Introduction to python

Python is a widely used general-purpose, high level programming language. It was initially designed by Guido van Rossum in 1991 and developed by Python Software Foundation. It was mainly developed for emphasis on code readability, and its syntax allows programmers to express concepts in fewer lines of code.

Python is a programming language that lets you work quickly and integrate systems more efficiently.

There are two major Python versions- **Python 2 and Python 3**. Both are quite different.

Beginning with Python programming:

1) Finding an Interpreter:

Before we start Python programming, we need to have an interpreter to interpret and run our programs. There are certain online interpreters .

Windows:There are many interpreters available freely to run Python scripts like IDLE (Integrated Development Environment) which is installed when you install the python software from http://python.org/

Linux:For Linux, Python comes bundled with the linux.

Keywords

print
print to console

while
controlling the flow of the program

for
iterate over items of a collection in order that they appear

break
interrupt the (loop) cycle, if needed

continue

used to interrupt the current cycle, without jumping out of the whole cycle. New cycle will begin.

if
used to determine, which statements are going to be executed.

elif
stands for else if.If the first test evaluates to False, then it continues with the next one

else
is optional. The statement after the else keyword is executed, unless the condition is True

is
tests for object identity

not
negates a boolean value

and
all conditions in a boolean expression must be met

or
at least one condition must be met.

import
import other modules into a Python script

as
if we want to give a module a different alias

from
for importing a specific variable, class or a function from a module

def
used to create a new user defined function

return
exits the function and returns a value

lambda
creates a new anonymous function

global
access variables defined outside functions

try

specifies exception handlers

except
catches the exception and executes codes

finally
is always executed in the end. Used to clean up resources.

raise
create a user defined exception

del
deletes objects

pass
does nothing

assert
used for debugging purposes

class
used to create new user defined objects

exec
executes Python code dynamically

yield
is used with generators