Terna Engineering College
Computer Engineering Department

Program: Sem V

Course: Web Technology Laboratory (CSL504)

Faculty: Mrs Reshma Koli

LAB Manual

PART A

(PART A: TO BE REFERRED BY STUDENTS)

Experiment No.04

A.1 Aim:

Write a JavaScript function to calculate compound and simple interest of user provided input.

A.2 Prerequisite:

1. Knowledge of HTML
2. Knowledge of Client side scripting using JavaScript.

A.3 Outcome:

After successful completion of this experiment students will be able to

1. Design dynamic web pages using HTML5 and JavaScript.
2. Able to perform simple mathematical calculations at the client side using JavaScript.

A.4 Theory:

Formula to calculate Compound and Simple Interest

■ What is Simple Interest?

- The money borrowed is called the principal (P).
- Extra money paid back is called the simple interest (S.I).
- Interest is expressed as rate percent per annum (p.a.) i.e., 12% per month means the interest on $100 for 1 year is $12.
- The total money paid back after the given time is called the amount.
- Time for which money is borrowed is called the time period.

Here,

P = Principal

R = rate% per annum

T = time

I = simple interest

A = amount

Formula for calculating simple interest is **S.I = (P × R × T)/100**

Example on simple interest:

1. Find simple interest on $2000 at 5% per annum for 3 years. Also, find the amount.

Solution:

Principal = $2000

Rate = 5% p.a.

T = 3 years

S.I = (P × R × T)/100

= (2000 × 5 × 3)/100

= $ 300

Amount = P + I

= $ (2000 + 300)

= $ 2300

What is Compound Interest?

- The concept of **compound interest** is that interest is added back to the principal sum so that interest is earned on that added interest during the next compounding period.

- The formula for annual compound interest, including principal sum, is:
  $A = P(1 + r/n)^{(nt)}$

  Where:

  A = the future value of the investment/loan, including interest
  P = the principal investment amount (the initial deposit or loan amount)
  r = the annual interest rate (decimal)
  n = the number of times that interest is compounded per year
  t = the number of years the money is invested or borrowed for.

  Note that this formula gives you the future value of an investment or loan, which is compound interest **plus** the principal. Should you wish to calculate the compound interest only, you need this:

  Total compounded interest = $P(1 + r/n)^{(nt)} - P$

Examples on simple interest:

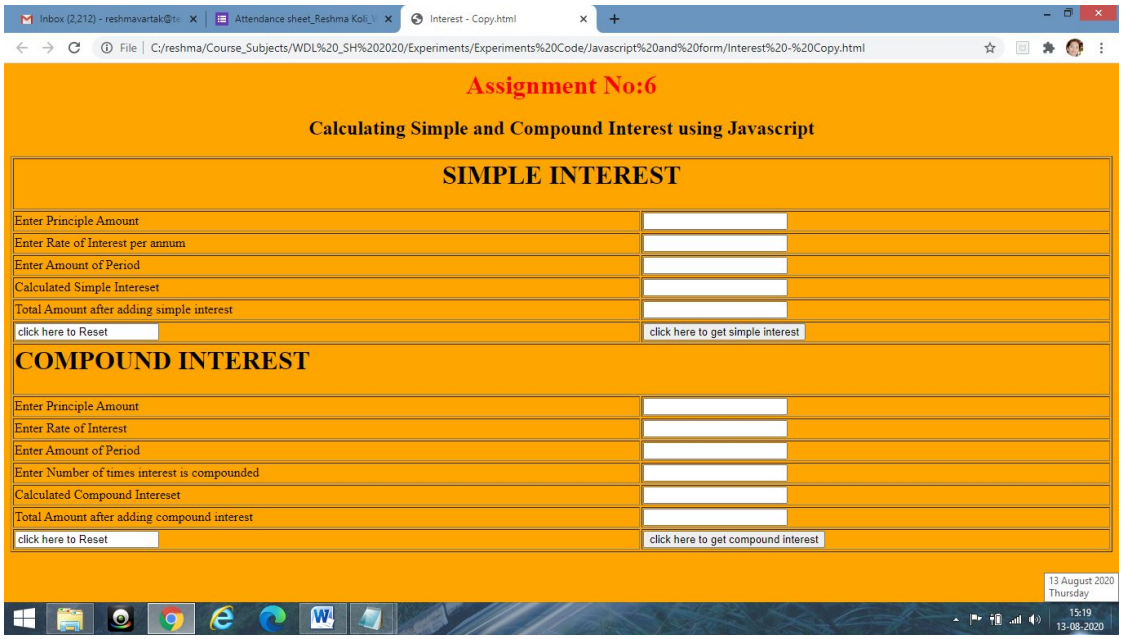
P = 5000. **r** = 5/100 = 0.05 (decimal). **n** = 12. **t** = 10.

If we plug those figures into the formula, we get the following (note that ^ indicates 'to the power of'):

A = 5000 (1 + 0.05 / 12) ^ (12(10)) = 8235.05.

So, the investment balance after 10 years is **$8,235.05**.

- **Sample Page Layout**



PART B

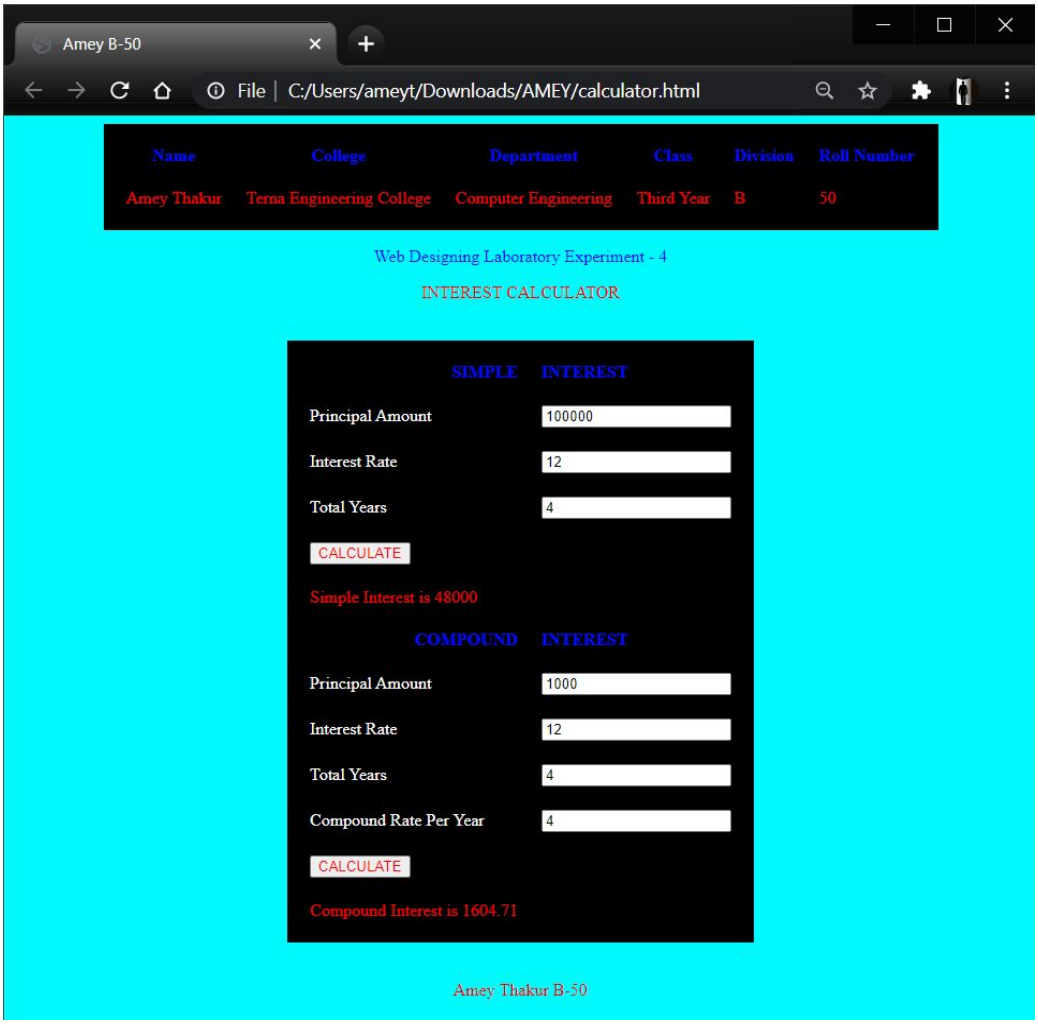
(PART B: TO BE COMPLETED BY STUDENTS)

(Students must submit the soft copy as per following segments within two hours of the practical. The soft copy must be uploaded on the Blackboard or emailed to the concerned lab in charge faculties at the end of the practical in case the there is no Blackboard access available)

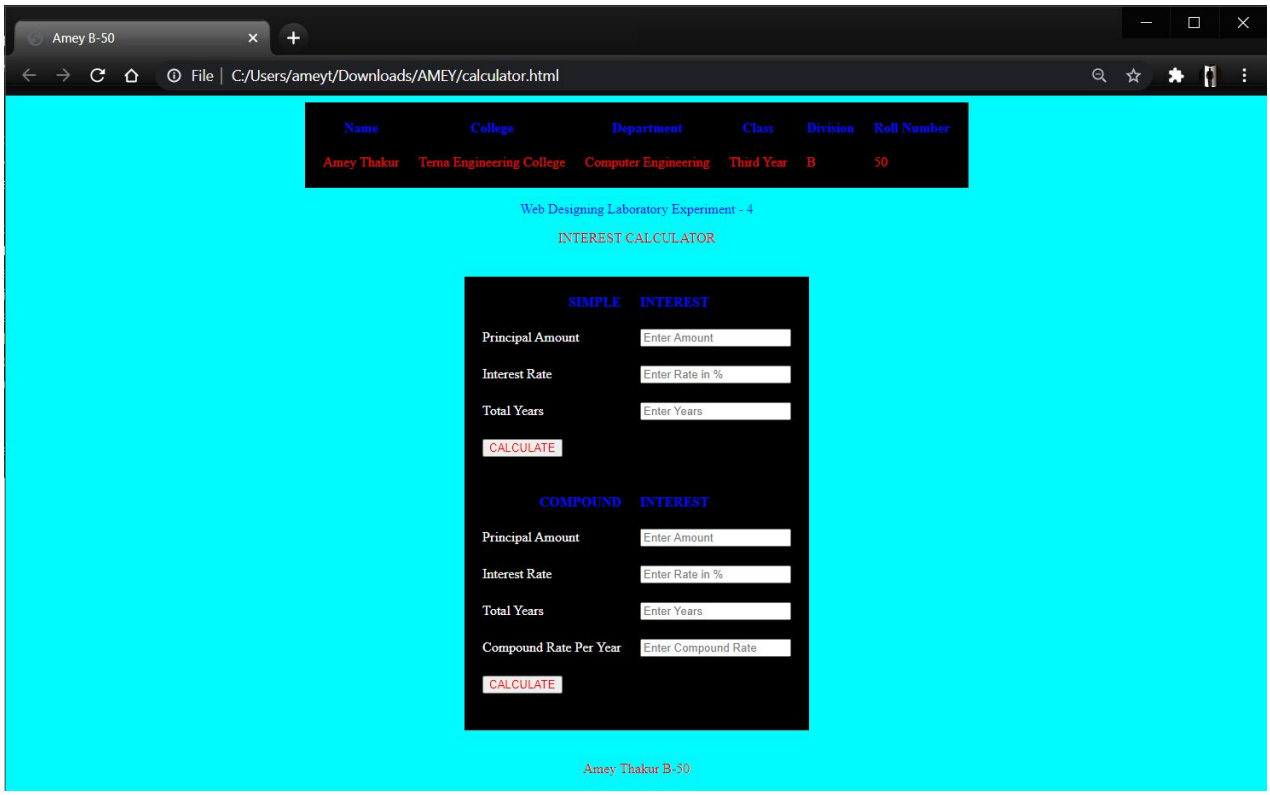
Roll No. 50	Name: Amey Thakur
Class: TE-Comps B	Batch: B3
Date of Experiment: 13/08/2020	Date of Submission: 13/08/2020
Grade :	

B.1 Web page Snapshot: (Add all snapshots of output.)

- Calculator



- Interest Calculator



B.2 Web page source code:

- Calculator.html

```
<DOCTYPE html>

<html lang="en" dir="ltr">

 <head>

  <meta charset="utf-8">

  <meta name="viewport" content="width=device-width">

  <meta name="description" content="calculator">

        <meta name="keywords" content="calculator">

      <meta name="author" content="Amey Thakur">

  <title>Amey B-50</title>

  <link rel="stylesheet" href="./css/style.css">
```

```
 <script src="javascript.js"></script>
 </head>
<body>
 <aside id="sidebar">
            <p>
            <table border="0" bgcolor="#000000" align="center" cellspacing= "20">
            <tr style="color:#000dff">
            <th >Name</th>
            <th>College</th>
            <th>Department</th>
            <th>Class</th>
            <th>Division</th>
            <th>Roll Number</th>
            </tr>
            <tr style="color:#ff0000">
            <td>Amey Thakur</td>
            <td>Terna Engineering College</td>
            <td>Computer Engineering</td>
            <td>Third Year</td>
            <td>B</td>
            <td>50</td>
            </tr>
            </table>
           </p>
           </aside>
```

```
        <center>
          <p style="color:#000dff">Web Designing Laboratory Experiment - 4</p>
          <p style="color:#ff0000">INTEREST CALCULATOR</p>
        </center>
<div class="ex">
 <table border="0" bgcolor="#000000" align="center" cellspacing= "20">
 <tr style="color:#000dff">
    <th align="right">SIMPLE</th>
    <th align="left">INTEREST</th>
 </tr>
 <tr style="color:#ffffff">
  <td>Principal Amount</td>
       <td><input type="number" placeholder="Enter Amount" size="25" id="sprincipal" name="sprincipal" required></td>
 </tr>
 <tr style="color:#ffffff">
  <td>Interest Rate</td>
   <td><input type="number" placeholder="Enter Rate in %" size="25" id="srate" name="srate" required></td>
 </tr>
 <tr style="color:#ffffff">
  <td>Total Years</td>
    <td><input type="number" placeholder="Enter Years" size="25" id="syear" name="syear" required></td>
 </tr>
 <tr>
```

```
  <td>
   <button style="color:#ff0000" onclick="myFunction1()">CALCULATE</button>
   </td>
 </tr>
 <tr>
  <td>
   <p id="SI"></p>
   </td>
 </tr>
<tr style="color:#000dff">
  <th align="right">COMPOUND</th>
  <th align="left">INTEREST</th>
 </tr>
 <tr style="color:#ffffff">
  <td>Principal Amount</td>
        <td><input type="number" placeholder="Enter Amount" size="25" id="cprincipal" name="sprincipal" required></td>
 </tr>
 <tr style="color:#ffffff">
  <td>Interest Rate</td>
   <td><input type="number" placeholder="Enter Rate in %" size="25" id="crate" name="srate" required></td>
 </tr>
 <tr style="color:#ffffff">
  <td>Total Years</td>
```

```
    <td><input type="number" placeholder="Enter Years" size="25" id="cyear" name="syear" required></td>
 </tr>
 <tr style="color:#ffffff">
  <td>Compound Rate Per Year</td>
      <td><input type="number" placeholder="Enter Compound Rate" size="25" id="cnd" name="syear" required></td>
 </tr>
 <tr>
  <td>
   <button style="color:#ff0000" onclick="myFunction2()">CALCULATE</button>
   </td>
 </tr>
 <tr>
  <td>
   <p id="CI"></p>
   </td>
 </tr>
</table>
</div>
<footer>
  <center><p style="color:#f70000">Amey Thakur B-50</p></center>
</footer>
</body>
</html>
```

- Javascript.js

```
function myFunction1(){

 var p = document.getElementById('sprincipal').value;

 var r = document.getElementById('srate').value;

 var t = document.getElementById('syear').value;

 var inr = p*r*t;

 var inrst = inr/100;

   document.getElementById('SI').innerHTML=   "<span class='color-class'>Simple Interest is "+inrst+"</span>";

}


function myFunction2(){

 var p = document.getElementById('cprincipal').value;

 var r = document.getElementById('crate').value;

 var t = document.getElementById('cyear').value;

 var n = document.getElementById('cnd').value;

 var r1 = r/100;

 var nbr = r1/n;

 var x = (1+nbr);

 var nt = n*t;

 var xnt = Math.pow(x,nt);

 var cmpndint = p*xnt;

 var z = cmpndint.toFixed(2);

  document.getElementById('CI').innerHTML=  "<span class='color-class'>Compound Interest is "+z+"</span>";

}
```

- Style.css

```
body{
  background-color: #00fbff;
}
h1.hc{
  font-family: cursive;
  font-size: 50px;
  color: #ff0000;
  position: absolute;
  left: auto;
}
div.ex{
  width: auto;
  padding: 20px;
  border: 0px;
  margin: 0px;
}
p.in{
  font-family: cursive;
  font-size: 15px;
}
.color-class{
   color:#ff0008;
}
```

B.3 Questions:

1. What is client side scripting?

Ans:

1. Client-side Scripting is utilized when the client's (user's) browser has all the code and the page is modified on the basis of the client's (user's) information. The Web Browser executes the client-side scripting that locates within the user's computer. Client-side scripts are also known as the embedded script (as they are often embedded within an HTML or XHTML document).
2. The browser gets the page sent by the server & executes the client-side scripts. Client-side scripting can't be utilized to join with the databases on the web server. Client-side scripting cannot get the file system which lies at the web server.
3. The records and settings which are local at the client's (user's) computer can be approached employing Client-side scripting language. It's generally observed the response from a client-side script is faster when compared to a server-side scripting language as the scripts are prepared on the local computers. JavaScript, VB scripts are some common examples of client-side scripting.

Examples of Popular Client-side Scripting:

1. JavaScript
2. ActionScript
3. VBScript (can be used on server side also)
4. Dart
5. TypeScript
6. Python

Advantages of Client-side Scripting:

1. Client-side Scripting offers faster response times, less overhead on the web server, and a more interactive application.
2. Ideal for altering the page elements without the need to contact the database.

Disadvantages of Client-side Scripting:

1. The drawback of Client-side scripting is that the scripting language calls for more effort and time, while the user's browser must support the scripting language also.

2. What is Dynamic HTML?

Ans:

1. Dynamic HTML is a collective term for a combination of Hypertext Markup Language (HTML) tags and options that can make Web pages more animated and interactive than previous versions of HTML. Much of dynamic HTML is specified in HTML 4.0.
2. Simple examples of dynamic HTML capabilities include having the color of a text heading change when a user passes a mouse over it and allowing a user to "drag and drop" an image to another place on a Web page.
3. Dynamic HTML can allow Web documents to look and act like desktop applications or multimedia productions.

The Concepts and Features in Dynamic HTML

1. An object-oriented view of a Web page and its elements
2. Cascading style sheets and the layering of content
3. Programming that can address all or most page elements
4. Dynamic fonts

B.4 Conclusion

Hence we studied how to simple mathematical calculations at client side using JavaScript and apply the concept of client side validation and design dynamic web pages using JavaScript