Python:

Object oriented

Scripting

Multipurpose - web programming

High level

Interpreted

Dynamically typed

It is much faster because ,It is don’t need compilation.

PythonPath:

Variable tells the python interpreter where to locate **the module file imported into the program**.

PythonStartsUp:

**It contains path of initialization file** containing the python source code.

PEP 8:

Coding convention , set of recommendation how to write your python code.

Bugs tool:

Pychecker tool used to analyze the detect bugs

Variables:

Variable data types based on the values.

Like a=12 ,b=”opti”,c=23.232

Multiple assignment :

A=b=c=23;

A,b,c=12,32,43;

Tuples :

Collection of different data types , but cannot be change

( )

List:

Collection of different data types ,can be changed .

[ ]

Dictionary :

Collection key and value pair

{}

Name.keys();

Name.values();

Operator:

\*\* - power

< > - not equal

and,or,not - logical

**Membership Operator:**

in and not in. Ex: if(a in list)

**Identity Operator:**

Is => =

Not is => !=

Commands:

Single - #

Multi line - ‘’’ ‘’’

If elif else:

If 1==b:

Print

Elif b==c:

Print

Else:

print

for ;

for <variable> in <sequence>: ex: for i in range (1,10):

while;

while a>b:

pass statement :

do not need any code execute you can use pass statement

OOPS concepts :

Class and object:

Class classname :

Def display():

Print

Emp1=classname()

Emp1.display()

Constructor:

Def \_init\_(): - this is the synytax

Inheritance:

Class a:

Def ab():

Class b(a):

Def bc():

B1=b()

B1.ab()

B1.bc()

Multiple - class a(b,c):

String :

Concatenation = st1+st2

Replication = st1\*5

Membership = st1 in st2

Relational = st1=>st2

Slice = st1[startindex:endindex]

St1[stratindex:]

St1[:endindex]

Important:

'abc'.capitalize() - integer

msg.count(substr1, 4, 16) - integer

string1.endswith(substring1) - true/false

str.find(substr1) - start index

stt.index(‘CO’) - integer

st1.isalnum() - t/f

str.isalpha() - t/f

str.isdigit() - t/f

str.islower() - “

str.isupper() - “

str.isspace() - “

len(str) - integer

str.lower() - convert to lower

str.upper() - convert to upper

str.stratswith(“hello”) - t/f

str.swapcase() -

str.lstrip(‘@’)

str.rstrip(‘#’)

  List :

<list\_name>=[value1,value2,value3,...,valuen];

1. data1=[1,2,3,4];  data1[0:]   - [1,2,3,4] data1[0:2] - [1,2]
2. data2=['x','y','z'];  data2[:2]  - [‘’x’,’y’] data2[0] - [‘x’]

Add List : - List=list1+list2;

Replication - list1\*3

Update - list[1]=”opti”

Append - list1.append(10.23);

Delete - del list1[1]

Important methods:

Min - min(list1);

Max - max(list1);

Len - len(list1);

Cmp - Cmp(list1,list2)

List - list(tuple);

Other:

List1.Index(“opti”)

list1.count(“opti”)

list1.pop() or pop(1)

list1.insert(“12”) or list1.insert(2,12)

list1.remove()

list1.reverse ()

list1.sort()

Tuple :

Same as list

But not used other methods

Dictionaries:

Access - data[‘key’]

Delete - del data[‘key’]

Function:

Len - len(dict)

Cmp - cmp(dict1,dict2);

Str - str(dict)

Methods:

Keys - dict.keyys()

Values - dict.values()

Items - dict.items()

Update - dict1.update(dict2)

Clear - dict1.clear()

Copy - dict1.copy()

Has\_key - dict1.has\_key(“name”)

Get - dict1.get()

Function:

Anonymous function:

Square =lamda x:x\*x

I/O :

Output:

Print “”;

Input:

Input(“enter the name”)

Raw\_input(“”) `- return string

Int(raw\_input()) - integer

Float(raw\_input()) - float

File:

obj=open(filename , mode , buffer)

fileobject.close()

fileobject.write(string str)

fileobject.read(value)

obj2.read(20)

obj = open("data.txt", "w")

**print**  obj.name

**print**  obj.mode

**print**  obj.closed

**MODULE:**

**Import** square

from area import square,rectangle

from area import \*

date/time :

**import** time;

localtime = time.localtime(time.time())

|  |  |
| --- | --- |
| tm\_year | Returns the current year |
| tm\_mon | Returns the current month |
| tm\_mday | Returns the current month day |
| tm\_hour | Returns the current hour. |
| tm\_min | Returns the current minute |
| tm\_sec | Returns current seconds |
| tm\_wday | Returns the week day |
| tm\_yday |  |

**import** time

t = time.localtime()

printtime.asctime(t)

time.sleep( 10 )

**import** calendar

**print** "Current month is:"

cal = calendar.month(2014, 6)

**import** calendar

calendar.prcal(2014)

Exception Handling:

1. ZeroDivisionError: Occurs when a number is divided by zero.
2. NameError: It occurs when a name is not found. It may be local or global.
3. IndentationError: If incorrect indentation is given.
4. IOError: It occurs when Input Output operation fails.
5. EOFError: It occurs when end of file is reached and yet operations are being performed

Syntax:

**try**:

malicious code

**except** Exception1:

execute code

**except** Exception2:

execute code

....

....

**except** ExceptionN:

execute code

**else**:

finally :

**try**:

Code

**finally**:

user defined exception:

**class** ErrorInCode(Exception):

**def** \_\_init\_\_(self, data):

self.data = data

**def** \_\_str\_\_(self):

**return** repr(self.data)

**try**:

**raise** ErrorInCode(2000)

**except** ErrorInCode as ae:

**print** "Received error:", ae.data

Case sensitive:

Yes ,it is case sensitive.

Data types:

Numbers

String

Tuples

List

Dictionary

Difference between tuples and list:

Tuples are immutable()

List are mutable[]

Mutable:

List ,set ,dictionaries

Immutables:

String ,numbers ,tuples

Lamda:

Single expression , anonymous function used as inline function.

Anonymous :