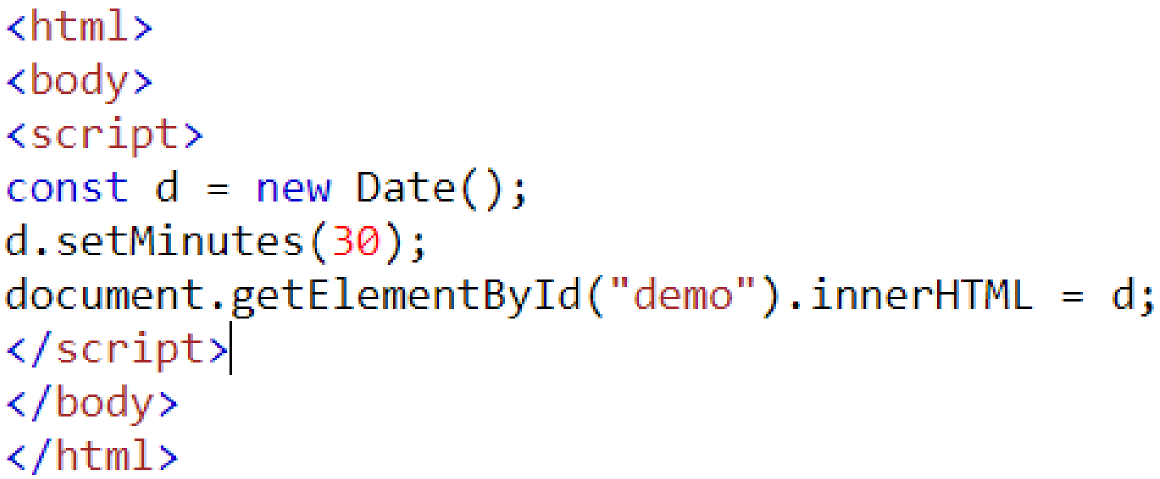
- Date set methods are:

Method	Description
setDate()	Set the day as a number (1-31)
setFullYear()	Set the year (optionally month and day)
setHours()	Set the hour (0-23)
setMinutes()	Set the minutes (0-59)
setMonth()	Set the month (0-11)
setSeconds()	Set the seconds (0-59)
setTime()	Set the time (milliseconds since January 1, 1970)

Date Set Methods Example



```
<html>
<body>
<p id="demo"></p>
<script>
const d = new Date();
d.setMonth(09);
document.getElementById("demo").innerHTML = d;
</script>
</body>
</html>
```



```
<html>
<body>
<script>
const d = new Date();
d.setMinutes(30);
document.getElementById("demo").innerHTML = d;
</script>
</body>
</html>
```

Math Objects

- The Math object allows you to perform mathematical tasks.
- The Math object includes several mathematical methods.
- The built-in Math object includes mathematical constants and functions.

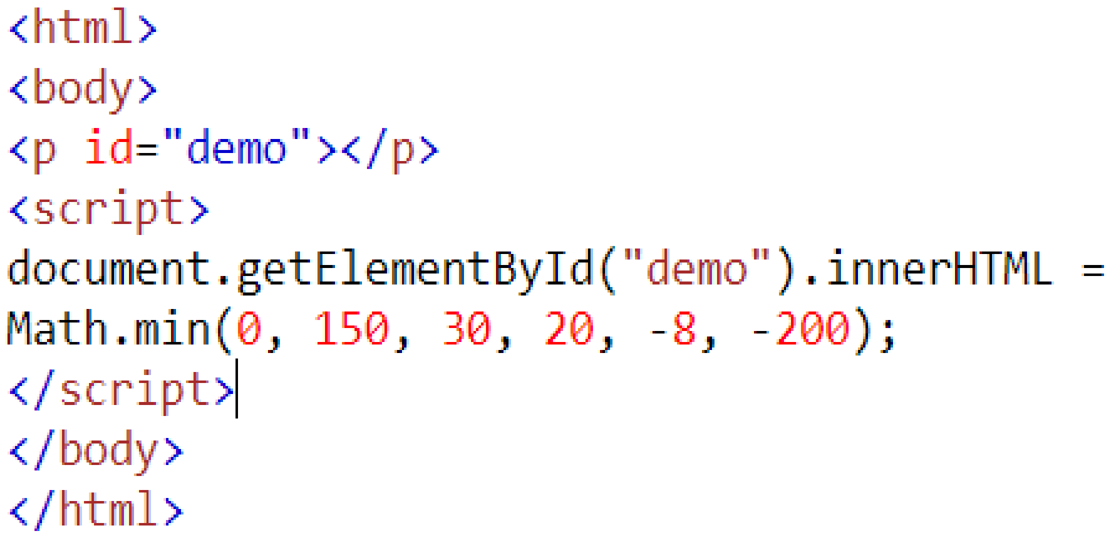
Functions of Math object

- Round
- Max number
- Min number

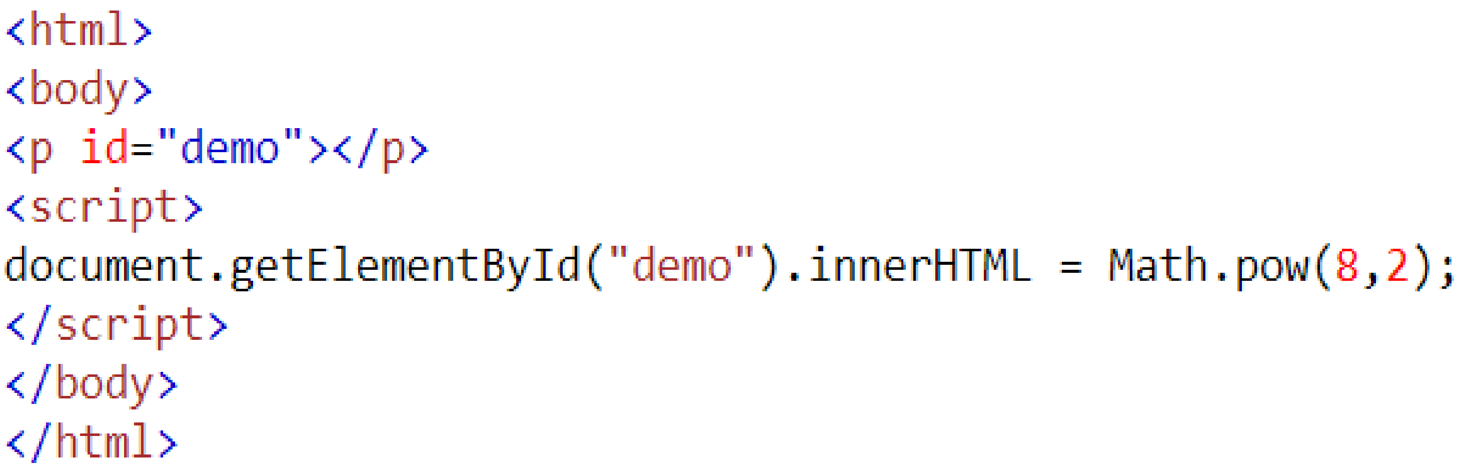
Math Objects

- **Math.PI** :
  - use to get the constant value 3.14159.
- **Math.abs(number):**
  - returns the absolute (positive) value of the number passed as a parameter value.
  - Eg -1 returns as 1
- **round()**
  - **Math.round(44.5) :** returns 45
- **Pow**
  - **Math.pow(2,8)** : result will be 256

Math Objects



```
<html>
<body>
<p id="demo"></p>
<script>
document.getElementById("demo").innerHTML =
Math.min(0, 150, 30, 20, -8, -200);
</script>
</body>
</html>
```



```
<html>
<body>
<p id="demo"></p>
<script>
document.getElementById("demo").innerHTML = Math.pow(8,2);
</script>
</body>
</html>
```

Questions…