If … else statements – good morning example

Exercise

Write a script to check someone's age and determine if the person can consume alcohol

to the minimum legal drinking age is 21

If statement- example

```
var score = 75;
var msg;
if (score >= 50) {
msg = 'Congratulations!';
msg += ' Proceed to the next round.';
}

var el = document.getElementById('answer')
el.textContent = msg;
```

If statement- example

```
<!DOCTYPE html>
<html>
 <head>
 <title>- If Statement</title>
<link rel="stylesheet" href="css/c04.css" />
</head> <body>
 <section id="page1">
 <h1>Bullseye</h1>
<img src="images/teacher.png" id="teacher" alt="teacher" /> <section id="answer">
</section>
</section>
<script src="js/if-statement.js"></script>
</body> </html>
```

If statement- example

BULLSEYE!

TARGET PRACTICE FOR YOUR MIND

Congratulations!
Proceed to the next
round.

If …else statement- example

```
var pass = 50;
 var score = 75;
var msg;
if (score > pass) {
msg = 'Congratulations, you passed!';
 } else {
msg = 'Have another go!';
 }
var el = document.getElementById('answer');
el.textContent = msg;
```

If …else statement with function- example

```
var score = 75;
var msg = '';
function congratulate() {
        msg += 'Congratulations! ';
 }
 if (score >= 50) {
        congratulate();
        msg += 'Proceed to the next round.';
 }
var el = document.getElementById('answer');
 el.innerHTML = msg;
```

If … else statements – good morning example

Exercise

Write a script to print “DES 350 class day” if current day is M or W

Switch statement

Switch statement starts with a variable switch value.
Each case indicates a possible value for the switch variable and the code that should run if the variable matches that value.

If a match is found, that code is executed. The break statement stops switch statement.

Better performance than multiple if statements.

Switch statement

```
switch (level) {                        //switch value variable
        case 'One':                  //if switch value is "One' this code executed
        title='Level 1';
        break;

        case 'Two':                  //if switch value is 'Two' this code executed
        title='Level 2';
        break;

        default:                     //if none of the above this code executed
        title='Test';
        break;
}
```

Switch statement- example

```
var msg;
var level = 2;
switch (level) {
case 1:
msg = 'Good luck on the first test';
break;
case 2:
msg = 'Second of three - keep going!';
break;
case 3:
msg = 'Final round, almost there!';
break;
default:
msg = 'Good luck!'; break;
 }
```

Switch statement

```
var el = document.getElementById('answer');
el.textContent = msg;
```

Switch statement

```
<!DOCTYPE html>
<html>
<head>
<title>Switch Statement</title>
 <link rel="stylesheet" href="css/c04.css" />
 </head>
<body>
 <section id="page1">
 <h1>Bullseye</h1>
<img src="images/teacher.png" id="teacher" alt="teacher" />
 <section id="answer"></section>
</section>
<script src="js/switch-statement.js"></script>
</body>
 </html>
```

Weak typing

JavaScript allows you not to specify what data type each variable will
(in declaration). JavaScript uses weak typing.

Data type for a value can change.

Data type	Purpose
string	Text
number	Number
boolean	true or false
null	Empty value
undefined	variable has been declared but not yet assigned a value

Type Coercion

Converts data types behind the scenes to compete the operation.

('1' >0) returns true

String is converted to a number

('ten'/2) returns NaN (Not a Number)

Type Coercion

Because of type coercion, the strict equality operators === and !==
Result in fewer unexpected values than == and != do.
false, 0 and ‘ ‘

(0 == ‘ ‘) true
(0 === ‘ ‘) false

(false == ‘ ‘) true
(false === ‘ ‘) false

For Loop

Loop checks a condition. If the condition is true, the statements in curly braces will be executed. The cycle repeats until the condition returns false.

```
for (var i = 0; i < 10; i++)  {
        document.write(i);
}
```

(initialization; condition; increment)

For Loop

Often used to loop through the items in an array.

<html>
<head>
<title>loop</title>
<script>

For Loop

```
function myFunction() {
var x="";
for (i=0;i<50;i++) {
  x=x + "The number is " + i + "<br>";
  }
document.getElementById("demo").innerHTML=x;
}
</script>
</head>
<body>
```

For Loop

```
<p>Click the button to loop through a block of as long as <em>i</em>
less than 50.</p>
<button onclick="myFunction()">Try it</button>
<p id="demo"></p>


</body>
</html>
```

while Loop

While loop will run as long as the condition is true.

```
while ( i < 10 )  {
        statements;
        i ++;
}
```

do while Loop

Do while loop will execute statements first, before it checks the condition .

```
do {
        statements;
        i ++;
} while ( i < 10 );
```

do while Loop - example

```
var i = 1; // Set counter to 1
var msg = ''; // Message // Store 5 times table in a variable
do {
msg += i + ' x 5 = ' + (i * 5) + '<br />';
i++;
}  while (i < 1); // Note how this is already 1 and it still runs
document.getElementById('answer').innerHTML = msg;
```

do while Loop - example

```
<!DOCTYPE html>
<html>
  <head>
    <title>Do While Loop</title>
    <link rel="stylesheet" href="css/c04.css" />
  </head>
  <body>
          <section id="page1">
                  <h1>Bullseye</h1>
                  <img src="images/teacher.png" id="teacher" alt="teacher" />
                  <section id="answer"></section>
            </section>
    <script src="js/do-while-loop.js"></script>
  </body> </html>
```

do while Loop - example

Review chapter 4 example