

CONTROL STRUCTURES:

- USED TO CONTROL FLOW OF THE PROGRAM.
- THERE ARE THREE TYPES OF CONTROL STRUCTURES.

I. CONDITIONAL CONTROL STRUCTURES

II. BRANCHING CONTROL STRUCTURES

III. ITERATION CONTROL STRUCTURES

I. CONDITIONAL CONTROL STRUCTURES:

I. SIMPLE IF: IT CONTAINS ONLY TRUE BLOCK.

SYNTAX:

IF <CONDITION> THEN

<EXEC-STATEMENTS>; -- TRUE BLOCK

END IF;

II. IF. ELSE: IT CONTAINS BOTH TRUE BLOCK & FALSE BLOCK.

SYNTAX:

IF <CONDITION> THEN

<EXEC-STATEMENTS>; -- TRUE BLOCK

ELSE

<EXEC-STATEMENTS>; -- FALSE BLOCK

END IF;

III. NESTED IF:

-> IF WITHIN THE IF IS CALLED AS NESTED IF.

SYNTAX:

IF <CONDITION> THEN

IF <CONDITION> THEN

<EXEC-STATEMENT>;

ELSE

<EXEC-STATEMENTS>;

END IF;

ELSE

```
IF <CONDITION> THEN
    <EXEC-STATEMENT>;
ELSE
    <EXEC-STATEMENTS>;
END IF;
END IF;
```

IV. IF..ELSE LADER:

SYNTAX:

```
IF <CONDITION> THEN
    <EXEC-STATEMENTS>;
ELSIF <CONDITION> THEN
    <EXEC-STATEMENTS>;
ELSIF <CONDITION> THEN
    <EXEC-STATEMENTS>;
.....

ELSE
    <EXEC-STATEMENTS>;
END IF;
```

II. BRANCHING CONTROL STURCTURES:

I. CASE:

SYNTAX:

```
CASE <VARIABLE/EXPRESSION>
    WHEN <COND> THEN
        <EXEC-STATEMENTS>;
    WHEN <COND> THEN
        <EXEC-STATEMENTS>;
    WHEN <COND> THEN
        <EXEC-STATEMENTS>;
```

ELSE

<EXEC-STATEMENT>;

END CASE;

ITERATION CONTROL STATEMENTS:

I. SIMPLE LOOP:

-> IT IS AN INFINITE LOOP. IF WE WANT BREAK A SIMPLE LOOP THEN WE SHOULD USE "EXIT" STATEMENT.

SYNTAX:

LOOP

<EXEC-STATEMENTS>;

END LOOP;

II. WHILE LOOP:

SYNTAX:

WHILE <CONDITION>

LOOP

<EXEC-STATEMENTS>;

<INCR/DECR>;

END LOOP;

III. FOR LOOP:

-> BY DEFAULT, IT IS INCREMENTED BY 1.

SYNTAX:

FOR <INDEX_VARIABLE> IN <START_VALUE>..<END_VALUE>

LOOP

<EXEC-STATEMENTS>;

END LOOP;