

ENUMERATED DATA TYPES

- Enumeration or Enum in C is a special kind of data type defined by the user. It consists of constant integrals or integers that are given names by a user. The use of enum in C to name the integer values makes the entire program easy to learn, understand, and maintain by the same or even different programmer.

Syntax to Define Enum in C

- An enum is defined by using the 'enum' keyword in C, and the use of a comma separates the constants within. The basic syntax of defining an enum is:

```
enum enum_name{int_const1, int_const2, int_const3, .... int_constN};
```
- In the above syntax, the default value of int_const1 is 0, int_const2 is 1, int_const3 is 2, and so on. However, you can also change these default values while declaring the enum.

Example

- Below is an example of an enum named cars and how you can change the default values.

```
enum cars{BMW, Ferrari, Jeep, Mercedes-Benz};
```

- Here, the default values for the constants are:

BMW=0, Ferrari=1, Jeep=2, and Mercedes-Benz=3. However, to change the default values, you can define the enum as follows:

```
enum cars{
    BMW=3,
    Ferrari=5,
    Jeep=0,
    Mercedes-Benz=1
};
```

Enumerated Type Declaration to Create a Variable

- Similar to pre-defined data types like int and char, you can also declare a variable for enum and other user-defined data types. Here's how to create a variable for enum.

- **enum condition (true, false);** //declaring the enum
 - **enum condition e;** //creating a variable of type condition
- Suppose we have declared an enum type named condition; we can create a variable for that data type as mentioned above. We can also converge both the statements and write them as: **enum condition (true, false) e;**
- For the above statement, the default value for true will be 1, and that for false will be 0.

Implementing enum using C Program

Example program 2.19: Printing the Values of Weekdays

```
#include <stdio.h>
enum days{Sunday=1, Monday, Tuesday, Wednesday, Thursday, Friday,
Saturday};
int main(){
    // printing the values of weekdays
    for(int i=Sunday;i<=Saturday;i++){
        printf("%d, ",i);
    }
    return 0;
}
```

Output

1, 2, 3, 4, 5, 6, 7