**Control Flow**

**If statement:**

The if statement is used to execute a block of code if a specified condition is true. If the condition evaluates to false, the code inside the if block is skipped. Optionally, you can use else if and else clauses to specify additional conditions to check if the initial condition is false.

**Switch statement:**

The switch statement is used to perform different actions based on different conditions. It evaluates an expression and executes the case that matches the expression's value. You can use multiple case clauses to specify different possible values for the expression. The break statement is used to exit the switch block once a matching case is found.

**While loop:**

The while loop executes a block of code as long as a specified condition evaluates to true. It continuously loops through the code block until the condition becomes false. It's important to ensure that the condition will eventually become false to prevent an infinite loop.

**Do-While loop:**

The do-while loop is similar to the while loop, except that it executes the code block at least once before checking the condition. After the first iteration, it checks the condition, and if it's true, it continues to execute the loop.

**For loop:**

The for loop is used to repeatedly execute a block of code a specified number of times. It consists of three parts: initialization, condition, and iteration expression. The initialization is executed once at the beginning of the loop. The condition is evaluated before each iteration, and if it's true, the loop continues. The iteration expression is executed at the end of each iteration.

**For-In loop:**

The for-in loop is used to iterate over the properties of an object. It loops through each enumerable property of an object and executes the specified block of code for each property. It's important to note that the order of iteration is not guaranteed, and it may vary between different JavaScript engines.

**For-Of loop:**

The for-of loop is used to iterate over iterable objects like arrays, strings, and other iterable objects. It provides a more concise syntax compared to traditional for loops and is particularly useful for iterating over the elements of an array or the characters of a string.

**Break statement:**

The break statement is used to terminate the execution of a loop prematurely. When the break statement is encountered inside a loop, the loop is exited immediately, and the program continues with the next statement following the loop.

**Continue statement:**

The continue statement is used to skip the current iteration of a loop and continue with the next iteration. When the continue statement is encountered inside a loop, the remaining code inside the loop for the current iteration is skipped, and the loop proceeds with the next iteration.