

Conditional statements like **if**, **else if**, **else**, and nested variations help control the flow of code based on certain conditions. Here's an explanation with examples:

if Statement:

The **if** statement executes a block of code if a specified condition is true.

```
const number = 10;

if (number > 0) {
  console.log("The number is positive");
}
```

else Statement:

The **else** statement executes a block of code if the same condition specified in the **if** statement is false.

```
const number = -5;

if (number > 0) {
  console.log("The number is positive");
} else {
  console.log("The number is non-positive");
}
```

else if Statement:

The **else if** statement allows us to specify multiple conditions to execute different blocks of code.

```
const number = 0;

if (number > 0) {
  console.log("The number is positive");
} else if (number < 0) {
  console.log("The number is negative");
} else {
  console.log("The number is zero");
}
```

Nested if-else Statements:

Nested **if-else** statements are using an **if** or **else** statement inside another **if** or **else** statement.

```
const number = -5;

if (number >= 0) {
  if (number === 0) {
    console.log("The number is zero");
  } else {
    console.log("The number is positive");
  }
} else {
  console.log("The number is negative");
}
```

Explanation:

- **if** checks a condition. If true, its block of code executes; otherwise, it's skipped.
- **else** checks an alternate condition when the **if** condition is false.
- **else if** allows adding another condition after an **if** to check multiple scenarios.
- Nested **if-else** statements help to check conditions within conditions for more complex scenarios.

These conditional statements are fundamental for controlling the flow of your program based on different conditions!