Values and types: A value is one of the most basic things in any program works with. A value may be characters i.e. ‘Hello, World!’ or a number like 1,2.2 ,3.5 etc.Values belong to different types: 1 is an integer, 2 is a float and ‘Hello, World!’ is a string etc.

First, we type the python command to start the interpreter.

type('Hello, World!') <class 'str'> type(17) <class 'int'>

Numbers: Python supports 3 types of numbers: integers, float and complex number. If you want to know what type a value has you can use type() function. Paste the following code and click the run button to check the output.

print(type(1))

print(type(2.2))

print(type(complex(2,3)))

Strings: Strings are defined either with a single quote or a double quotes. The difference between the two is that using double quotes makes it easy to include apostrophes.

print(type('Hello World'))

print(type("Today's News Paper"))

Variables: A variable is nothing but a name that refers to a value. An assignment statement creates new variables and gives them values.

print(type(name))

print(type(id))

print(type(height))

Variable Names: A Programmer should always choose a meaningful name for their variable. Rules for Variable Names:

1.A variable can contain both letters and numbers, but they cannot start with a number. So, variable1 is valid while 1variable is a invalid name.

2.We can use uppercase letters for variable names but it is always perfectly fine to begin variable names with a lowercase letter.

3.If your Variable name is long, then you can use underscore character (\_) in the name. For example, top\_five\_members, var\_1 etc. all are valid example.

4.You can’t use special characters like !, @, #, $, % etc. in variable name.

5.Python keywords cannot be used as variable name.

If you give a variable an illegal name, you will get a syntax error: 1var=20 class=5 global=10 all@1=100

Keywords: Python reserves 35 keywords for its use. Keywords are case sensitive in python. You can’t use a keyword as variable name, function name or any other identifier name. Here is the list of keywords in python.

Python reserves 35 keywords: and del from None True as elif global nonlocal try assert else if not while break except import or with class False in pass yield continue finally is raise a sync def for lambda return await

Statements: Statements are instructions or piece of codes that Python interpreter can execute. We have already seen two kinds of statements: print and assignment. There are other kinds of statements like if statement, for statement, while statement etc. When you type a statement, the interpreter executes it and displays the result, if something is there. If you write a script it usually contains a sequence of statements. If there is more than one statement, the results appear one at a time as the statements execute one by one.

print(100)

x = 200

y=400

z=x+y

print(z)

output:100 600

Multi-line statement: In Python, end of a statement is marked by a newline character. But You can write a statement with multiple lines using character (). Check the following example. st = "I " + "am" + " Mr." +" X."+" I live in "  
"city Y."

print(st)

Line continuation is implied inside parentheses ( ), brackets [ ] and braces { } in Python. This is called explicit line continuation. For example, you can write the above multi-line statement as the following code.

st = ("I " + "am" + " Mr." + " X."+" I live in " "city Y.")

print(st)

In Python, end of a statement is marked by a newline character. But You can write a statement with multiple lines using character (). Check the following example. You can use [ ] and { } for the same purpose described above.

st = ["I " + "am " + "Mr. " + " X."+" I live in "+ "city Y"]

print(st)

Python : Indentation One of the most distinctive features of Python is its use of certain indentation style to mark blocks of code. Once you are wrting python code just be careful of few things:

1. In Python white spaces are important.

2. The indentation is important.

3. If you write program that is not correctly indented shows either errors or does not give result what you want.

4. Python is case sensitive.

5. You can’t safely mix tabs and spaces in Python.