Nested Statements / Loops

1. NESTED IF STATEMENT

A **nested if statement** is an `if` condition inside another `if` block. This allows checking multiple conditions **in a hierarchical manner**.

Syntax:

```
if (condition1) {
    if (condition2) {
        // Code to execute if both conditions are true
    }
}
```

Example 1: Nested If Statement

```
#include <stdio.h>

int main() {
    int num = 10;

    if (num > 0) {  // Outer if
        printf("The number is positive.\n");

        if (num % 2 == 0) {  // Inner if
            printf("The number is even.\n");
        }
    }

    return 0;
}
```

Output:

```
The number is positive.
The number is even.
```

---

2. NESTED WHILE LOOP

A **nested while loop** is a `while` loop inside another `while` loop. The **inner loop** executes completely **for each iteration of the outer loop**.

Syntax:

```
while (condition1) {
    while (condition2) {
        // Code to execute
    }
}
```

Example 2: Multiplication Table using Nested While Loop

```
#include <stdio.h>

int main() {
    int i = 1, j;

    while (i <= 5) {
        j = 1;
        while (j <= 5) {
            printf("%d\t", i * j);
            j++;
        }
        printf("\n");
```

```
        i++;
    }

    return 0;
}
```

Output:

```
1  2  3  4  5
2  4  6  8  10
3  6  9  12 15
4  8  12 16 20
5  10 15 20 25
```

---

3. NESTED DO-WHILE LOOP

A **nested do-while loop** is a `do-while` loop inside another `do-while` loop. The inner loop will always execute **at least once** before checking the condition.

Syntax:

```
do {
    do {
        // Code to execute
    } while (condition2);
} while (condition1);
```

Example 3: Number Grid using Nested Do-While

```
#include <stdio.h>

int main() {
```

```
    int i = 1, j;

    do {
        j = 1;
        do {
            printf("%d ", j);
            j++;
        } while (j <= 5);

        printf("\n");
        i++;
    } while (i <= 5);

    return 0;
}
```

Output:

```
1 2 3 4 5
1 2 3 4 5
1 2 3 4 5
1 2 3 4 5
1 2 3 4 5
```

---

4. NESTED FOR LOOP

A **nested for loop** is a `for` loop inside another `for` loop. The **inner loop runs completely** for each iteration of the outer loop.

Syntax:

```
for (initialization; condition1; increment) {
    for (initialization; condition2; increment) {
        // Code to execute
    }
}
```

Example 4: Printing a Square Pattern

```
#include <stdio.h>

int main() {
    for (int i = 1; i <= 5; i++) {
        for (int j = 1; j <= 5; j++) {
            printf("* ");
        }
        printf("\n");
    }

    return 0;
}
```

Output:

```
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
```

---

5. NESTED SWITCH-CASE

A **nested switch-case** is when a `switch` statement is placed inside another `switch` statement.

Syntax:

```
switch (variable1) {
    case value1:
        switch (variable2) {
            case value2:
                // Code to execute
```

```
            break;
        }
        break;
}
```

Example 5: Nested Switch-Case for User Role and Permission

```
#include <stdio.h>

int main() {
    int role = 1; // 1 = Admin, 2 = User
    int action = 2; // 1 = View, 2 = Edit

    switch (role) {
        case 1:
            printf("Role: Admin\n");
            switch (action) {
                case 1:
                    printf("Action: Viewing data\n");
                    break;
                case 2:
                    printf("Action: Editing data\n");
                    break;
                default:
                    printf("Invalid action!\n");
            }
            break;

        case 2:
            printf("Role: User\n");
            switch (action) {
                case 1:
                    printf("Action: Viewing data\n");
                    break;
                default:
                    printf("Users cannot edit data!\n");
```

```
            }
            break;

        default:
            printf("Invalid role!\n");
    }

    return 0;
}
```

Example Output:

```
Role: Admin
Action: Editing data
```

Revision #1
Created 7 February 2025 16:31:01 by BH
Updated 7 February 2025 16:39:04 by BH