Comments in C++

A well-documented program is a good practice as a programmer. It makes a program more readable and error finding become easier. One important part of good documentation is Comments.

- In computer programming, a comment is a programmer-readable explanation or annotation in the source code of a computer program
- Comments are statements that are not executed by the compiler and interpreter.

In C/C++ there are two types of comments :

1. Single line comment
2. Multi-line comment

Single line Comment

Represented as // double forward slash

It is used to denote a single line comment. It applies comment to a single line only. It is referred to as C++-style comments as it is originally part of C++ programming.

for example:

```
// single line comment
```

Example:

```
// C++ program to illustrate
// use of single-line comment
#include <iostream>int main()
{
    // Single line Welcome user comment
```

```
    std::cout << "Welcome to GeeksforGeeks";
    return 0;
}
```

Output

```
Welcome to GeeksforGeeks
```

Multi-line comment

Represented as **/* any_text */** start with forward slash and asterisk (/*) and end with asterisk and forward slash (*/).

It is used to denote multi-line comment.

```
/*Comment starts
continues
continues
.
.
.
Comment ends*/
```

Example:

```
/* C++ program to illustrate
use of
multi-line comment */
#include <iostream>int main()
{
    /* Multi-line Welcome user comment
    written to demonstrate comments
    in C/C++ */
    std::cout << "Welcome to GeeksforGeeks";
    return 0;
}
```

Output

```
Welcome to GeeksforGeeks
```

Comment at End of Code Line

You can also create a comment that displays at the end of a line of code. But generally its a better practice to put the comment before the line of code.

Example:

```
int age; // age of the person
```

When and Why to use Comments in programming?

1. A person reading a large code will be bemused if comments are not provided about details of the program.
2. Comments are a way to make a code more readable by providing more description.
3. Comments can include a description of an algorithm to make code understandable.
4. Comments can be helpful for one’s own self too if code is to be reused after a long gap.