PROG8020

Week 3a

Advanced Decisions and Loops

Some Simple Schoolroom Statistics

It all adds up: Summing

Program to sum test scores using a sentinel value to end the input:

```
1. <html> <head>
2. <title>Summing</title>
3. <script>
4.          function getSum()      {
5.                    let score = 0; let sum = 0;
6.                    while (score != -999)      {
7.                              sum += score;
8.                              score = parseInt(prompt("Enter a score or
                                        enter -999 when you're finished:"," "));
9.                    }
10.                   document.write(`the sum of these scores is: ${sum}.`);
11.         }
12. </script> </<head>
13. <body>
14.         <h3>Click to enter students' scores</h3>
15.         <p><input type="button" id="scores" value="Enter the scores"
                                        onclick="getSum();" /></p>
17. </body> </html>
```

Computing Averages

To find the average, we must sum and count the numbers entered. The example finds the average of values entered using a counter to keep track of how many values are entered.

```
1. function getAverage()  {
2.       let score = 0; let sum = 0;
3.       let count = 0; let average = 0;
4.       while (score != -999)   {
5.                sum += score;
6.                count++;
7.                score = parseInt(prompt("Enter a score or
                           enter -999 when you're finished:"," "));
8.       }
9.       average = sum/(count - 1);
10.      document.write("<p>The sum of these scores is: " +
                           sum + ".</p>");
11.    document.write("<p>The average of these scores is: "
                           + average + ".</p>");
12. }
```

All together: Sum, Average, Lowest & Highest

```
1. function getStats()
2. {
3.     let sum = 0; let count = 1;
5.     let score = parseInt(prompt("Enter a score or enter -999 when finished:"," "));
6.     let low = score; let high = score;
7.     while (score != -999)  {
8.         sum += score;
9.         count++;
10.        score = parseInt(prompt("Enter a score or -999 when finished:"," "));
11.        if (score > high)
12.          high = score;
13.        if ((score < low) && (score != -999))
14.          low = score;
15.    }
16.    let average = parseInt(sum/(count - 1));
17.    document.write("<p>Number of scores entered: " + (count - 1) + ".</p>");
18.    document.write("<p>Sum of these scores: " + sum + ".</p>");
19.    document.write("<p>Average of these scores: " + average + ".</p>");
20.    document.write("<p>Lowest score is: " + low + ".</p>");
21.    document.write("<p>Hghest score is: " + high + ".</p>");
22.}
```

Odd or Even?

It is often important to identify if a number is even or odd.

If a number is odd, then it will have a remainder when divided by 2. That remainder will always be 1. If it is even, the remainder, when dividing by 2 will be 0. We can use that to check if a number is odd or even.

Given a numeric variable, **myNum**:

- If **myNum** `%` `2` `==` `1` → **myNum** is odd
- If **myNum** `%` `2` `==` `0` → **myNum** is even

Some Math methods

- The `Math.round()` method rounds off floating point numbers mathematically to an integer value
  - `Math.round(89.001)` results in `89`
  - `Math.round(89.678)` results in `90`
- The `Math.floor()` method truncates the decimal part of any floating point number
  - `Math.floor(89.001)` results in `89`
  - `Math.floor(89.678)` results in `89`
- The `Math.ceil()` method rounds any floating point number up to the next integer
  - `Math.ceil(89.001)` results in `90`
  - `Math.ceil(89.678)` results in `90`

To continue or Not to continue?

The `break` statement

The `break` statement is essential to a `switch` structure. It can also be used in a loop if necessary but often creating a compound condition can do the same thing. The general pseudocode for a possible use of the `break` statement is given. In this situation a user can continue shopping until a maximum amount is spent. Then the user should be informed that further purchases will increase shipping charges.

- Declare and initialize variables
- Start a while loop
  - Prompt the user for the item desired and how many will be bought
  - Use a switch to identify the cost of that item, calculate cost of the number of that item ordered, keep a running sum of the total cost.
  - Check if total cost puts the user over the max allowed
    - If this is true, break out of the loop and inform the customer that another item will increase the shipping cost
  - Continue with the loop until the customer is done shopping

The `continue` statement

- The **`continue` statement** allows you to skip an iteration in a loop once (or more, depending on the conditions) but return and complete more iterations.

Example: `continue`

The following program uses the `continue` statement to count by threes:

```
1. function getThrees()
2. {
3.      for (let i = 0; i <= 100; i++)
4.      {
5.              if ((i/3) != parseInt(i/3))
6.                      {  continue;  }
7.              document.write(i + "   ");
8.      }
9. }
```

Output will be:

0 3 6 9 12 15 18 21 24 27 30 33 36
39 42 45 48 51 54 57 60 63 66 69 72
75 78 81 84 87 90 93 96 99

Nested for Loops

Desk checking

- Most important when programs become complex
- Use pencil and paper to go through the program, line by line
- Write down the value of each variable at each line and any output

Desk checking example

```
1.function getLoops()
2.{
3.    var x = 0; var y = 0; var z = 0;
4.    for (x = 1; x < 4; x++)
5.    {
6.          document.write("<h3>Pass " + x + "</h3>");
7.          for (y = 1; y < 10; y+=3)
8.          {
9.             z = x + y;
10.            document.write("<p>"+ x +" + "+ y +" = "+ z +"</p>");
11.         }
12.   }
13. }
```

Pass	value of x	value of y	value of z	output
Outer pass 1	1	0	0	
Inner pass 1	1	1	2	1 + 1 = 2
Inner pass 2	1	4	5	1 + 4 = 5
Inner pass 3	1	7	8	1 + 7 = 8
loop ends,test fails	1	10	8	
Outer pass 2	2	10	8	
Inner pass 1	2	1	3	2 + 1 = 3
Inner pass 2	2	4	6	2 + 4 = 6
Inner pass 3	2	7	9	2 + 7 = 9
loop ends, test fails	2	10	9	
Outer pass 3	3	10	9	
Inner pass 1	3	1	4	3 + 1 = 4
Inner pass 2	3	4	7	3 + 4 = 7
Inner pass 3	3	7	10	3 + 7 = 10
loop ends, test fails	3	10	10	

Which way to nest loops?

- No clear answer to this question.
- You can nest a pre-test loop inside a post-test loop
- Or nest a `for` loop inside a `while` loop or nest several `while` loops inside a `for` loop or any other combination.
- The programming problem that you are faced with may determine how you write the code.
- If there is no clear reason to select one option over another, the choice is yours.

Drawing Shapes and Patterns

Drawing Shapes: Square

```
1.   function getSquare()
2.   {
3.         var side = 0; row = 1; col = 1; symbol = "* ";
4.         symbol = prompt("Pick a keyboard symbol for your square,↵
                    such as a * or # "," ");
5.         side = prompt("How big is the side of your square? The ↵
                         value must be a positive number: "," ");
6.         side = parseInt(side);
7.         while (side < 1)
8.         {
9.               side = prompt("How big is the side of the square? The↵
                         value must be a positive number: "," ");
10.              side = parseInt(side);
11.        }
12.        for (row = 1; row <= side; row++)
13.        {
14.              for (col = 1; col <= side; col++)
15.                    document.write(symbol + " ");
16.              document.write("<br />");
17.        }
18.  }
```

Drawing Shapes: Rectangle

```
1.  function getRectangle()
2.  {
3.        var width = 0; var length = 0; row = 1;
4.        col = 1; symbol = "* ";
5.        symbol = prompt("Pick a keyboard symbol for your rectangle,↵
                    such as a * or # ", " ");
6.        width = prompt("What is the width? The value must be a ↵
                    positive number: "," ");
7.        width = parseInt(width);
8.        while (width < 1)
9.        {
10.             width = prompt("What is the width? The value must be ↵
                    a positive number: "," ");
11.             width = parseInt(width);
12.       }
13.       length = prompt("What is the length? The value must be ↵
                    a positive number: "," ");
14.       length = parseInt(length);
15.       while (length < 1)
16.       {
17.             length = prompt("What is the length? The value must↵
                    be a positive number: "," ");
18.             length = parseInt(length);
19.       }
20.       for (row = 1; row <= width; row++)
21.       {
22.             for (col = 1; col <= length; col++)
23.                   document.write(symbol + " ");
24.             document.write("<br />");
25.       }
26. }
```

Drawing Shapes: Right Triangle

```
1.    function getTriangle()
2.    {
3.          var row = 1; var base = 1; var symbol = "* "; var col = 0;
4.          symbol = prompt("Pick a keyboard symbol for your triangle,↵
                          such as a * or # "," ");
5.          base = prompt("How big is the base of your triangle? The ↵
                          value must be a positive number: "," ");
6.          base = parseInt(base);
7.          while (base < 1)
8.          {
9.                base = prompt("How big is the base of your triangle?↵
                          The value must be a positive number: ↵
                          "," ");
10.               base = parseInt(base);
11.         }
12.         for (row = 1; row <= base; row++)
13.         {
14.               for(col = 1; col <= row; col++)
15.                     document.write(symbol + " ");
16.               document.write("<br />");
17.         }
18.   }
```

Mouse Events: Creating a Rollover

In this example the JavaScript is within the HTML page, not in the `<head>` section.

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <title>Page Title</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
</head>
<body>
  <main>
    <h1>The Rollover</h1>
    <h3>To change the image, roll your mouse over it</h3>
    <div>
      <a href="#" onmouseover="document.photo.src = 'rainbow_falls.jpg'" onmouseout="document.photo.src =
        'falls.jpg'">
        <img src="falls.jpg" name="photo">
      </a>
    </div>
  </main>

</body>
</html>
```

Rollover with functions

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <title>Page Title</title>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <script>
        function change_falls(){
          document.getElementById('photo').src = 'falls.jpg';
        }
        function change_rainbow(){
          document.getElementById('photo').src = 'rainbow_falls.jpg';
        }

    </script>
</head>
<body>
  <main>
    <h1>The Rollover</h1>
    <h3>To change the image, roll your mouse over it</h3>
    <div>
      <a href="#" onmouseover="change_rainbow();" onmouseout="change_falls();">
        <img src="falls.jpg" id="photo">
      </a>
    </div>
  </main>

</body>
</html>
```

More Mouse Events

Attribute	Value	Description: what it does
onclick	JavaScript	what happens when the mouse is clicked
ondblclick	JavaScript	what happens when the mouse is double-clicked
onmousedown	JavaScript	what happens when a mouse button is pressed
onmousemove	JavaScript	what happens when the mouse pointer moves
onmouseout	JavaScript	what happens when the mouse pointer moves off an element
onmouseover	JavaScript	what happens when the mouse pointer moves over an element
onmouseup	JavaScript	what happens when the mouse button is released

Thank you!