

## CHAPTER 18



# Enum

Enum is a user-defined type consisting of a fixed list of named constants. In the example below, the enumeration type is called `Color` and contains three constants: `Red`, `Green` and `Blue`.

```
enum Color { Red, Green, Blue };
```

The `Color` type can be used to create variables that may hold one of these constant values.

```
int main()
{
    Color c = Red;
}
```

In order to avoid naming conflicts the constant can be qualified with the name of the enum.

```
Color c = Color.Red;
```

## Enum example

The `switch` statement provides a good example of when enumerations can be useful. Compared to using ordinary constants, the enumeration has the advantage that it allows the programmer to clearly specify what values a variable should contain.

```
switch(c)
{
    case Red:    break;
    case Green: break;
    case Blue:  break;
}
```

## Enum constant values

Usually there is no need to know the underlying values that the constants represent, but in some cases it can be useful. By default, the first constant in the enum list has the value zero and each successive constant is one value higher.

```
enum Color
{
    Red    // 0
    Green  // 1
    Blue   // 2
};
```

These default values can be overridden by assigning values to the constants. The values can be computed and do not have to be unique.

```
enum Color
{
    Red    = 5,        // 5
    Green  = Red,      // 5
    Blue   = Green + 2 // 7
};
```

## Enum conversions

The compiler can implicitly convert an enumeration constant to an integer. However, converting an integer back into an enum variable requires an explicit cast, since this conversion makes it possible to assign a value that is not included in the enum's list of constants.

```
int i = Red;
Color c = (Color)i;
```

## Enum scope

An enum does not have to be declared globally. It can also be placed within a class as a class member, or locally within a function.

```
class MyClass
{
    enum Color { Red, Green, Blue };
};

void myFunction()
{
    enum Color { Red, Green, Blue };
}
```