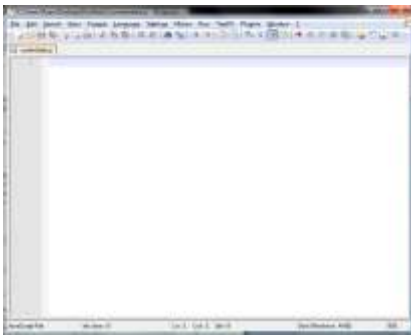


Creating a Current Date JavaScript

Purpose: To display the user's system date on the web page. This allows them to see the full date as they are browsing your website.

Issues: mobile devices may not be able to read the script, users may have browser scripting turned off, browser may be too old to read JavaScript, administration rights may interfere with the running of the script

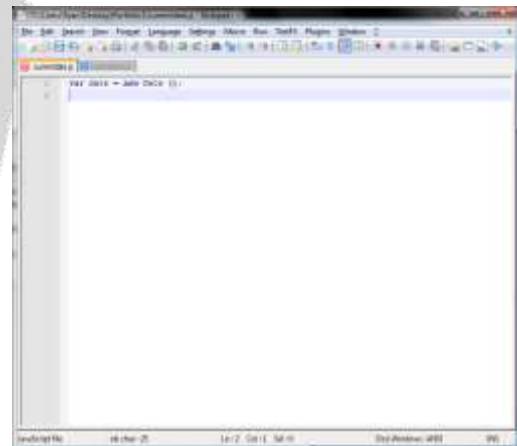


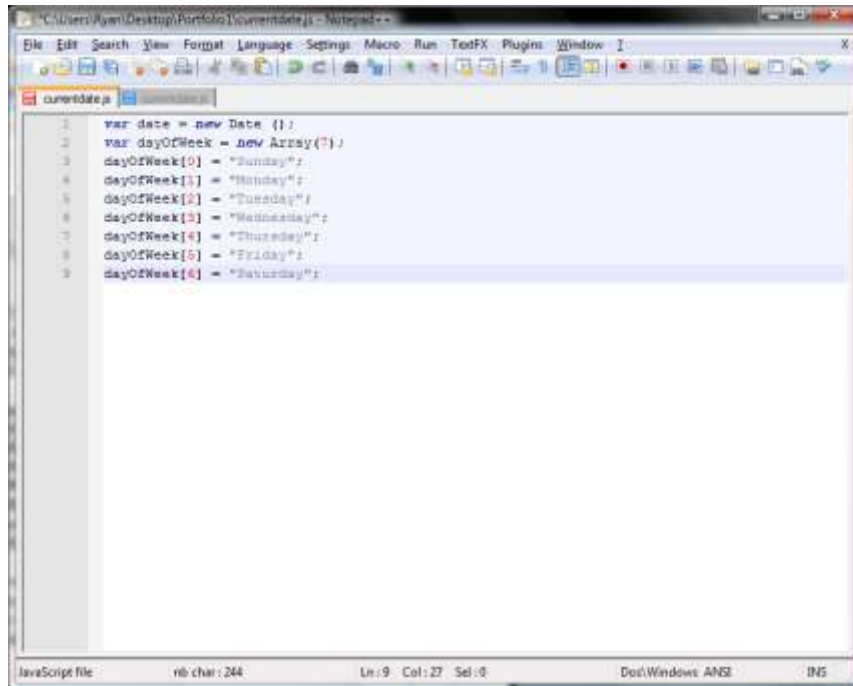
Step 1: Start with a blank JavaScript File, to create a blank JavaScript file open a blank text document and save it as “currentdate.js” the .js file extension makes it a JavaScript File.

Step 2: The key element to any JavaScript that is working with a date is a predefined function called **new Date ()** we have to make a variable that holds the read out of this predefined function. To do this we write the following code:

```
var date = new Date ( );
```

This allows us to reference the readout of the function by just “date”, the “var” defines a variable.





```

1  var date = new Date ();
2  var dayOfWeek = new Array(7);
3  dayOfWeek[0] = "Sunday";
4  dayOfWeek[1] = "Monday";
5  dayOfWeek[2] = "Tuesday";
6  dayOfWeek[3] = "Wednesday";
7  dayOfWeek[4] = "Thursday";
8  dayOfWeek[5] = "Friday";
9  dayOfWeek[6] = "Saturday";

```

Step 3: Now we need to create an Array to change the number from the predefined function **new Date ()**; it display the day of the week as a integer (0 through 6), to replace these we need to make an array to fix that. We make an array with the following code:

```

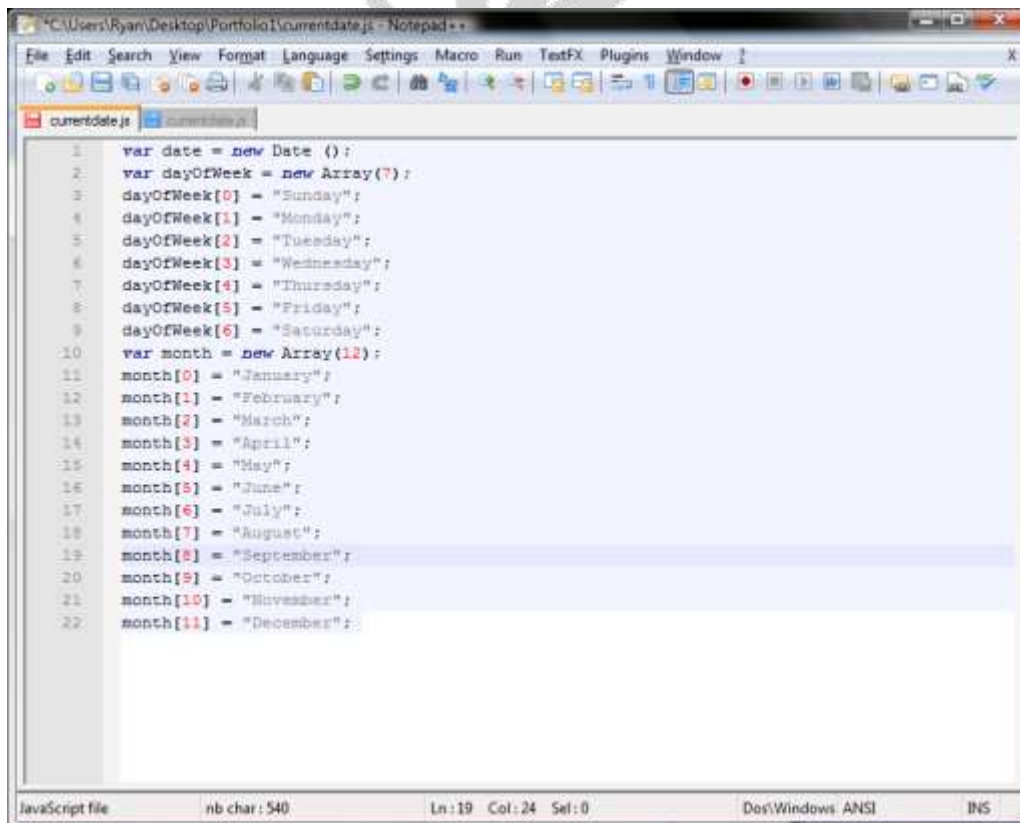
var dayOfWeek = new Array(7);
  dayOfWeek[0] = "Sunday";
  dayOfWeek[1] = "Monday";
  dayOfWeek[2] = "Tuesday";
  dayOfWeek[3] = "Wednesday";
  dayOfWeek[4] = "Thursday";
  dayOfWeek[5] = "Friday";
  dayOfWeek[6] = "Saturday";

```

As we look at the piece of code above this will help assign the name of the days of the weeks to predefined function. We store the array in a variable again and access it by "dayOfWeek". As we look at the array in the brackets it is the integer defined by **new Date ()**, and will substitute the number for the value of that specific number. We enclose the days of week in quotations because they are character data.

Step 4: Now we need to create a second Array for the months since it displays the months as a integer as well. This is done in the same way only this time we will name the variable month and type the months of the year instead of days of the week.

```
var month = new Array(12);  
    month[0] = "January";  
    month[1] = "February";  
    month[2] = "March";  
    month[3] = "April";  
    month[4] = "May";  
    month[5] = "June";  
    month[6] = "July";  
    month[7] = "August";  
    month[8] = "September";  
    month[9] = "October";  
    month[10] = "November";  
    month[11] = "December";
```



The screenshot shows a Notepad++ window titled "C:\Users\Ryan\Desktop\Portfolio1\currentdate.js - Notepad++". The code in the window is as follows:

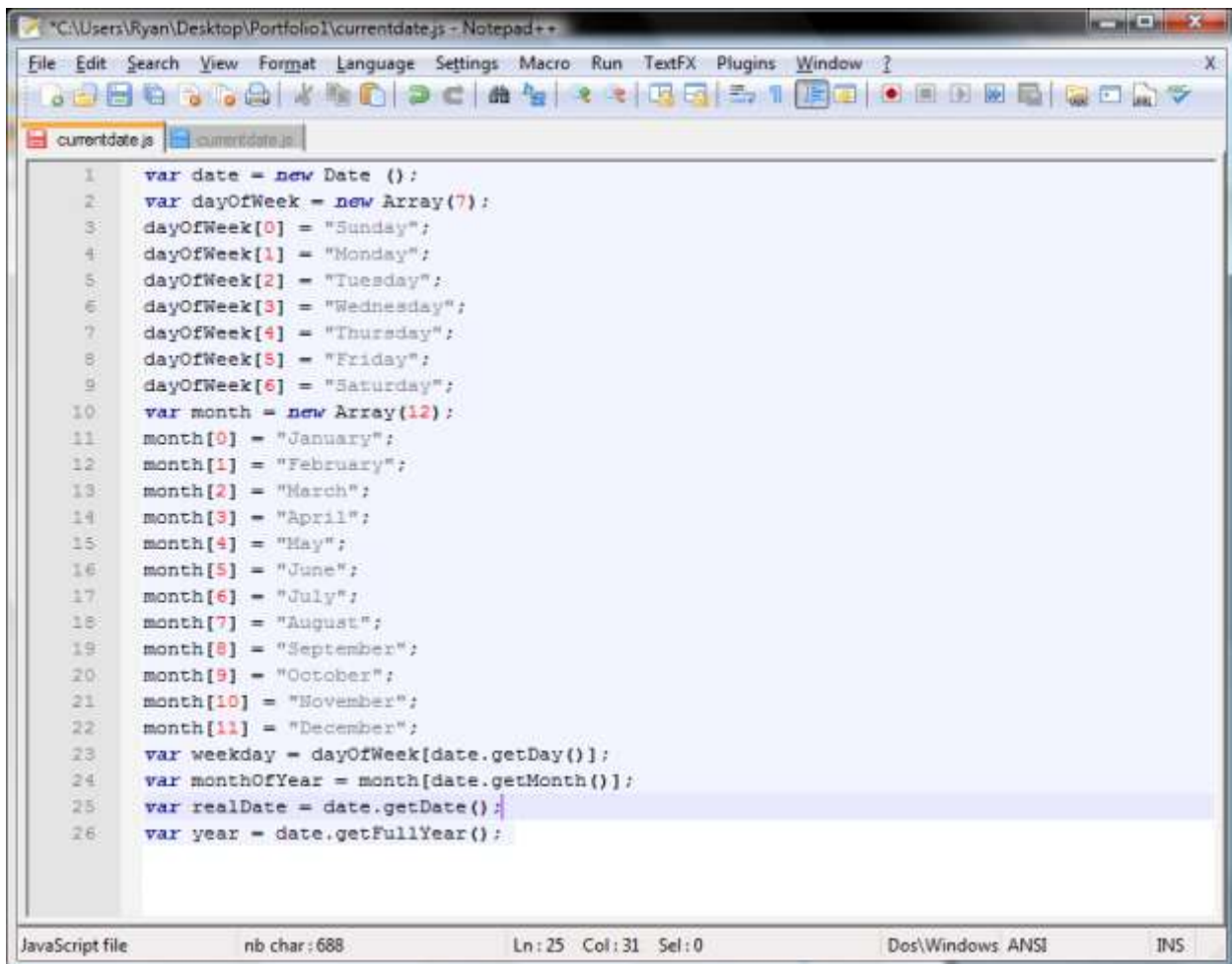
```
1  var date = new Date ();  
2  var dayOfWeek = new Array(7);  
3  dayOfWeek[0] = "Sunday";  
4  dayOfWeek[1] = "Monday";  
5  dayOfWeek[2] = "Tuesday";  
6  dayOfWeek[3] = "Wednesday";  
7  dayOfWeek[4] = "Thursday";  
8  dayOfWeek[5] = "Friday";  
9  dayOfWeek[6] = "Saturday";  
10 var month = new Array(12);  
11 month[0] = "January";  
12 month[1] = "February";  
13 month[2] = "March";  
14 month[3] = "April";  
15 month[4] = "May";  
16 month[5] = "June";  
17 month[6] = "July";  
18 month[7] = "August";  
19 month[8] = "September";  
20 month[9] = "October";  
21 month[10] = "November";  
22 month[11] = "December";
```

The status bar at the bottom indicates "JavaScript file", "nb char: 540", "Ln: 19 Col: 24 Sel: 0", "Dos/Windows ANSI", and "INS".

Step 5: Now we need to attach the arrays we wrote to the new Date (); function. To do this we use the following code.

```
var weekday = dayOfWeek[date.getDay()];  
var monthOfYear = month[date.getMonth()];  
var realDate = date.getDate();  
var year = date.getFullYear();
```

The code above attaches the array to the **new Date()**; Function and get the day of the month and the Year.

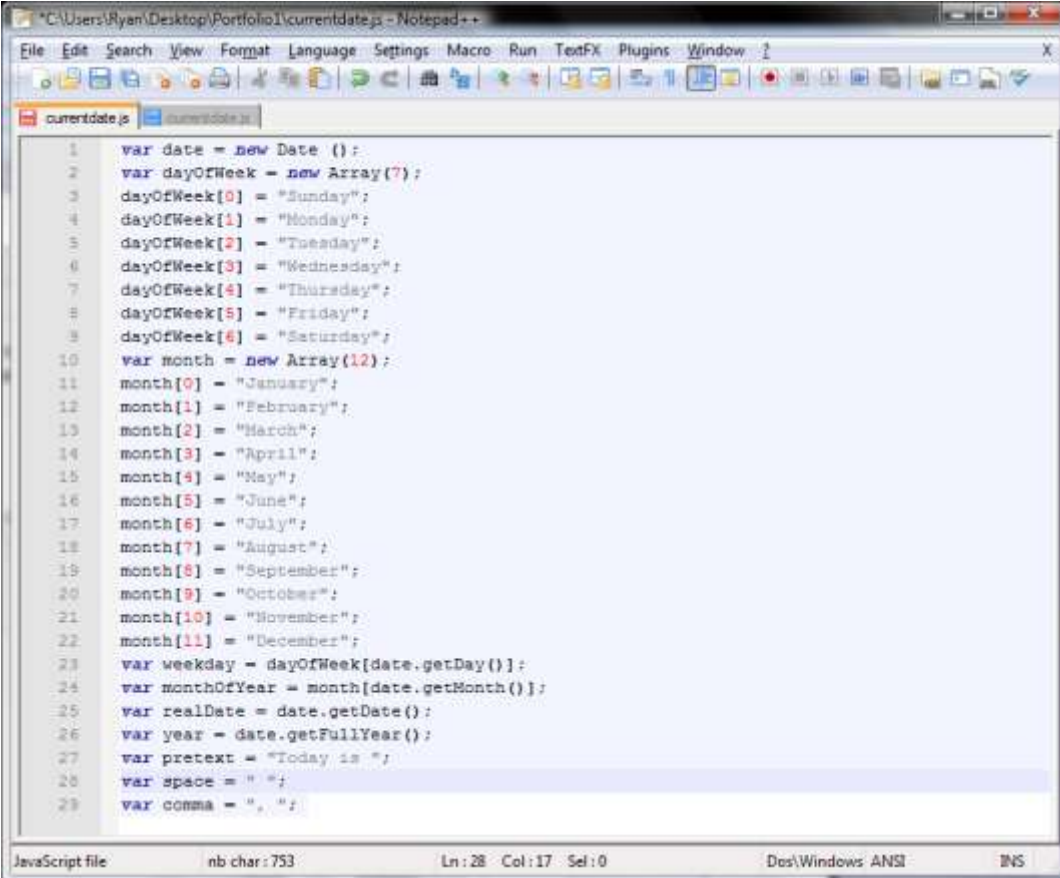


```
*C:\Users\Ryan\Desktop\Portfolio\currentdate.js - Notepad++  
File Edit Search View Format Language Settings Macro Run TextFX Plugins Window ? X  
currentdate.js  
1 var date = new Date ();  
2 var dayOfWeek = new Array(7);  
3 dayOfWeek[0] = "Sunday";  
4 dayOfWeek[1] = "Monday";  
5 dayOfWeek[2] = "Tuesday";  
6 dayOfWeek[3] = "Wednesday";  
7 dayOfWeek[4] = "Thursday";  
8 dayOfWeek[5] = "Friday";  
9 dayOfWeek[6] = "Saturday";  
10 var month = new Array(12);  
11 month[0] = "January";  
12 month[1] = "February";  
13 month[2] = "March";  
14 month[3] = "April";  
15 month[4] = "May";  
16 month[5] = "June";  
17 month[6] = "July";  
18 month[7] = "August";  
19 month[8] = "September";  
20 month[9] = "October";  
21 month[10] = "November";  
22 month[11] = "December";  
23 var weekday = dayOfWeek[date.getDay()];  
24 var monthOfYear = month[date.getMonth()];  
25 var realDate = date.getDate();  
26 var year = date.getFullYear();  
JavaScript file nb char: 688 Ln: 25 Col: 31 Sel: 0 Dos\Windows ANSI INS
```

Step 6: We need to create three variables to hold a space, a comma, and pretext. This will help format the output of the JavaScript File. This is done with the following code:

```
var comma = " , ";  
var pretext = "Today is";  
var space = " ";
```

These create three variables to add formatting marks for our output that we write in the next step.

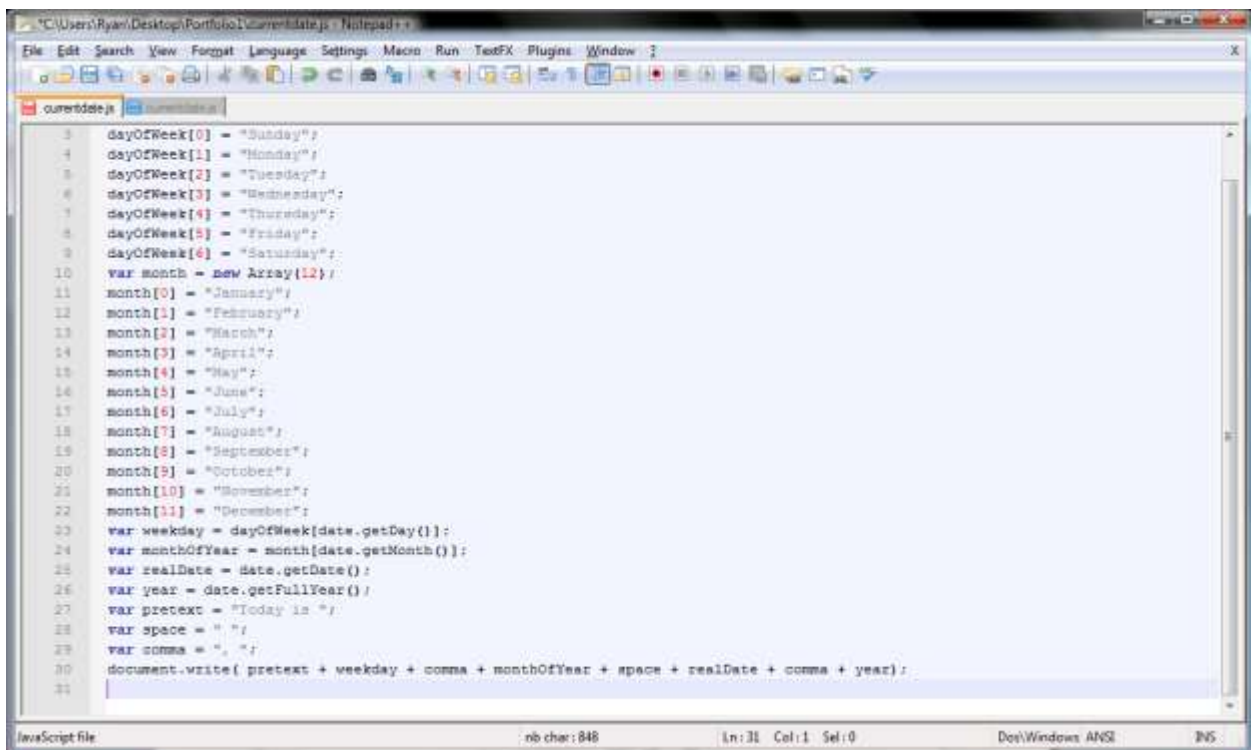


```
*C:\Users\Ryan\Desktop\Portfolio\currentdate.js - Notepad++  
File Edit Search View Format Language Settings Macro Run TextFX Plugins Window  
currentdate.js  
1 var date = new Date ();  
2 var dayOfWeek = new Array (7);  
3 dayOfWeek[0] = "Sunday";  
4 dayOfWeek[1] = "Monday";  
5 dayOfWeek[2] = "Tuesday";  
6 dayOfWeek[3] = "Wednesday";  
7 dayOfWeek[4] = "Thursday";  
8 dayOfWeek[5] = "Friday";  
9 dayOfWeek[6] = "Saturday";  
10 var month = new Array (12);  
11 month[0] = "January";  
12 month[1] = "February";  
13 month[2] = "March";  
14 month[3] = "April";  
15 month[4] = "May";  
16 month[5] = "June";  
17 month[6] = "July";  
18 month[7] = "August";  
19 month[8] = "September";  
20 month[9] = "October";  
21 month[10] = "November";  
22 month[11] = "December";  
23 var weekday = dayOfWeek[date.getDay()];  
24 var monthOfYear = month[date.getMonth()];  
25 var realDate = date.getDate();  
26 var year = date.getFullYear();  
27 var pretext = "Today is ";  
28 var space = " ";  
29 var comma = " , ";  
JavaScript file nb char : 753 Ln : 28 Col : 17 Sel : 0 Dos\Windows ANSI INS
```

Step 7: Finally we create the output. We do this with a line of code **document.write()** this write to the webpage the variables and text inside it. We write the date output as follows:

```
document.write( pretext + weekday + comma + monthOfYear + space + realDate +  
comma + year);
```

We have to separate each variable with a plus sign so they are not confused as one long variable to statement. If you wanted to add text you would just put the text in quotes to get a readout of what was in the quotes.

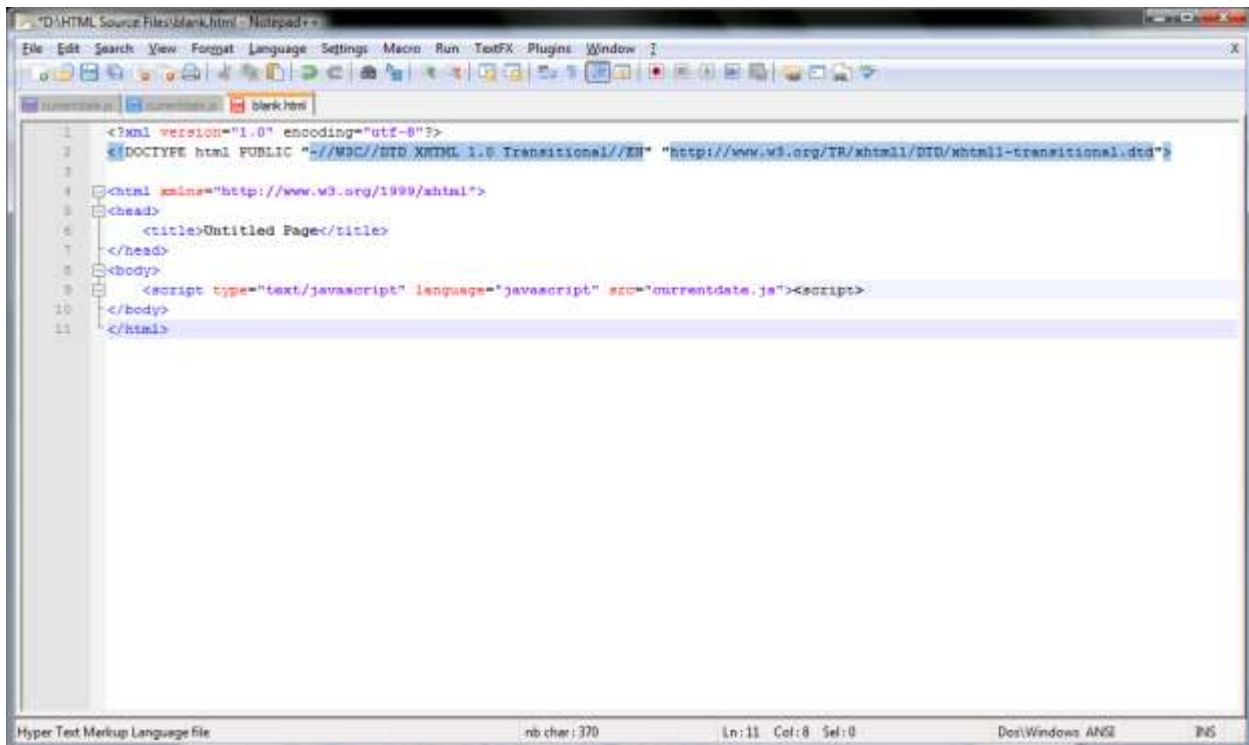


```
"C:\Users\Ryan\Desktop\Portfolio\CurrentDate.js - Notepad++
File Edit Search View Format Language Settings Macro Run TextFX Plugins Window ?
currentDate.js
3  dayOfWeek[0] = "Sunday";
4  dayOfWeek[1] = "Monday";
5  dayOfWeek[2] = "Tuesday";
6  dayOfWeek[3] = "Wednesday";
7  dayOfWeek[4] = "Thursday";
8  dayOfWeek[5] = "Friday";
9  dayOfWeek[6] = "Saturday";
10 var month = new Array(12);
11 month[0] = "January";
12 month[1] = "February";
13 month[2] = "March";
14 month[3] = "April";
15 month[4] = "May";
16 month[5] = "June";
17 month[6] = "July";
18 month[7] = "August";
19 month[8] = "September";
20 month[9] = "October";
21 month[10] = "November";
22 month[11] = "December";
23 var weekday = dayOfWeek[date.getDay()];
24 var monthOfYear = month[date.getMonth()];
25 var realDate = date.getDate();
26 var year = date.getFullYear();
27 var pretext = "Today is ";
28 var space = " ";
29 var comma = ",";
30 document.write( pretext + weekday + comma + monthOfYear + space + realDate + comma + year);
31
JavaScript file          nb char: 848          ln: 31 Col: 1 Sel: 0          Doc/Window: ANS          INS
```

Step 8: Now we need to attach the JavaScript File to the webpage and it will display the date on the webpage. This done by adding the following code.

```
<script type="text/javascript" language="javascript" src="currentdate.js"></script>
```

This is how to attach an external JavaScript to a webpage. When using JavaScript, a `<noscript></noscript>` should follow the JavaScript. What you place inside the noscript tags is up to you but should say something to the effect to enable JavaScript.



The screenshot shows a Notepad++ window titled "D:\HTML Source Files\blank.html - Notepad++". The code in the editor is as follows:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
3
4 <html xmlns="http://www.w3.org/1999/xhtml">
5 <head>
6 <title>Untitled Page</title>
7 </head>
8 <body>
9 <script type="text/javascript" language="javascript" src="currentdate.js"></script>
10 </body>
11 </html>
```

The status bar at the bottom indicates "Hyper Text Markup Language file", "nb char: 370", "Ln: 11 Col: 8 Sel: 0", "Doc/Window: ANSE", and "INS".