029 -
C SWITCH STATEMENTS

The switch statement in C is an alternate to if-else-if ladder statement which allows us to execute multiple operations for the different possibles values of a single variable called switch variable.

```
SWITCH STATEMENT SYNTAX :-

switch (expression) {
    case value1:
        //code to be executed;
        break;  //optional

    case value2:
        //code to be executed;
        break;  //optional
    ......

    default:
        code to be executed if all cases are not
matched;
}
```

RULES FOR SWTICH STATEMENT IN C LANGUAGE

The switch expression must be of an integer or character type.

The case value must be an integer or character constant.

The case value can be used only inside the switch statement.

The break statement in switch case is not must. It is optional.
If there is no break statement found in the case, all the cases will be executed present after the matched case.
It is known as fall through the state of C switch statement.

FUNCTIONING OF SWITCH CASE STATEMENT

First, the integer expression specified in the switch statement is evaluated.

This value is then matched one by one with the constant values given in the different cases.

If a match is found, then all the statements specified in that case are executed along with the all the cases present after that case including the default statement.

No two cases can have similar values.

If the matched case contains a break statement, then all the cases present after that will be skipped, and the control comes out of the switch.

Otherwise, all the cases following the matched case will be executed.

C SWITCH STATEMENT IS FALL-THROUGH

In C language, the switch statement is fall through; it means if you don't use a break statement in the switch case, all the cases after the matching case will be executed.

NESTED SWITCH CASE STATEMENT

We can use as many switch statement as we want inside a switch statement. Such type of statements is called nested switch case statements.