# Exercise 1 – Types & Operators

## Objectives

In this exercise, you will investigate JavaScript flow of control statements, including branching and looping constructs.

## Exercise Instructions

### Part 1 – **if else** statements

1. Start by creating an **age** variable and initialise it to be **15**.

***var age = 15***

1. Now we are going to add a simple branching if statement. The **if** statement is one of the building blocks of programming so it is essential we understand it. In this statement we will check the **age** variable against a logical condition and let the code decide what to do next.
2. Enter the following code to create an **if else** statement below the variable declaration

***if (age <=17) {***

***console.log(“Underage!”)***

***} else {***

***console.log(“Over 18”)***

***}***

1. Save your file and now open it in **Chrome**, bring up the developer tools with **F12** and you should see the output **Underage!** In the console. Our if statement has returned true when comparing the **age** variable to **17**, thus executing the block of code associated with it. Change the value of **age** to **42** and save/run your code again (Refresh the page!)

### Part 2 – **else if** statements

1. We now have a simple **if** statement in place. It is time to take our code a step further and add an **else if** into our code. The **else if** is a further logical check delivered a Boolean value. Additional checks are examined in order, as soon as any **if** or **else if** results to **true** then the program leaves the **if** block. So order is important
2. Modify your previous **if else** block to look like this

***If (age <= 17) {***

***console.log(“Underage!”);***

***} else if( ) {***

***console.log(“Insurable!”);***

***} else {***

***console.log(“out of range”);***

***}***

1. The **else if** block is going to let us put another **Boolean expression** inside of it, similar to the one existing insider of the **if** statement. We want the **else if** to check if **age** is between 18 and 65. This will be two Boolean expressions joined with an **and** operator. Refer back to your notes or ask your instructor for help to create a statement to achieve this.
2. Once you have done this, test your code by setting **age** to the following values and running your code each time.
   1. 10
   2. 50
   3. 80

### Part 3 – **for** loops

A lot of the time, code needs to be used repetitively. Very often you need to achieve the same activity more than once, copy and pasting lines of code can work but that would be a messy solution. Loops are a construct that is necessary in general programming and something that we will be using now.

1. The **for** loop has 3 parts to it. The counter, the condition and the iterator. You are going to code a simple **for** loop where the following properties need to be set:

|  |  |
| --- | --- |
| **parameter** | **value** |
| counter | Variable name **i** set to **1** |
| condition | While **i** is less than **10** |
| Iterator | Each loop must add **1** to the value of **i** |

1. Enter the following code, amended properly to achieve this:

***for(counter; condition; iterator){***

***console.log(i);***

***}***

1. How many times do you expect the loop to execute?
2. Save your code and run it in **Chrome** to check your code against your assumption

### Part 4 – **while** loops

1. **For** loops are best used when there is a fixed number of iterations that are required. The **while** loop is useful when the end is not known. It can be used for iterator-based logic but that would require a variable declaration external to the loop.
2. Enter the following code:

***var counter = 2***

***var loopCount = 0***

***while(counter2 < 100){***

***counter2 = //square of counter2***

***//code to increment the loopCount by 1***

***console.log(“total: “ + counter2)***

***console.log(“loopCount: “ + loopCount)***

***}***

1. Note the two commented lines above. You need to provide the square of counter2 (A square is a value multiplied by the same value, 2\*2, 3\*3 etc.) and add code to increment the value of loopCount by 1. Call your instructor if you need help.
2. Once you have solved this, save your code and run it in **Chrome**

### **Further Activities**

Only attempt these further activities if there is sufficient time remaining. Your instructor can help you with these if you need it. But do try them on your own, look at your notes and so on.

1. Alter your **for** loop to count from **10** down to **0**
2. Alter your **for** loop to count up in **3**s