

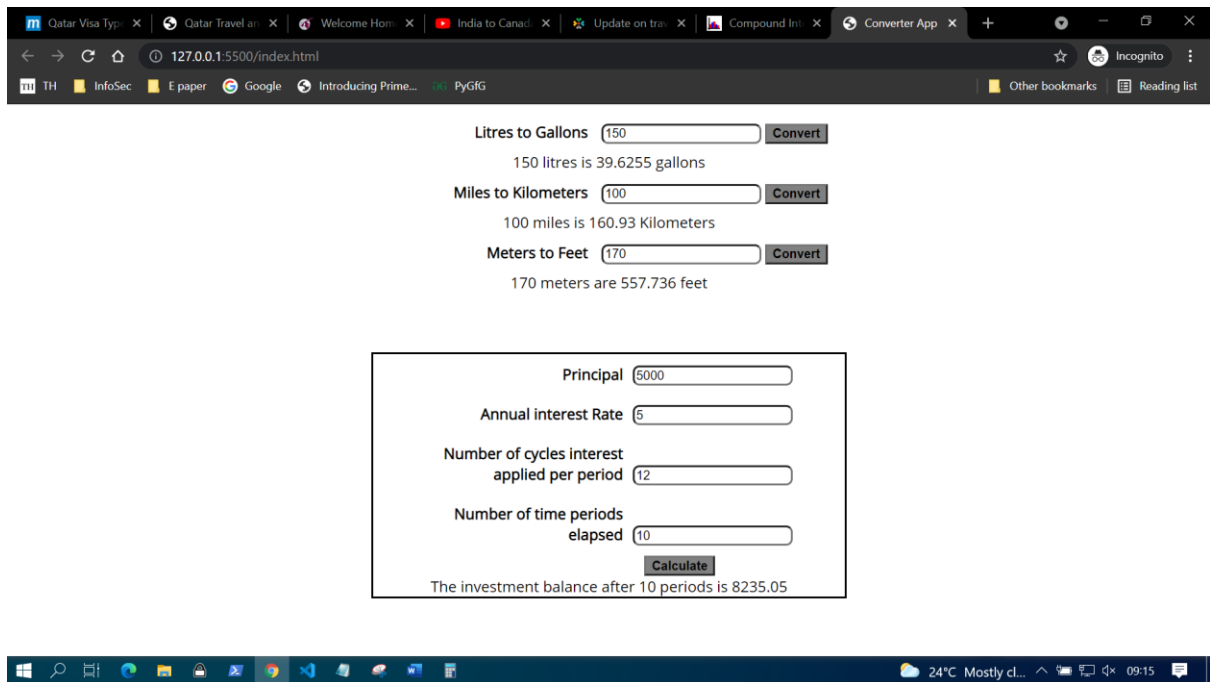
Mobile Device Fundamentals

Assignment 4

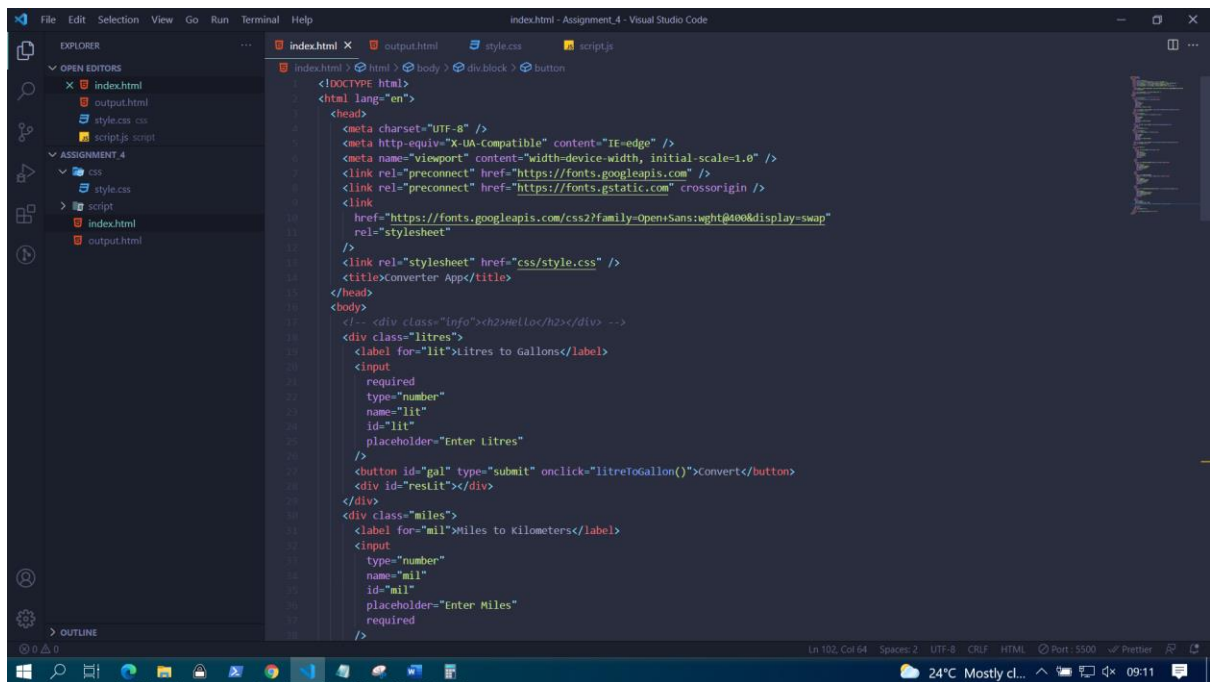
Submitted By
Umakanth Darapaneni
HID: H10008079
AID: A00232695

Name: Naveen chaldi
Hid: H10007521
Aid: A00230251

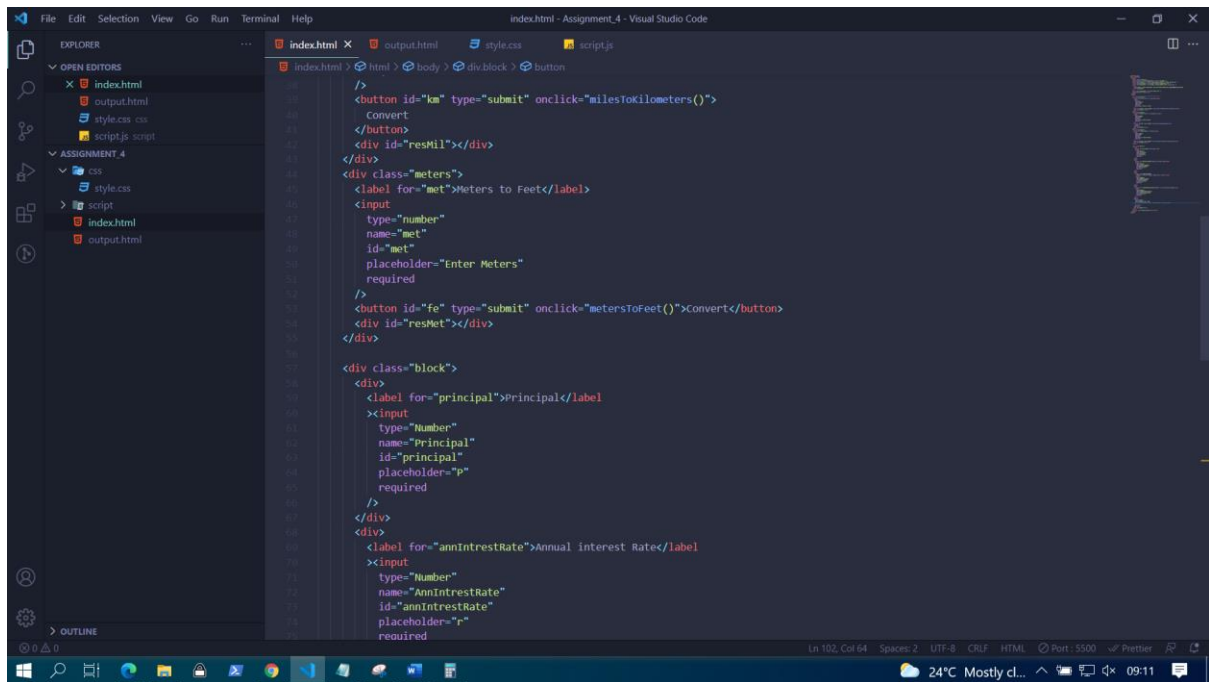
Manpreet kaur kashmir singh chauhan
AID: A00221489
HID: H10005441



HTML 1

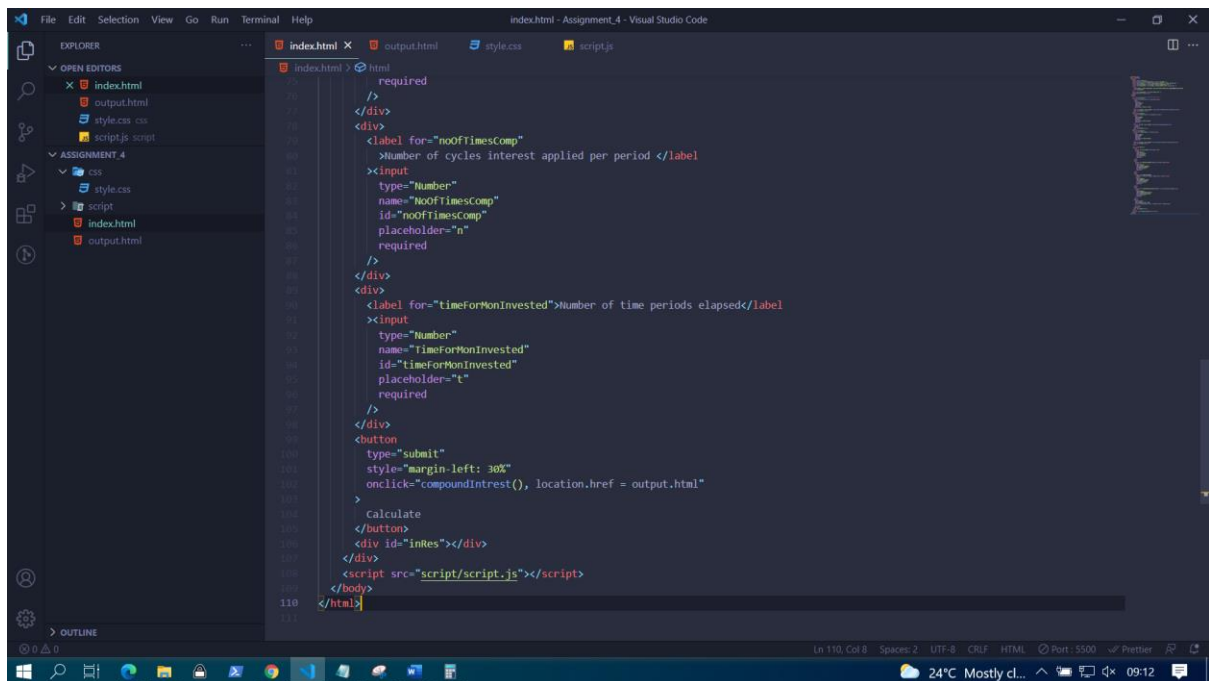


HTML 2



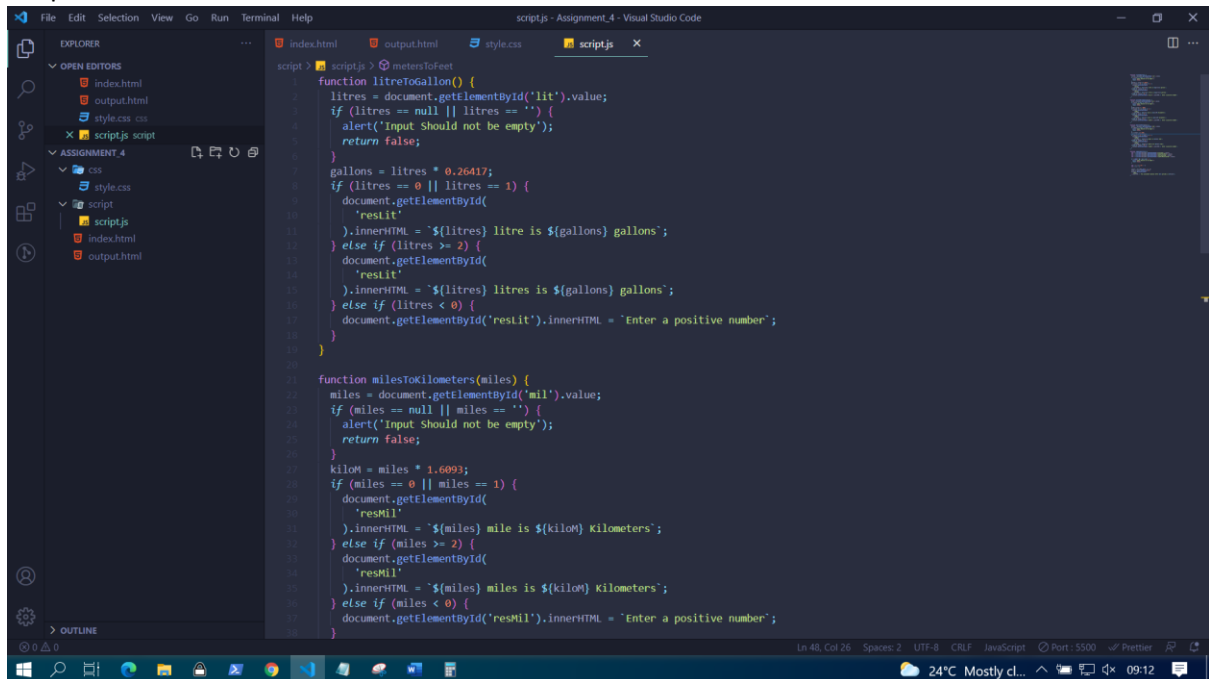
```
13  </div>
14  </div>
15  <div id="km" type="submit" onclick="milesToKilometers()">
16    Convert
17  </div>
18  <div id="resMil"></div>
19  </div>
20  <div class="meters">
21    <label for="met">Meters to Feet</label>
22    <input
23      type="number"
24      name="met"
25      id="met"
26      placeholder="Enter Meters"
27      required
28    />
29    <div id="fe" type="submit" onclick="metersToFeet()">Convert</div>
30    <div id="resMet"></div>
31  </div>
32
33  <div class="block">
34    <div>
35      <label for="principal">Principal</label>
36      <input
37        type="Number"
38        name="principal"
39        id="principal"
40        placeholder="p"
41        required
42      />
43    </div>
44    <div>
45      <label for="annIntrestRate">Annual interest Rate</label>
46      <input
47        type="Number"
48        name="AnnIntrestRate"
49        id="annIntrestRate"
50        placeholder="r"
51        required
52      />
53    </div>
54  </div>
```

HTML 3



```
101  </div>
102  </div>
103  <div>
104    <label for="noOfTimesComp">Number of cycles interest applied per period</label>
105    <input
106      type="Number"
107      name="noOfTimesComp"
108      id="noOfTimesComp"
109      placeholder="n"
110      required
111    />
112  </div>
113  <div>
114    <label for="timeForMonInvested">Number of time periods elapsed</label>
115    <input
116      type="Number"
117      name="timeForMonInvested"
118      id="timeForMonInvested"
119      placeholder="t"
120      required
121    />
122  </div>
123  <div>
124    <button
125      type="submit"
126      style="margin-left: 30px"
127      onclick="compoundIntrest(), location.href = output.html"
128    >
129      calculate
130    </button>
131  </div>
132  <div id="inRes"></div>
133  </div>
134  <script src="script/script.js"></script>
135  </body>
136  </html>
```

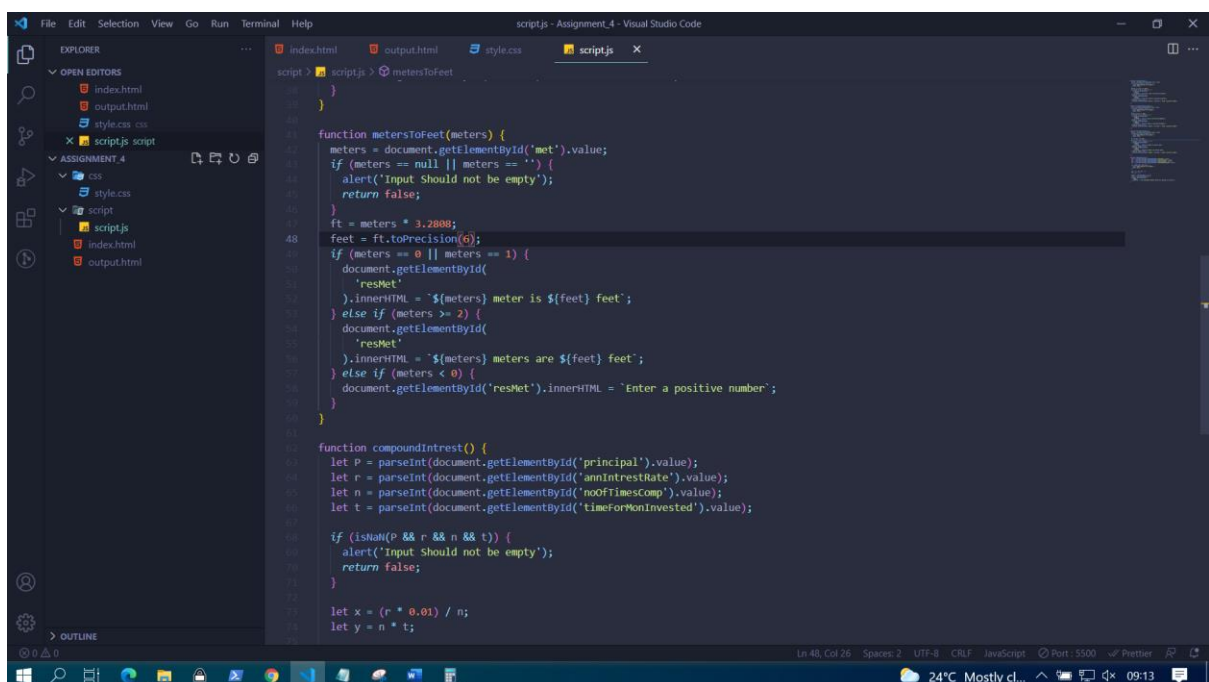
Script 1



The screenshot shows the Visual Studio Code editor with the file 'metersToGallon.js' open. The code defines two functions: 'litresToGallon' and 'milesToKilometers'. Both functions take a value from an input field, validate it, and update the output of a corresponding result field. The 'litresToGallon' function uses a conversion factor of 0.26417, and the 'milesToKilometers' function uses a conversion factor of 1.6093. Both functions include error handling for empty or non-positive inputs.

```
1 function litresToGallon() {
2   litres = document.getElementById('lit').value;
3   if (litres == null || litres == '') {
4     alert('Input should not be empty');
5     return false;
6   }
7   gallons = litres * 0.26417;
8   if (litres == 0 || litres == 1) {
9     document.getElementById(
10      'resLit'
11    ).innerHTML = `${litres} litre is ${gallons} gallons`;
12   } else if (litres >= 2) {
13     document.getElementById(
14      'resLit'
15    ).innerHTML = `${litres} litres is ${gallons} gallons`;
16   } else if (litres < 0) {
17     document.getElementById('resLit').innerHTML = 'Enter a positive number';
18   }
19 }
20
21 function milesToKilometers(miles) {
22   miles = document.getElementById('mil').value;
23   if (miles == null || miles == '') {
24     alert('Input should not be empty');
25     return false;
26   }
27   kilom = miles * 1.6093;
28   if (miles == 0 || miles == 1) {
29     document.getElementById(
30      'resMil'
31    ).innerHTML = `${miles} mile is ${kilom} Kilometers`;
32   } else if (miles >= 2) {
33     document.getElementById(
34      'resMil'
35    ).innerHTML = `${miles} miles is ${kilom} Kilometers`;
36   } else if (miles < 0) {
37     document.getElementById('resMil').innerHTML = 'Enter a positive number';
38   }
39 }
```

Script 2



The screenshot shows the Visual Studio Code editor with the file 'metersToFeet.js' open. The code defines two functions: 'metersToFeet' and 'compoundIntrest'. The 'metersToFeet' function takes a value from an input field, validates it, and updates the output of a result field. It uses a conversion factor of 3.28084 and includes error handling for empty or non-positive inputs. The 'compoundIntrest' function takes four values from input fields (principal, annual interest rate, number of times compounded, and time for money invested), validates them, and calculates the compound interest using the formula $x = (r * 0.01) / n$ and $y = n * t$.

```
1 }
2 }
3
4 function metersToFeet(meters) {
5   meters = document.getElementById('met').value;
6   if (meters == null || meters == '') {
7     alert('Input should not be empty');
8     return false;
9   }
10   ft = meters * 3.28084;
11   feet = ft.toFixed(0);
12   if (meters == 0 || meters == 1) {
13     document.getElementById(
14      'resMet'
15    ).innerHTML = `${meters} meter is ${feet} feet`;
16   } else if (meters >= 2) {
17     document.getElementById(
18      'resMet'
19    ).innerHTML = `${meters} meters are ${feet} feet`;
20   } else if (meters < 0) {
21     document.getElementById('resMet').innerHTML = 'Enter a positive number';
22   }
23 }
24
25 function compoundIntrest() {
26   let p = parseInt(document.getElementById('principal').value);
27   let r = parseInt(document.getElementById('annIntrestRate').value);
28   let n = parseInt(document.getElementById('noOfTimesComp').value);
29   let t = parseInt(document.getElementById('timeForMoneyInvested').value);
30
31   if (isNaN(p) || isNaN(r) || isNaN(n) || isNaN(t)) {
32     alert('Input should not be empty');
33     return false;
34   }
35
36   let x = (r * 0.01) / n;
37   let y = n * t;
38 }
```

Script 3

```
1  // metersToFeet
2
3  // validate input
4  if (isNaN(P) || r || n || t) {
5      alert('Input should not be empty');
6      return false;
7  }
8
9  // calculate
10 let x = (r * 0.01) / n;
11 let y = n * t;
12
13 // calculate result
14 result = P * Math.pow(1 + x, y);
15 final = result.toFixed(2);
16 document.getElementById(
17     'inRes'
18 ).innerHTML = `The investment balance after ${t} periods is ${final}`;
19 }
```

```
1 body {
2     text-align: center;
3 }
4
5 div {
6     font-family: 'Open Sans', sans-serif;
7 }
8
9 label {
10    font-weight: bold;
11    display: inline-block;
12    width: 200px;
13    margin: 10px;
14    text-align: right;
15 }
16
17 .block {
18    margin-left: 30%;
19    margin-right: 30%;
20    margin-top: 5%;
21    border-style: solid;
22 }
23
24 input {
25    border-radius: 7px;
26 }
27
28 button {
29    color: black;
30    font-weight: bolder;
31    background-color: grey;
32 }
33 }
```