

Conditionals

Overview

Conditional statements are used to run different sets of statements according to the conditions. This means that based upon various conditions, we can perform different actions. These are the same as used in other languages.

In JavaScript, we have the following conditional statements:

- Use **if** to specify a block of code to be executed if a specified condition is true.
- Use **else** to specify a block of code to be executed if the same condition is false.
- Use **else if** to specify a new condition to test, if the first condition is false
- Use the **switch** to specify many alternative blocks of code to be executed.

if Statements

The **if** statement specifies a block of statements (JavaScript code) that gets executed when the condition specified in the **if** statement evaluates to true.

```
Syntax :    if (condition) {  
                //block of code to be executed if the condition is true  
            }
```

First, the condition is evaluated, and if it is true, then the statements inside it are executed. Else if the condition is false, statements inside 'if' are not executed.

Example: Print Coding Ninjas if marks are greater than 90

```
if(marks>90){  
    console.log("Coding Ninjas");  
}
```

if-else Statements

The **if** statement specifies a block of statements (JavaScript code) that gets executed when the **condition** specified in the if statement evaluates to true.

Else, if the condition is false, then the statements inside the **else** block are executed.

```
Syntax :  if (condition) {  
            //code to be executed if condition1 is true  
        } else {  
            //code to be executed if the condition is false  
        }
```

Example: Print Good if marks are greater than 90, else Bad

```
if(marks>90){  
    console.log("Good");  
}else{  
    console.log("Bad");  
}
```

else-if Statements

The above two works for only two options, i.e. whether the condition is true or false.

Using **else-if**, we can define more than one condition and form a chain. The first condition that returns true gets executed, else the last 'else' block is executed.

```
Syntax :  if (condition1) {  
            // code to be executed if condition1 is true  
        } else if (condition2) {  
            // code to be executed if the condition1 is false and condition2 is true  
        } else {  
            // code to be executed if the condition1 is false and condition2 is false  
        }
```

Now the first condition is checked. If it evaluates to true, it gets executed and other **else if** blocks are not checked. If the first condition is false, then the second condition is checked.

Like this, each **if** condition is checked one by one until one is true or the **else** block is reached. If any one of them is true, the block associated with it gets executed. Else the **else** block is executed.

Example : If marks <50 then print bad , if marks <80 print average , if marks >= 80 print good

```
if (marks < 50) {  
    console.log("Bad");  
} else if (marks < 80) {  
    console.log("Average");  
} else {  
    console.log("Good");  
}
```

Switch Statement

The **switch** statements are similar to **else-if** statements but **work on a single expression**. This expression evaluates different values which decide which block of code needs to be executed.

Syntax :

```
switch(expression) {  
    case x:  
        // code block  
        break;  
    case y:  
        // code block  
        break;  
    default:  
        // code block  
}
```

Work Flow of Switch Statements :

- The switch expression is evaluated once.
- The value of the expression is compared with the values of each case.
- If there is a match, the associated block of code is executed.
- If there is no match, the default code block is executed.

```
Example : const course= 'Web-Dev';

switch (course) {
  case 'DSA':
    console.log('Your course is DSA');
    break;
  case 'Web-Dev':
    console.log('Your course is Web-Dev');
    break;
  default:
    console.log(` Sorry, course not found `);
}
```

Output: Your course is Web-Dev

break Keyword

We have provided a **break**; statement after each case is completed to stop the execution of the **switch** block. If the **break** is not present, then the cases after the one that matches also get executed until a **break**; is found or the **switch** block does not end.

Note: If you omit the break statement, the following case will be executed even if the evaluation does not match the case.

default Keyword

The default keyword specifies the code to run if there is no case match

If the course would've been **'Android'** in the above example, then the code would've printed **"Sorry, course not found."**

default case does not have to be the last case in a switch statements

```
Example: const course='Android';

    switch (course) {
      default:
        console.log(` Sorry, course not found `);
        break; // Mandatory
      case 'DSA':
        console.log('Your course is DSA');
        break;
      case 'Web-Dev':
        console.log('Your course is Web-Dev');
        break;
    }
```

Output: Sorry, course not found

Note: If the default is not the last case in the switch block, remember to end the default case with a break

Points to Remember

- If multiple cases have the same case value, the first case is selected.
- If no matching cases are found, the program continues to the default label.
- If no default label is found, the program continues to the statement(s) after the switch.

Strict Comparison

- Switch cases use **strict comparison (===)**, i.e. the same values must match the switch case value.

Example : `let temp = "1" ; //type String`
switch (temp) {
 case 0:
 `console.log("Number is 0");`
 `break;`
 case 1:
 `console.log("Number is 1");`
 `break;`
 default:
 `console.log("Number not found");`
 }

Output: Number not found

Common Code Blocks for Different Switch Case Values

Sometimes you will want different switch cases to use the same code

In the below Example for both HTML and CSS course assigned is Web-Dev

Example: `const language = 'HTML' ;`
switch (language){
 case 'HTML' :
 case 'CSS' :
 `console.log('Your course is Web-Dev');`
 `break;`
 case 'KOTLIN' :
 `console.log('Your course is Android');`
 `break;`
 default:
 `console.log(` Sorry, course not found `);`
 }

Output: Your course is Web-Dev