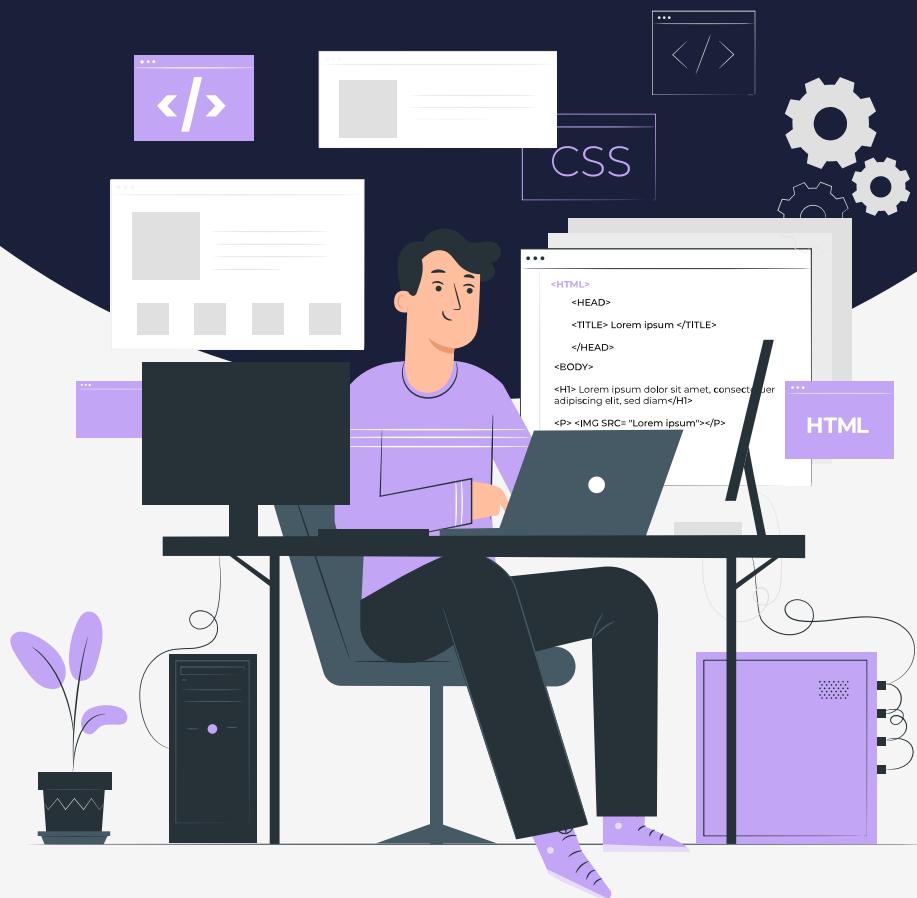


Lesson:

Condition

(If, If-else and if else ladder)



Topics Covered

1. Introduction to Conditional Statements
2. Conditional Statements in JavaScript
 - a. if statement
 - b. if-else statement
 - c. If else ladder

Introduction to Conditional Statements

Programming Languages are tools that allow us to write code that instructs the computer to do something. In every programming language, the code needs to make decisions and carry out actions accordingly depending on different inputs.

Human Beings make decisions all the time. For example, every morning, we make a decision between eating or not eating before starting our daily chores. Conditional statements allow us to represent such decision-making in JavaScript, from the choice that must be made.

JavaScript is a programming language that is commonly used to create interactive and dynamic elements on websites. One of the key features of JavaScript is the ability to use conditional statements to control the flow of a program.

Conditions work on boolean values, true or false. It is true if it meets the requirement, false otherwise. That is expressions (conditions) are evaluated to be either true or false.

Conditional Statements in JavaScript

There are three ways of writing conditionals in Javascript:

- If/else Statement
- Switch Statement
- Ternary Operator

In this lecture let's look at **if/else** conditional and later on will look at other conditional statements.

1. if statement

The most basic form of a conditional statement is the if statement. The syntax for an if statement is as follows:

```
JavaScript
if (condition) {
    // statements
}
```

The condition is any expression that can be evaluated as true or false. For example, you can use a comparison operator (such as `<`, `>`, `==`) to compare two values, or you can use a logical operator (such as `&&`, `||`) to combine multiple conditions.

Example: Divide only, when the divisor is `!= 0`

JavaScript

```
let dividend = 10;

let divisor = 5;

if(divisor != 0){
    let result = dividend/divisor;
    console.log(result)
}

// Output
// 2
```

2. if-else statement

An if statement can also include an optional else statement, which will execute if the condition is false. The syntax for an if-else statement is as follows:

JavaScript

```
if (condition) {
    //if statements
} else {
    //else statements
}
```

Example 1: Only age above 18 are eligible for registration.

JavaScript

```
age = 15;.

if(age < 18){
    console.log("Sorry, you are not eligible")
}else{
    console.log("Registered Successfully")
}

// Output
// Sorry, you are not eligible
```

Example 2: Allow only admin to fetch user details

```
isAdmin = false;

if(isAdmin){
    console.log("fetch user details")
}else{
    console.log("This operation is only for
Admins")
}

// Output
// This operation is only for Admins
```

3. Nested if else

We can nest if else statements inside another if else statements.

```
JavaScript
if (condition1) {
    if(condition2){
        // statement1
    }else{
        // statement2
    }
} else {
    if(condition3){
        // statement3
    }else{
        // statement4
    }
}
```

Example: Find the greatest number among 3 numbers.

JavaScript

```

let a = 10;
let b = 14;
let c = 54;

if(a > b){

    if(a > c){
        console.log("a is greatest");
    }else{
        console.log("c is greatest");
    }
}else{
    if(b > c){
        console.log("b is greatest");
    }else{
        console.log("c is greatest");
    }
}

// Output - c is greatest

```

4. If else ladder

JavaScript also supports the use of else if statement, which allows you to chain multiple conditions together also called if else ladder. The syntax for an if-else ladder is as follows:

JavaScript

```

if (condition1) {
    //statement1
} else if (condition2) {
    //statement2
} else {

    //statement3
}

```

Note 1: You can chain as many **else if** statements as you want.

Note 2: In the if-else ladder we can omit the last **else** block, it is not mandatory to put.

Example: The challenge here is to find out if the given number is even or odd.

Before solving any problem through programming it is important to first analyze what is the input that will be taken, what the conditionals involved, and what the output expected.

In this case, we will be taking integers as input. The output is expected to be a message telling if the number is even or odd.

The conditions to be considered to solve this problem are

2. Any number that is completely divisible [remainder must be 0] by 2 then it is an even number.
3. Any number that is not completely divisible [remainder must be 0] by 2 then it is an odd number.
4. Zero is neither an odd number nor an even number

Before handling each condition, let declare our input variable,

```
JavaScript
// Input

var num;
```

In the above block of code, we have declared a variable named **num**. The variable **num** will be our input.

Implement 1st condition

Now let's handle the 1st condition: that is if any number that is completely divisible [remainder must be 0] by 2 then it is an even number.

To check if the number is completely divisible by 2 we will be making use of the modulo operator (%) which returns the remainder. If the result of the modulo operation is 0 then the number is even.

```
JavaScript
// Handling 1st Condition

var num = 10;

// Condition 01: Any number that is completely
// divisible [ remainder must be 0 ] by 2 then it
// is an even number.
if (num % 2 == 0) {

    console.log("The number given is an even
    number");
}
```

If the condition is false, the code inside the if block will not be executed, and this code will not give any output.

Implement 2nd Condition:

Since we are not getting any output if the condition is false. It's time to handle the 2nd condition which is any number that is not completely divisible [remainder must be 0] by 2 then it is an odd number.

As we have already checked for the even number condition, now if the condition for even fails it is an odd number. We can check this through the else statement.

```
JavaScript
// Condition 01: Any number that is completely
// divisible [ remainder must be 0 ] by 2 then it
// is an even number
if (num % 2 == 0) {
    console.log("The number given is an even
number");
}

// Condition 02: Any number that is not
// completely divisible [ remainder must be 0 ] by
// 2 then it is an odd number.
else {
    console.log("The number given is an odd
number");
}
```

Now the code is capable of checking if the given number is odd or even.

Implementing 3rd Condition:

We also have our third condition which is that zero is neither an odd number nor an even number. So the first condition we need to check is if the number is zero, then if the number is even, and at last, if both the conditions fail it is an odd number

Final Solution

```
JavaScript
// Condition 03: Zero is neither an odd number
// nor an even number.
if (num == 0) {
    console.log("Zero is neither an odd number
nor an even number");
```

```
}

// Condition 01: Any number that is completely
// divisible [ remainder must be 0 ] by 2 then it
// is an even number.
else if (num % 2 == 0) {

    console.log("The number given is an even
number");
}

// Condition 02: Any number that is not
// completely divisible [ remainder must be 0 ] by
// 2 then it is an odd number.
else {
    console.log("The number given is an odd
number");
}
```

