3. Control Flow

(a) Conditional Statements.

(i) if Statement

→ (1) checks a condition
(2) If the condition is true, executes the code block.
(3) If not, the program skips it.

Syntax :-
```
if (condition ) {
    // code to execute
}
```

(ii) if-else condition

→ (1) checks a condition
(2) If the condition is true, executes first block.
(3) If false, executes the else block.

Syntax :-
```
if (condition) {
    // code if true
} else {
    // code if false
}
```

(iii) else if Ladder

(1) Used for multiple conditions
(2) checks each if or else if until one is true
(3) If none are true, the else block runs.

Syntax :-
```
if (condition 1) {
    // code
} else if (condition 2) {
    // code
} else {
    // default code
}
```

Switch Condition

Best when you have multiple fixed values to match.
Cleaner than writing many if-else statements.

Syntax :

```
Switch (variable) {
    Case value1:
        //code
        break;
    Case value2:
        //code
        break;
    default:
        //code
}
```

Use Cases of Conditional Statements.

Features	Use Cases
if	Run Code if a specific Condition is true.
if - else	choose between two possibilities
else if	choose from multiple conditions.
Switch	Match a variable against fixed values (e.g. menu options, days).