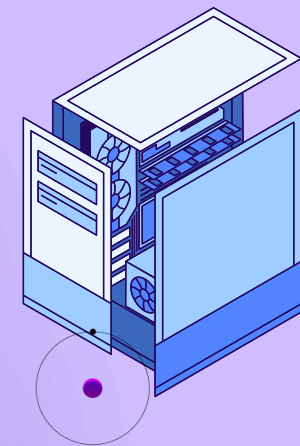
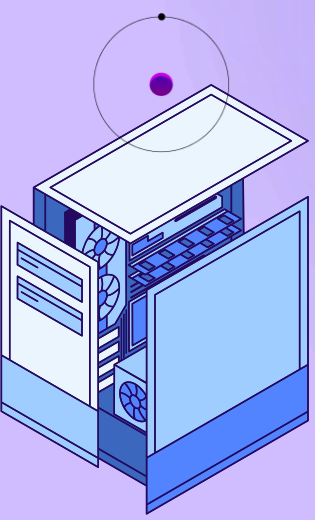


TLE/ICT 9

Third Quarter

Lesson 6

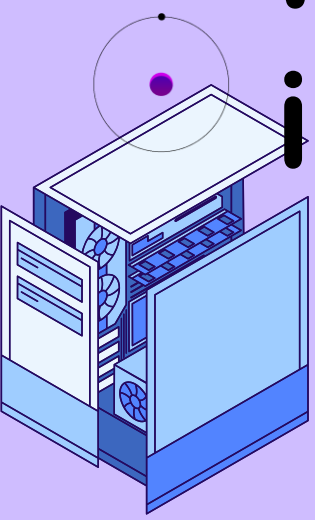




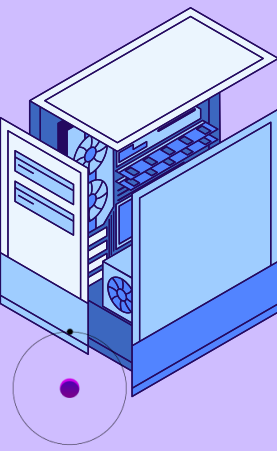
Starting JavaScript



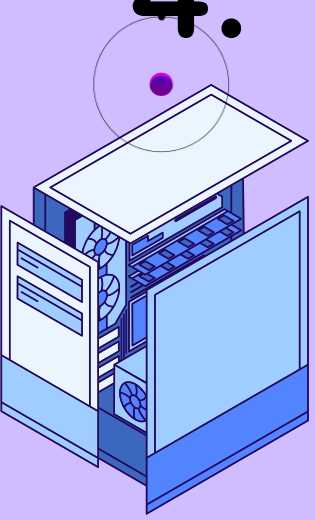
- `<script></script>` is used to connote JavaScript.
- We use the *type* attribute to define the scripting language and identify its version, however it is only optional.
- If the browser does not support JS, it will display JS as page content. To prevent this from happening, enclose the code in comments instead.
- HTML tells the browser how to display the text and image in the format you desired.



Starting JavaScript



- If you are coding JS and want to display an output in the Web page, the *document.write* command should be used.
 1. Open Notepad and type the code.
 2. Click File and select Save As.
 3. Determine the location you want to save your file.
 4. Give your file a filename ending in .html.

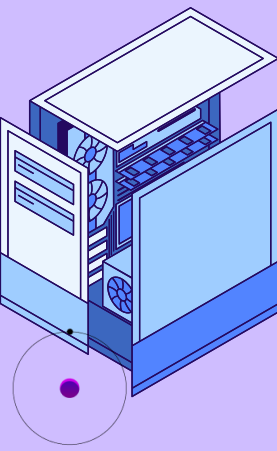




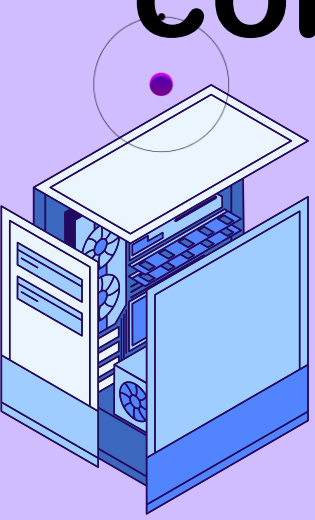
```
<html>
<head>
<title>It's JavaScript time</title>
</head>
<body>
<h1>JavaScript Sample</h1>
  <script type="text/javascript">
    <!--
      document.write("Displaying text using JavaScript");
    //-->
  </script>
</body>
</html>
```



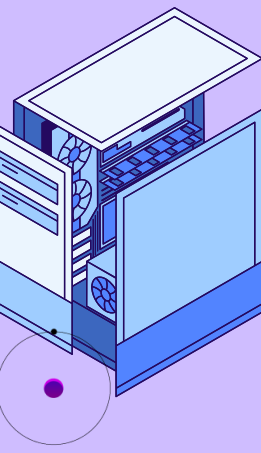
Starting JavaScript



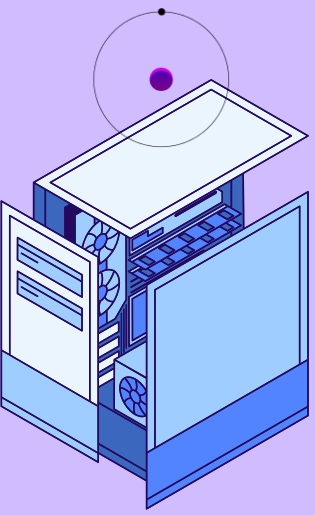
In the example given, `<script type="text/javascript">` and `</script>` tells the JS where to start and where to end. Remember that in HTML, the tags should have a beginning and an end to separate it from the other elements. If the *script* tags were not declared, the web browser will consider the *document.write* command as pure text and will be displayed as is.



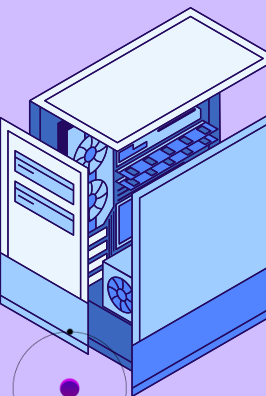
Starting JavaScript



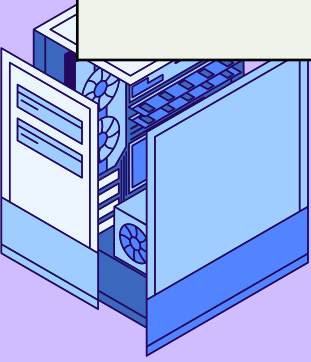
In the previous lesson, JS is an object-based language. Every object has its own method or property. Example is the *document.write("Hi");* .. *document* is the object name, *write* is the method and *"Hi"* is the parameter that holds the instructions.



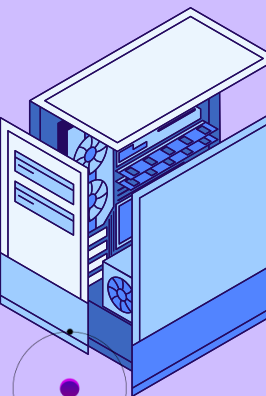
Commonly used Properties and Methods



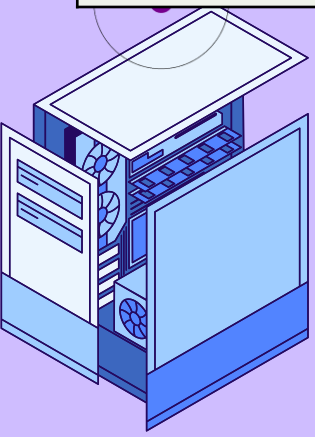
Object	Commonly Used Properties and Methods
window	<p>The <i>window</i> object is the main object. Think of it, as if you are pertaining to your whole browser. Many of the window-object properties are objects in their own right such as document, frame, and location. There is no need to write the object name window, since it is understood.</p> <p>Sample methods:</p> <ul style="list-style-type: none"><i>location</i> – to go to a new location<i>open</i> – to open a new window<i>setTimeout</i> – to set a time interval before activating an event



Commonly used Properties and Methods



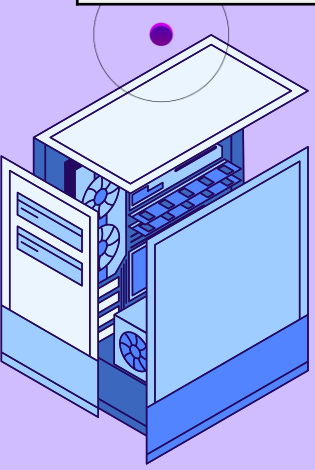
Object	Commonly Used Properties and Methods
document	<p>The <i>document</i> object is derived from the window object. It is the container for all HTML, HEAD, and BODY objects associated within the HTML tags of the HTML documents. You need to write the object name to use these methods.</p> <p>Sample methods:</p> <ul style="list-style-type: none"><i>fgColor</i> – to set the document foreground color<i>cookie</i> – to read the information stored in a cookie text file<i>write</i> – to display a message<i>lastModified</i> – to display the date it was last modified.



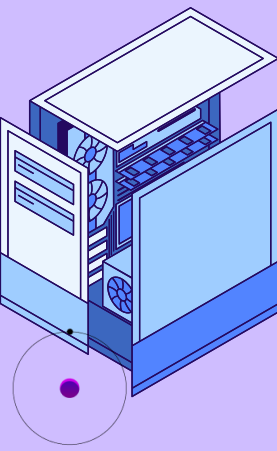
Commonly used Properties and Methods



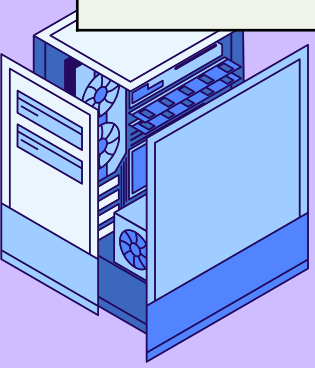
Object	Commonly Used Properties and Methods
math	<p>The <i>math</i> object provides the capability to perform math operations. You need to write the object name to use these methods.</p> <p>Sample methods:</p> <ul style="list-style-type: none"><i>PI</i> – has the value of PI<i>pow(a,b)</i> – takes the value of a to the power of b<i>max(a,b)</i> – returns the larger value between a and b<i>min(a,b)</i> – returns the smaller value between a and b



Commonly used Properties and Methods

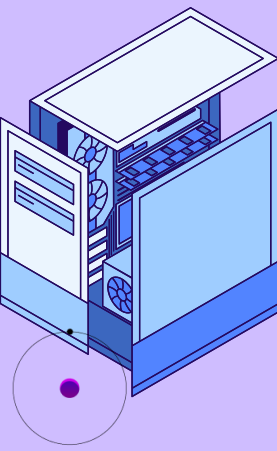


Object	Commonly Used Properties and Methods
navigator	<p>The <i>navigator</i> object determines which browser is running. You need to write the object name to use these methods.</p> <p>Sample methods:</p> <ul style="list-style-type: none"><i>appName</i> – to determine the browser's code name<i>appVersion</i> – to determine the browser's release version<i>cookieEnabled</i> – to determine whether cookies are enabled or not<i>platform</i> – a description of the operating system



Syntax for creating an event

<tag attribute eventHandler="function/javascript code">



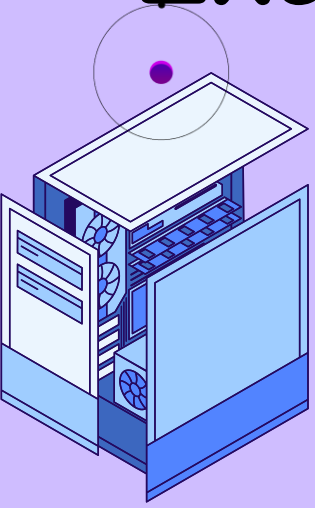
Event – is the result of a user's action; are objects with properties.

Event handlers – are the way to connect that action to a function or a set of JS codes to be executed.

Loading of an HTML document, mouse clicks, and even keyboard press are examples of events.

Example of an event:

`<input type="text" onclick="displaySum()">`

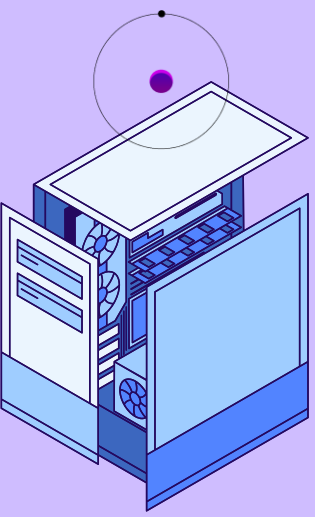




Where can you place Scripts?

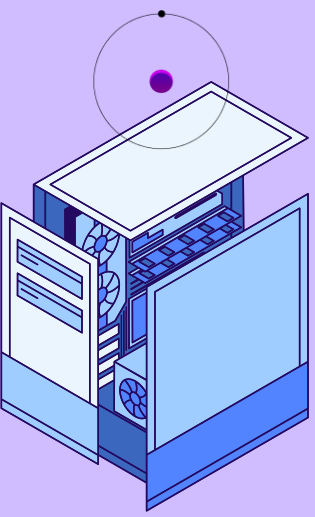
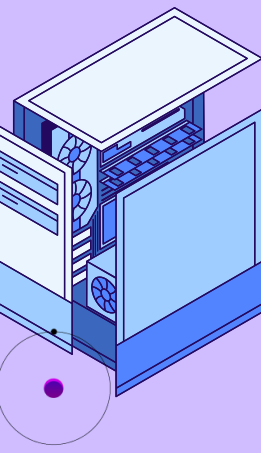
There are times when we want to execute the codes depending on the situation. It can be pre-loaded, or be triggered whenever an event happens. This is possible.


You can control the execution of the codes by placing it in different sections of the HTML file.



- **Head**

Whenever the scripts are placed here, you are assured that they will be pre-loaded – it means the scripts will be executed before anyone triggers an event. Scripts that are ideally place here are function calls.



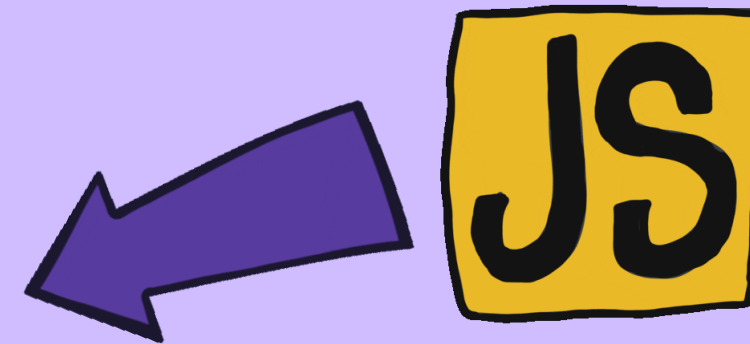
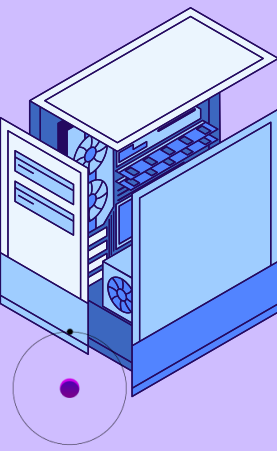
 head.txt
File Edit View

```
<html>
<head><title>Head</title>
  <script type="text/javascript">
    function intro()
      {alert("Welcome Visitor!");}
  </script>
</head>
<body onload="intro()">
</body>
</html>
```



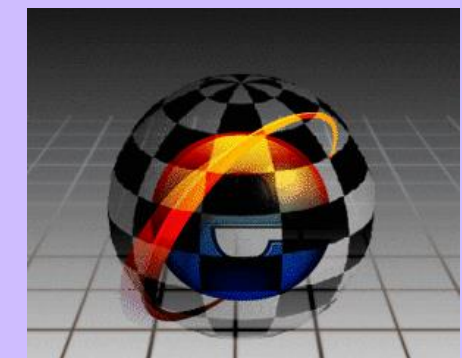
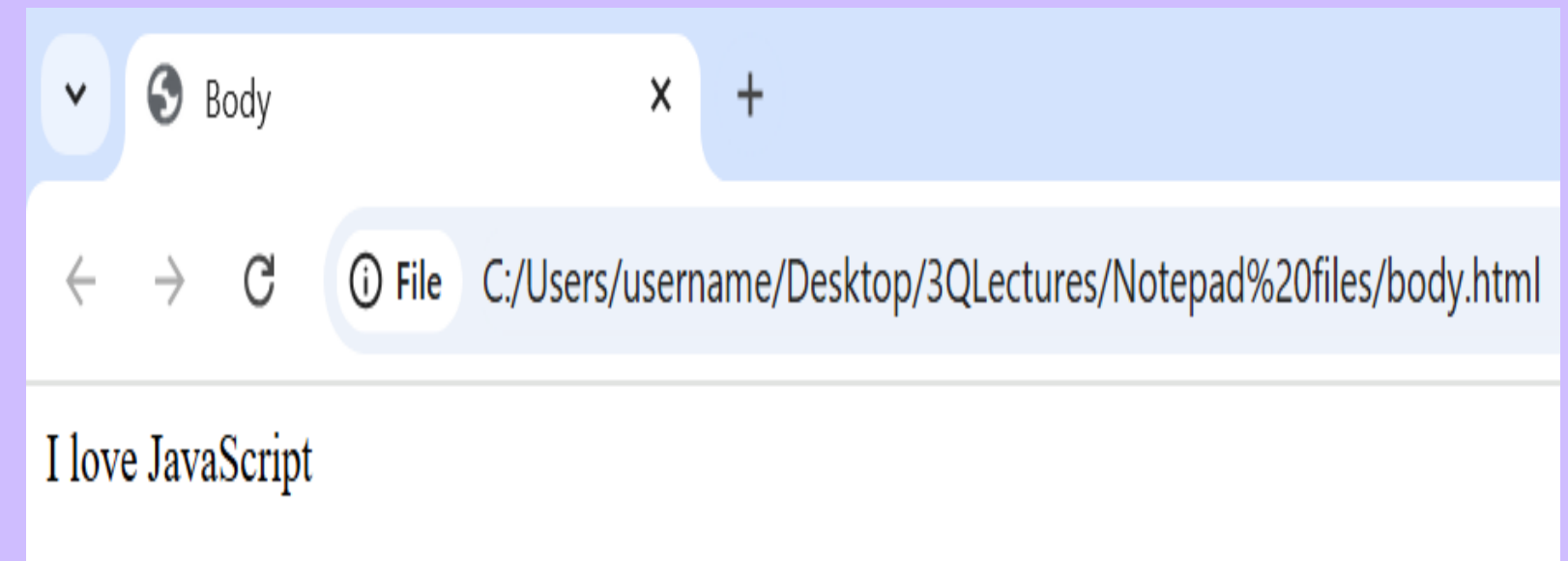
• Body

Scripts that are placed here are executed when the page loads.



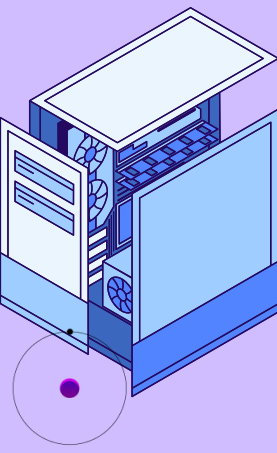
```
body.txt
File Edit View

<html>
<head><title>Body</title></head>
<body>
<script type="text/javascript">
    document.write("I love JavaScript");
</script>
</body>
</html>
```



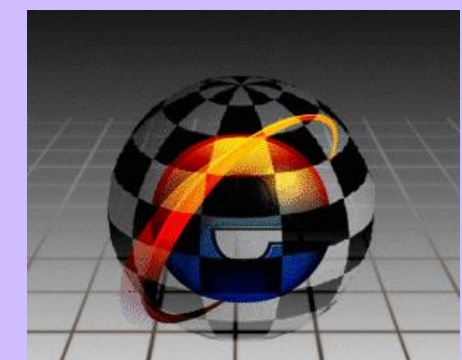
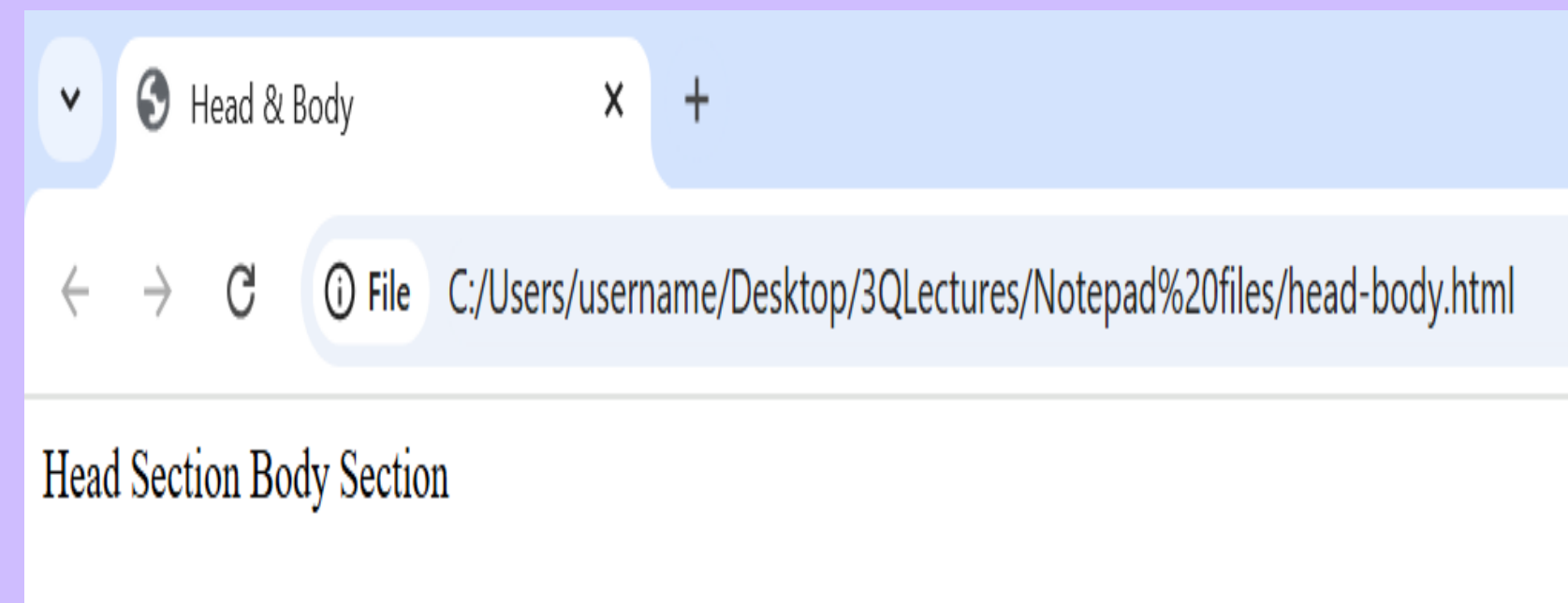
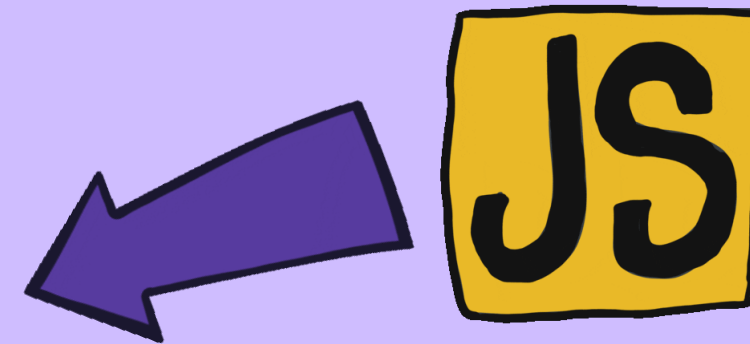
• Head and Body

You can place your script tags whenever and how many you want to.

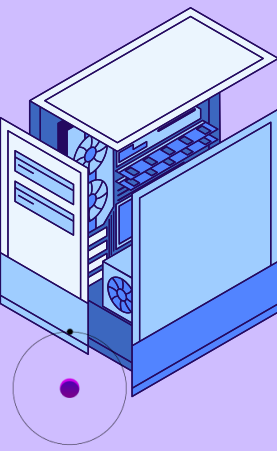


```
head-body.txt
File Edit View

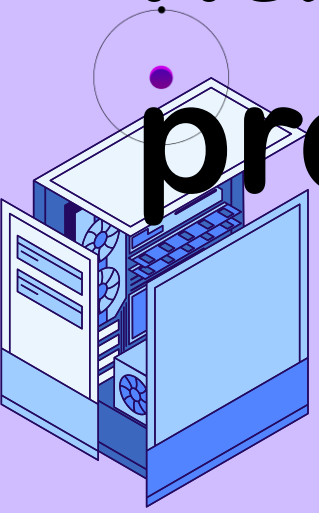
<html>
<head><title>Head & Body</title>
  <script type="text/javascript">
    document.write("Head Section");
  </script>
</head>
<body>
  <script type="text/javascript">
    document.write("Body Section");
  </script>
</body>
</html>
```



- External JavaScript



External JS files are helpful especially if you are using them to control different HTML files. There is no need to rewrite them all over again. Save the file having an extension of `.js`. To use the JS file, all you have to do is use the *src* attribute and provide the destination of the file.



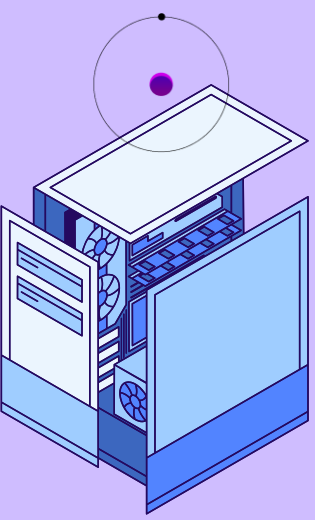
- **External JavaScript**



Syntax:

<script type="text/javascript" src="myEvents.js"></script>

The src attribute will point to the source of the JS external file. Take note that the JS file does not contain the opening and closing *script* tags.





comment.txt

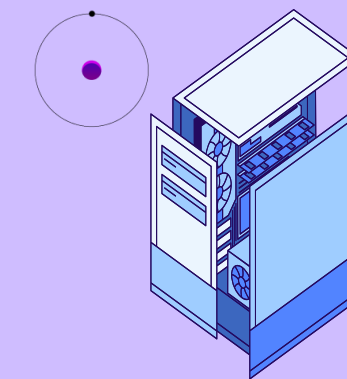


File

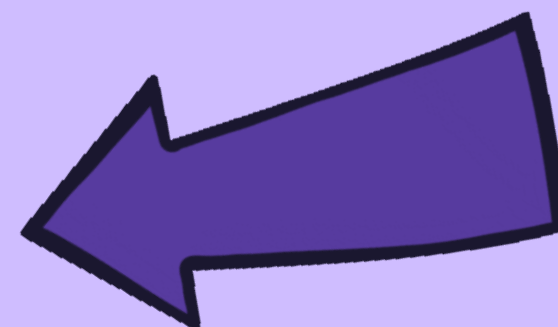
Edit

View

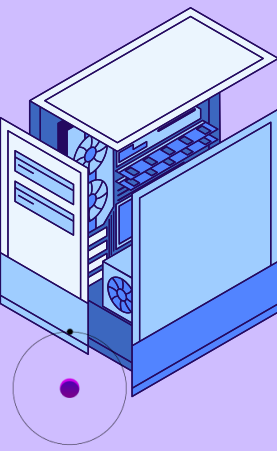
```
<html>
<head><title>Comments</title></head>
<body>
  <script type="text/javascript">
    //This is a single line comment
    document.write("My Comment");
    /* This is a multiple line comment*/
    document.write("Their Comment");
  </script>
</body>
</html>
```



There are times when you want to add a statement to explain a set of codes, or when you want to omit certain codes for debugging purposes, this can be done by inserting a comment. For single line comments you can use two backslash symbols; for multiple line or block comments, start it with a /* and end it with a */.

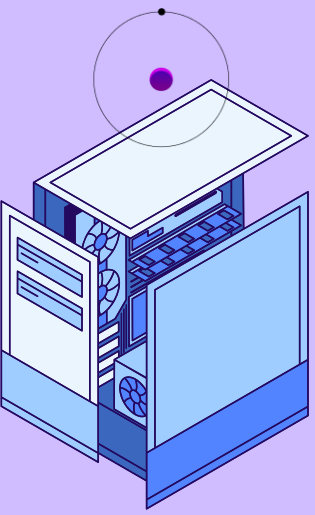


What are the JavaScript Guidelines?

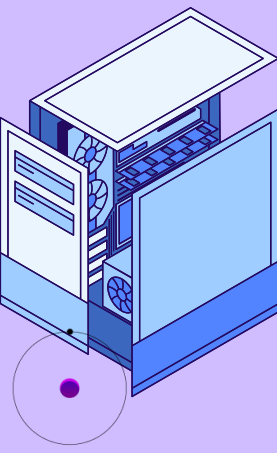


In coding JS, it is important to know the Do's and Don'ts. This will save you time and keep you away from trouble in case errors occur. Below are the guidelines.

1. Case sensitive – you must be careful in naming you variables. The variable *myHome* is not equal to *MyHome*. It is important to keep track of your variables so that they won't be problems in calling variables, objects, or functions. Unlike HTML, uppercase letters are not equivalent to lowercase letters in JS.



What are the JavaScript Guidelines?

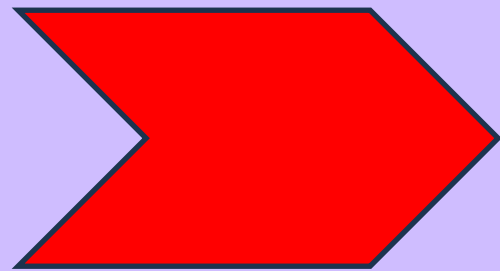


2. White Space – JS ignores white space as long as it is within the line of the code. It is a good practice to use white spaces between expressions to make your codes easier to read.

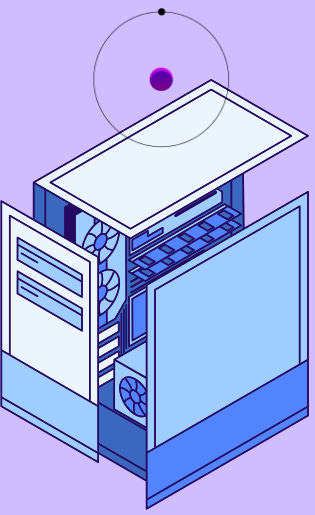
Example:

```
var name = "Christy"
```

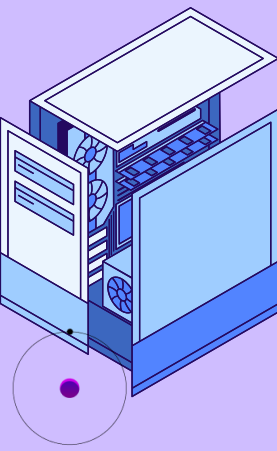
```
var name="Christy"
```



These two variable declarations are accepted.



What are the JavaScript Guidelines?



3. Breaking up of code lines – you can break a code line by using a backslash ('\'). However, do not break the code in between commands or else it will not work properly.

Example:

<code>document.write("wow \ Philippines");</code>	Accepted
<code>document.\write("wow \ Philippines")</code>	Not Accepted

