Const in C++

Whenever const keyword is attached to any method( ), variable, pointer variable, and with the object of a class it prevents that specific object/method()/variable from modifying its data items value.

```
// C++ program to demonstrate the
// the above concept
#include <iostream>
using namespace std;
// Driver Codeint main()
{

    // const int x; CTE error
    // x = 9; CTE error
    const int y = 10;
    cout << y;

    return 0;
}
```

Output

`10`

There are a certain set of rules for the declaration and initialization of the constant variables -:

- The const variable cannot be left un-initialized at the time of the assignment.
- It cannot be assigned value anywhere in the program.
- Explicit value needed to be provided to the constant variable at the time of declaration of the constant variable.

Ex -   const int num = 1;

Alternative to const - We can also use Macros to define constant, but there is a catch,

#define var 5

Since Macros are handled by the preprocessor(the pre-processor does text replacement in our source file, replacing all occurrences of ‘var’ with the literal 5), not by the compiler.

Hence it wouldn’t be recommended because Macros don’t carry type-checking information and are also prone to error.