JavaScript Conditional Statements

In JavaScript, **conditional statements** allow you to make decisions in your code. You can execute certain blocks of code based on whether a condition is true or false. This is very useful when you want your program to behave differently under certain conditions.

1. `if` Statement

The `if` statement checks a condition and runs a block of code if the condition is **true**.

Syntax:

```
if (condition) {
    // code to run if condition is true
}
```

Example:

```
let age = 18;

if (age >= 18) {
    console.log("You are an adult.");
}
// Output: You are an adult.
```

In this example, the code checks if the variable `age` is 18 or more. If true, it prints "You are an adult."

2. `else` Statement

The `else` statement provides an alternative block of code to run when the `if` condition is **false**.

Syntax:

```
if (condition) {
    // code to run if condition is true
} else {
    // code to run if condition is false
}
```

Example:

```
let age = 16;

if (age >= 18) {
    console.log("You are an adult.");
} else {
    console.log("You are a minor.");
}
// Output: You are a minor.
```

Here, if the age is less than 18, the `else` block runs and prints "You are a minor."

3. `else if` Statement

Sometimes, you need to check multiple conditions. In this case, you can use `else if` to test additional conditions if the first `if` condition is false.

Syntax:

```
if (condition1) {
    // code to run if condition1 is true
} else if (condition2) {
    // code to run if condition1 is false and condition2 is true
} else {
```

```
    // code to run if both condition1 and condition2 are false
  }
```

Example:

```
let score = 75;

if (score >= 90) {
    console.log("You got an A!");
} else if (score >= 80) {
    console.log("You got a B!");
} else if (score >= 70) {
    console.log("You got a C!");
} else {
    console.log("You failed.");
}
// Output: You got a C!
```

In this example:

- If the score is 90 or above, it prints "You got an A!"
- If the score is between 80 and 89, it prints "You got a B!"
- If the score is between 70 and 79, it prints "You got a C!"
- If none of these conditions are true, the final `else` block runs and prints "You failed."

4. **Ternary Operator** (Shorthand for `if-else`)

The **ternary operator** is a short form of an `if-else` statement. It's useful when you have a simple condition and want to write less code.

Syntax:

```
condition ? expressionIfTrue : expressionIfFalse;
```

Example:

```
let age = 20;
let message = age >= 18 ? "You are an adult." : "You are a minor.";
console.log(message);
// Output: You are an adult.
```

In this case, `age >= 18` is the condition. If it's true, the message is "You are an adult." If it's false, the message is "You are a minor."

5. `switch` Statement

The `switch` statement is used to compare one value to multiple possible values and execute different code based on the matching value. It's useful when you have many conditions to check.

Syntax:

```
switch (expression) {
    case value1:
        // code to run if expression matches value1
        break;
    case value2:
        // code to run if expression matches value2
        break;
    default:
        // code to run if no cases match
}
```

Example:

```javascript
let day = "Tuesday";

switch (day) {
    case "Monday":
        console.log("Start of the week.");
        break;
    case "Tuesday":
        console.log("Second day of the week.");
        break;
    case "Friday":
        console.log("End of the workweek.");
        break;
    default:
        console.log("It's a regular day.");
}
// Output: Second day of the week.
```

Here:

- If `day` is "Monday", it prints "Start of the week."
- If `day` is "Tuesday", it prints "Second day of the week."
- If no case matches (e.g., if the day was "Sunday"), the `default` block would run and print "It's a regular day."

Summary of Key Points

1. `if` : Runs code if a condition is true.
2. `else` : Runs code if the `if` condition is false.
3. `else if` : Checks additional conditions if the previous `if` is false.
4. **Ternary Operator**: A shorthand version of `if-else`.
5. `switch` : Used to compare one value to multiple possible cases.

Example Combining Everything:

```javascript
let temperature = 30;

if (temperature > 30) {
    console.log("It's hot outside.");
} else if (temperature < 15) {
    console.log("It's cold outside.");
} else {
    console.log("The weather is pleasant.");
}

// Using ternary
let weather = temperature > 30 ? "hot" : "not hot";
console.log(weather);  // Output: not hot

// Using switch
let season = "summer";

switch (season) {
    case "winter":
        console.log("Wear a jacket.");
        break;
    case "summer":
        console.log("Wear light clothes.");
        break;
    default:
        console.log("Dress normally.");
}
```

This shows how conditional statements allow JavaScript to make decisions based on different situations!