Python Comments

Comments are lines that exist in computer programs that are ignored by compilers and interpreters.

Including comments in programs makes code more readable for humans as it provides some information or explanation about what each part of a program is doing.

In general, it is a good idea to write comments while you are writing or updating a program as it is easy to forget your thought process later on, and comments written later may be less useful in the long term.

In Python, we use the hash (#) symbol to start writing a comment.

In [2]:

```python
#Print Hello, world to console
print("Hello, world")
```

```
Hello, world
```

Multi Line Comments

If we have comments that extend multiple lines, one way of doing it is to use hash (#) in the beginning of each line.

In [3]:

```python
#This is a long comment
#and it extends
#Multiple lines
```

Another way of doing this is to use triple quotes, either ''' or """.

In [4]:

```python
"""This is also a
perfect example of
multi-line comments"""
```

Out[4]:

```
'This is also a\nperfect example of\nmulti-line comments'
```

DocString in python

Docstring is short for documentation string.

It is a string that occurs as the first statement in a module, function, class, or method definition.

In [8]:

```python
def double_1(num):
    """
    function to double the number
    """
    return 2 * num

print double_1(10)
```

```
  File "<ipython-input-8-98b5d4698b3f>", line 7
    print double_1(10)
                 ^
SyntaxError: invalid syntax
```

In [ ]:

```
print double.__doc__   #Docstring is available to us as the attribute __doc__  of the function
```

Python Indentation

1. Most of the programming languages like C, C++, Java use braces { } to define a block of code. Python uses indentation.
2. A code block (body of a function, loop etc.) starts with indentation and ends with the first unindented line. The amount of indentation is up to you, but it must be consistent throughout that block.
3. Generally four whitespaces are used for indentation and is preferred over tabs.

In [ ]:

```
for i in range(10):
    print i
```

Indentation can be ignored in line continuation. But it's a good idea to always indent. It makes the code more readable.

In [ ]:

```
if True:
    print "pyhton Learning"
    c = "MEN"
```

In [ ]:

```
if True: print "Machine Learning"; c = "MEN"
```

Python Statement

Instructions that a Python interpreter can execute are called statements.

Examples:

In [ ]:

```
a = 1   #single statement
```

Multi-Line Statement

In Python, end of a statement is marked by a newline character. But we can make a statement extend over multiple lines with the line continuation character ().

In [12]:

```
a = 1 + 2 + 3 + \
    4 + 5 + 6 + \
    7 + 8
```

In [11]:

```
print (a)
```

36

In [ ]:

```
#another way is
a = (1 + 2 + 3 +
     4 + 5 + 6 +
     7 + 8)
print a
```

In [13]:

```
a = 10; b = 20; c = 30    #put multiple statements in a single line using ;
```

In [3]:

```
a,b,c=20,100.7,"MG"
print (a)
print (b)
print (c)
```

```
20
100.7
MG
```