JavaScript

JavaScript Loop Control Structures

JavaScript provides several loop control structures to iterate over data or execute a block of code multiple times.

For loop

```
for (expression 1; expression 2; expression 3) {
  // code block to be executed
}
```

```
for (let i = 0; i < 5; i++) {
  text += "The number is " + i + "<br>";
}
```

The While Loop

The while loop loops through a block of code as long as a specified condition is true.

```
while (condition) {
  // code block to be executed
}
```

for eg.

```
 while (i < 10) {
  text += "The number is " + i;
  i++;
}
```

The Do-While Loop

The do while loop is a variant of the while loop. This loop will execute the code block once, before checking if the condition is true, then it will repeat the loop as long as the condition is true.

```
do {
 // code block to be executed
}
while (condition);
```

```
do {
 text += "The number is " + i;
 i++;
}
while (i < 10);
```

Switch Case Statement

The switch statement evaluates an expression, compares its results with case values, and executes the statement associated with the matching case value.

```
switch (expression) {
   case value1:
      statement1;
      break;
   case value2:
      statement2;
      break;
   case value3:
      statement3;
      break;
   default:
      statement;
}
```

```
let day = 3;
let dayName;

switch (day) {
  case 1:
    dayName = 'Sunday';
    break;
  case 2:
    dayName = 'Monday';
    break;
  case 3:
    dayName = 'Tuesday';
    break;
  case 4:
    dayName = 'Wednesday';
    break;
  case 5:
    dayName = 'Thursday';
    break;
  case 6:
    dayName = 'Friday';
    break;
  case 7:
    dayName = 'Saturday';
    break;
  default:
    dayName = 'Invalid day';
}

console.log(dayName); // Tuesday
```

Break And Continue Statements

The break statement "jumps out" of a loop.

The continue statement "jumps over" one iteration in the loop.

The Break Statement

You have already seen the break statement used in an earlier chapter of this tutorial. It was used to "jump out" of a switch() statement.

The Continue Statement

The continue statement breaks one iteration (in the loop), if a specified condition occurs, and continues with the next iteration in the loop.

DataTypes in Js

There are some primitive datatypes in JavaScript

1. null
2. undefined
3. boolean
4. number
5. string

eg.

```
let counter = 120;

console.log(typeof(counter)); // "number"

counter = false;

console.log(typeof(counter)); // "boolean"

counter = "Hi";

console.log(typeof(counter)); // "string"
```

```
let counter;
console.log(counter);       // undefined
console.log(typeof counter); // undefined

let obj = null;
console.log(typeof obj); // object
```

Objects in js:

In JavaScript, an object is an unordered collection of key-value pairs. Each key-value pair is called a property.

The key of a property can be a string. The value of a property can be any value, e.g., a string, a number, an array, and even a function.

```
let person = {
   firstName: 'John',
   lastName: 'Doe'
};
```

The dot notation (.)

objectName.propertyName

Array-like notation ([])

objectName['propertyName']

To print overall object use

onsole.log(person);

Function In JS

When developing an application, you often need to perform the same action in many places. For example, you may want to show a message whenever an error occurs.

To avoid repeating the same code all over places, you can use a function to wrap that code and reuse it.

Declare a function

To declare a function, you use the function keyword, followed by the function name, a list of parameters, and the function body as follows:

```
function functionName(parameters) {
   // function body
}
```

```
function say() {
        console.log(“Generated Alert Box Function”);
}
```

```
function add(a, b) {
}
```

```
function say(message) {
   console.log(message);
}
```

Returning a value

Every function in JavaScript implicitly returns undefined unless you explicitly specify a return value.

```
function say(message) {
   console.log(message);
}

let result = say('Hello');
console.log('Result:', result);

function add(a, b) {
   return a + b;
}
let sum = add(10, 20);
console.log('Sum:', sum);
```

Questions on if else statements:

1)Check between three numbers which is largest of them

2)Write a program to check the grade based on below range

90-100: A

80-89: B

70-79: C

60-69: D

Below 60: F

3)check day of week

4)check if the letter is vowel or consonant

5)check if a number is positive negative or zero

Using Switch case

1)number of days in the month

2)check day of week

3)check if the letter is vowel or consonant