**Python Assignment -1**

**Topic (python variables)**

**Q1. Write all the conventions being followed while declaring a variable.**

**Answer.**

• A variable name can contain alphabets, digits, and underscore.

• A variable name can only start with an alphabet and underscore.

• A variable can’t start with a digit.

• No white space is allowed to be used inside a variable name.

**Q2. What will happen if we declare a restricted keyword as a variable?**

**Answer.**

If you try to declare the restricted keyword as a variable , you will get a syntax error because restricted keywords have a predefined language and they cannot be used a variable names.

**Q3.** **Can we actually declare a string as a variable name in python?**

**Answer.**

No we cannot declare a string as a variable name in python . In python the variable name must follow certain rules and restrictions .

According to the python documentations the variable name must:

* Start with letter or underscore.
* Consists of letter , numbers and underscores.
* Not be a python keywords or inbuilt function name.

Therefore using a string as a variable name violates these rules and results in syntax error.

**Q4. Is it possible for us to declare “\_” as a variable? If so, then write an example of it.**

**Answer.**

Yes it is possible to declare the “\_”as a variable name python , the underscore character is a valid variable name ,and it has special meaning in certain contexts .

For example “\_” is often used as a throwaway variable name in situations where the value of variable is not needed .Here is an example :

# Assigning a value to the "\_" variable

\_ = "Hello, World!"

# Using the "\_" variable as a throwaway variable

for \_ in range(5):

print("Hello")

# Accessing the value of the "\_" variable

print(\_)

**Q5. Using an example, explain how the variables in python are dynamic in nature.**

**Answer.**

In python variables are dynamic in nature ,which means they can be assigned values of different types at runtime .This allow for greater flexibility in coding makes a python highly versatile language .

For example

a = 5

print(a)

a = "Hello, World!"

print(a)

a = [1, 2, 3, 4, 5]

print(a)