Python Introduction: ¶

- Python is high level programming language.
- It is developed by 'Guido Van Rossum' in the year of 1991.
- It is mainly used to create server side web applications.
- Python has a simple syntax similar to the english language.
- Python supports the object oriented programming.
- Python works on different platforms(windows,mac,linux,....etc)

In [2]:
```python
1  print('Hello world')
```

```
Hello world
```

In [3]:
```python
1  print("python programming")
```

```
python programming
```

Python Comments:

- Python comments can be used to explain python code.
- comments can be used to make the code more readable.
- compiler will not execute the comment section lines.
- Two types comments:
-   1. single line comments-> using '#' symbol
-   2. multi-line comments-> using tripple quotes ->''' '''

In [8]:
```python
# To print the hello world
# This is a comment
print('hello world')
'''welcome to apssdc
summer online internship
for diploma students'''
```

```
hello world
```

Out[8]: `'welcome to apssdc\nsummer online internship\nfor diploma students'`

In [13]:
```python
# TO join two strings using '+' operator?
print('welcome'+'python')
```

```
welcomepython
```

In [12]:
```python
# To join string and integer using ,
print('python',123)`
```

```
python 123
```

Python Variables:

- Variable name is used to store the single data.
- Rules of a variable declaration:
- 1. variable name should starts with alphabets,alphabets followed by digits. ex: abc = 100 , abc12334='python'
- 2. variable name should not allow the space in the middle of two words. ex: first name = 'abc'
- 3. To join two words using underscore. ex: first_name = 'abc'
- 4. Variable name should not starts with digits,special characters,keywords,inbuilt functions.. ex: 12 = 100, @abc = 12 , while = 45 , sum = 'python'
- Variable names are case-sensitive.

In [14]:
```python
# examples on python variables...
a = 10
print(a)
```

10

In [15]:
```python
b = 'python'
print('b')
```

b

In [16]:
```python
abc123 = 'apssdc'
print(abc123)
```

apssdc

In [17]:
```python
12 = 100
```

```
  File "<ipython-input-17-361d96271702>", line 1
    12 = 100
    ^
SyntaxError: cannot assign to literal
```

In [18]:
```python
sum = 10
print(sum)
```

10

In [19]:
```python
1  while = 10
2  print(while)
```

```
  File "<ipython-input-19-17ee9e643c69>", line 1
    while = 10
          ^
SyntaxError: invalid syntax
```

In [20]:
```python
1  %a = 10
```

```
UsageError: Line magic function `%a` not found.
```

In [23]:
```python
1  # case sensitive....
2  n1 = 100
3  print(n1)
```

```
100
```

In [25]:
```python
1  # To assign the single value to the different varible names..
2  a=b=c=10
3  print(a)
4  print(b)
5  print(c)
```

```
10
10
10
```

In [26]:
```python
1  # To assign the different values to different variable names?
2  a,b,c,d = 10,20,30,40
3  print(a,b,c,d)
```

```
10 20 30 40
```

Python Data Types:

- Data Type:
  - Data type is used to define the type of value to be used in a program.
- To know the data type we can use the type method():
- syntax: print(type(variable_name)

```python
In [28]:  1  # TO store the integer data.
          2  n = 123
          3  print(n)
          4  print(type(n))
```

```
123
<class 'int'>
```

```python
In [29]:  1  # TO store the string data.
          2  s = 'python'
          3  print(s)
          4  print(type(s))
```

```
python
<class 'str'>
```

```python
In [30]:  1  # To store the floating data.
          2  f = 23.45
          3  print(f)
          4  print(type(f))
```

```
23.45
<class 'float'>
```

Type casting:

- To convert the data into one data type to another data type.
- 1. TO convert the integer to float: float(integer_data)

- 2. To convert the float to integer: int(float_data)
- 3. To convert the string to integer: int(string_data)

In [32]:
```python
# TO convert the integer data into float?
n = 12 # integer data
print(type(n))
n = float(n) # Floating conversion
print(n)
print(type(n))
```

```
<class 'int'>
12.0
<class 'float'>
```

In [33]:
```python
# To convert the floating into integer?
f = 34.56
print(type(f))
f = int(f) # integer conversion
print(f)
print(type(f))
```

```
<class 'float'>
34
<class 'int'>
```

In [36]:
```python
# To convert the string data into integer?
s1 = '123'
print(type(s1))
s1 = int(s1)
print(type(s1))
```

```
<class 'str'>
<class 'int'>
```

input method:

- By using input method we can read the data from a user.

- syntax: variable_name = data_type(input(prompt message))

In [38]:
```python
# To read the integer data from a user?
m = int(input("enter m value: "))
print(m)
print(type(m))
```

```
enter m value: 12
12
<class 'int'>
```

In [40]:
```python
# TO floating data..
f1 = float(input("enter floating data: "))
print(f1)
```

```
enter floating data: 12
12.0
```

In [ ]:
```python
LMS Exam Login Instructions

Dear Student,
While appearing for your LMS exam, please follow these login instructions carefully:

👤 Username: Your Email ID registered on the Naipunyam Portal
🔑 Password: The Mobile number (used during registration on the Naipunyam Portal)

❓ If you do not know your registered email ID or mobile number,
✅ Please log in to your Naipunyam portal, click on your Profile and check your details overthere.

📌 Make sure your email and mobile number are active and updated.

All the Best!
```