20 May

Python Basic - 1

* 1. What are keywords in python? Using the keyword library, print all the python keywords.
* The keywords are the reserved words in python. The following is the way to print all

python keywords:

import keyword

print(keyword.kwlist)

Output:

['False', 'None', 'True', 'and', 'as', 'assert', 'async', 'await', 'break', 'class', 'continue', 'def', 'del', 'elif', 'else', 'except', 'finally', 'for', 'from', 'global', 'if', 'import', 'in', 'is', 'lambda', 'nonlocal', 'not', 'or', 'pass', 'raise', 'return', 'try', 'while', 'with', 'yield']

* 1. What are the rules to create variables in python?
* The rules to create variables in python are as follows:
* Starting letter can be A-Z, a-z or \_.
* Numbers from 0-9 can be written in between but not in the starting letter.
* The name of the variable should not be exactly same as any keyword.
  1. What are the standards and conventions followed for the nomenclature of variables in python to improve code readability and maintainability?
* The standards and conventions followed for the nomenclature of variables in python to improve code readability and maintainability are as follows:
* We should write the Python function name with all lower-case characters.
* We should not use uppercase character while naming a function in python.
* We should use underscore (\_) in between the words instead of space while naming a function.
  1. What will happen if a keyword is used as a variable name?
* If a keyword is used as a variable name it will give “SyntaxError”.
  1. For what purpose def keyword is used?
* def keyword is used to create a function.
  1. What is the operation of this special character ‘\’?
* First, the backslash character is a part of special character sequences such as the tab character \t or the new line character \n. Second, the backslash (\) escape other special characters. For example, if you have a string that has a single quote inside a single-quoted string, you need to use the backslash to escape the single quote character.
  1. Give an example of the following conditions:

1. Homogeneous list 🡪 [1, 2, 3]
2. Heterogeneous set 🡪 {“iNeuron”, 23, True, 8.0}
3. Homogeneous tuple 🡪 (‘r’, ‘l’, ‘t’)
   1. Explain the mutable and immutable data types with proper explanation & examples.

* The mutable data types include lists, dictionaries and sets. These data types can be changed or modified after they are created. For e.g.:

li = [‘a’, ‘b’, ‘c’, ‘d’]

print(li) # [‘a’, ‘b’, ‘c’, ‘d’]

li[2] = ‘g’

print(li) # [‘a’, ‘b’, ‘g’, ‘d’]

The immutable data types include tuples. These data types cannot be changed or modified after they are created. For e.g.:

tup = (‘a’, ‘b’, ‘c’, ‘d’)

print(tup) # (‘a’, ‘b’, ‘c’, ‘d’)

tup[2] = ‘t’ # TypeError: 'str' object does not support item assignment

* 1. Write a code to create the given structure using only for loop.

\*

\*\*\*

\*\*\*\*\*

\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*

* The code to create the given structure using only for loop is as follows:

rows=5

for i in range(rows):

space = rows – i

count = (2 \* i) + 1

print(“ ” \* space, “\*” \* count)

* 1. Write a code to create the given structure using while loop.

|||||||||

|||||||

|||||

|||

|

* The code to create the given structure using only while loop is as follows:

rows = 5

i = 0

while i != rows:

space = i

count = 2 \* (rows - i) – 1

print(“ ” \* space, “|” \* count)

i += 1