20 May

**Python Basic – 1**

**Assignment-6**

**Q.1. What are keywords in python? Using the keyword library, print all the python keywords.**

**Ans:**  Keywords in Python are the reserved words that have special predefined functionality and specific meaning. These all keywords cannot be used as variable name or any class or functions name. They all are used to perform reserved tasks in python program.

e.g: True, False, None, and, or, break, def, class etc.

We can access all keywords by using below statement.

A computer screen with white text

Description automatically generated

***Final output:*** *['False', 'None', 'True', '\_\_peg\_parser\_\_', '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']*

**Q.2. What are the rules to create variables in python?**

**Ans:** Variables are used to store data. There are some rules to follow while creating a variable given below.

* Variable name must be start with letter (a-z, A-Z) or an underscore (\_). They can contain letters, numbers and underscores.
* We cannot use fixed keywords as a variable name, For example “if, for, True, False, None, and, or, break” etc.
* It is allowed to use underscore or double underscore at the beginning of variable name but should avoid it because some of internal variables are stars with underscore.
* Don’t start variable name with numbers it will give an error.

**Q.3. What are the standards and conventions followed for the nomenclature of variables in**

**python to improve code readability and maintainability?**

**Ans:** In Python, there are several standards and conventions followed for variable naming to improve code readability and maintainability. These conventions are often referred to as Pythonic naming conventions. Below are the some commonly followed standards.

* Variable names are typically written in lower case letters.
* To improve readability, multiple words in variable name are separated by an underscore. The convention is known as snake\_case. For example

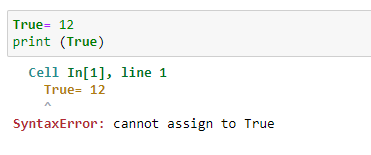
my\_variable, total\_expense etc.

* Choose variable names that are descriptive and convey the purpose or content of the variable. This helps make our code more self-explanatory and easier to understand.
* Do not use reserved keywords as variable name, as they have predefined meaning in Python.
* For the constant variables with values that should not change throughout the code use all uppercase letters with underscore. E.g. MAX\_NUM
* Class name should follow the CamelCase convention starting with uppercase. E.g. MyClass.

**Q.4. What will happen if a keyword is used as a variable name?**

**Ans:** If a keyword is used as a variable name in Python, it will result in a syntax error. Keywords are reserved words that have predefined meanings and specific functionality within the Python language.

For example:



In the above example we used “True” as a variable name, but it is a predefined keyword in python that’s why we get above error.

**Q.5. For what purpose def keyword is used?**

**Ans:** We use “def” keyword to define a function. Function is nothing but a block of code which is used to do repetitive tasks within a code.

**Q.6. What is the operation of this special character ‘\’?**

**Ans:** In Python, the special character “ \”is known as the backslash or escape character.

It is used to indicate that the character following it has a special meaning. The backslash is used in combination with certain characters to represent special characters, escape sequences, or Unicode characters. Here are some common uses of the backslash in Python:

* **Escape Sequences:** The backslash is used to represent escape sequences, which are special character combinations. Some commonly used escape sequences are:

\n - Newline

\t - Tab

\' - Single quote

\" - Double quote

\\ - Backslash

Example:

A white rectangular object with black text

Description automatically generated

* **Unicode Characters:** The backslash is used to represent Unicode characters using their hexadecimal value. It is done using the “\u” or “\U” followed by the hexadecimal value of the character.

Example:

A white rectangular object with red and blue text

Description automatically generated

* **Line Continuation:** The backslash can be used to continue a single logical line of code across multiple physical lines. It is often used when a line of code is too long and needs to be split for better readability.

Example:

A white background with black and green numbers

Description automatically generated

* **Raw String:** The backslash can be used as a prefix to create a raw string literal. In a raw string, backslashes are treated as literal characters and not as escape characters.

Example:

A white box with red text

Description automatically generated

**Q.7. Give an example of the following conditions:**

**(i) Homogeneous list**

A homogeneous list in Python is a list that contains elements of the same data type. In below example1 we have a list of numbers which are having same data type as integer. While in example2 we have a list of countries the elements in list are all having string data type.

Example: A black screen with white text

Description automatically generated

**(ii) Heterogeneous set:**

**Ans:** A heterogeneous set in Python that contains elements of different data types.

Example:



In above example my\_set contains the element of different data type.

1 is integer, ‘apple’ is string, 3.14 is float data type and (4, 5, 6) is a tuple.

**(iii) Homogeneous tuple**

**Ans:**  Tuples are the data structure of python. It is immutable and enclosed in rounded braces “()”.

The homogeneous tuple in python that contains elements of same data types.

Example:



**Q.8. Explain the mutable and immutable data types with proper explanation & examples.**

**Ans:** In Python, data types are categorized as either mutable or immutable based on whether their values can be changed after they are created.

**Immutable Data Types:**

Immutable data types are those whose values cannot be modified once they are created. Any attempt to change the value of an immutable object will result in the creation of a new object.

Examples of immutable data types in Python include:

* Numbers (int, float, complex)
* Strings (str)
* Tuples (tuple)
* Frozen sets (frozenset)

Example:

A screenshot of a computer

Description automatically generated

**Mutable Data Types:**

Mutable data types, in contrast, allow changes to their values without creating a new object. The object's identity remains the same even after modifications.

Examples of mutable data types in Python include:

* Lists (list)
* Sets (set)
* Dictionaries (dict)

Example:

A screenshot of a computer program

Description automatically generated

**Q.9. Write a code to create the given structure using only for loop.**

**\***

**\*\*\***

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

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

**Ans: A computer screen shot of a code

Description automatically generated**

**Q.10. Write a code to create the given structure using while loop.**

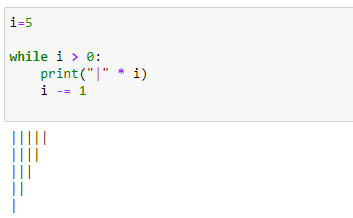
**|||||||||**

**|||||||**

**|||||**

**|||**

**|**

**Ans: **