**Python Assignment-1**

**1. In the below elements which of them are values or an expression? e.g.:- values can be integer or string and expressions will be mathematical operators.**

**\*** - Expression (Multiplication)

**‘Hello’ -** Values (String)

**-87.8** - Values (Integer)

**-** - Expression (Subtraction)

**/ -** Expression (Divide)

**+ -** Expression (Addition)

**6** - Values (Integer)

**2. What is the difference between string and variable?**

A string is a data type used to represent a sequence of characters. It can contain letters, numbers, symbols, and spaces.

A variable is a named container that holds a value. It allows you to store and manipulate data.

The main difference between string and variable is strings are immutable, variables can be mutable.

**3. Describe three different data types.**

The Three different data types in python are:

* Integer
* Float
* String

Integer (int):

An integer is a whole number without any decimal points. It can be a positive or negative number, or even zero.

E.g.: x=1, y=2

Float (float):

A float, short for "floating-point number," is a number that can have a decimal point. It's used to represent real numbers, including both integers and fractional values.

E.g.: pi=3.14

String (str):

A string is a sequence of characters, such as letters, digits, symbols, and spaces. Strings are used to represent textual data.

E.g.: Name=”Mani”

**4. What is an expression made up of? What do all expressions do?**

In Python, an expression is a combination of values, variables, operators, and function calls that can be evaluated to produce a result. Expressions are the building blocks of any program's logic and computation.

Expressions can do as follows:

* Calculations: Expressions can be used to perform arithmetic calculations, such as addition, subtraction, multiplication, and division.
* Comparison: Expressions involving comparison operators are used to compare values and determine relationships between them.
* Boolean Logic: Expressions using logical operators help with boolean logic and decision-making in programs.
* String Manipulation: Expressions can manipulate strings by concatenating them, extracting substrings, and more.

**5. This assignment statements, like spam = 10. What is the difference between an expression and a statement?**

An assignment statement, like spam = 10, is both a statement and an expression because it both assigns a value (10) to the variable spam and produces the value 10 as a result of the expression.

An expression is a combination of values, variables, operators, and function calls that can be evaluated to produce a value.

A statement is a complete unit of code that performs an action. Statements can include expressions, but they don't necessarily have to.

**6. after running the following code, what does the variable bacon contain?**

**bacon = 22**

**bacon + 1**

After running the given code:

bacon = 22

bacon + 1

The variable bacon will still contain the value 22.

In Python, when you perform an operation like bacon + 1, it produces a new value (23 in this case), but this value is not stored or assigned to any variable. So the first assigned value will be the output.

**7. What should the values of the following two terms be?**

**'spam' + 'spamspam'**

**'spam' \* 3**

The values of 'spam' + 'spamspam':

This expression concatenates the string 'spam' with the string 'spamspam'

Output: 'spamspamspam'.

The values of 'spam' \* 3:

This expression repeats the string 'spam' three times

Output: 'spamspamspam'.

**8. Why is eggs a valid variable name while 100 is invalid?**

In Python, variable names follow certain rules and conventions. While eggs is a valid variable name and 100 is not, this is because of the following reasons:

Starting Character:

Variable names in Python must start with a letter (a-z or A-Z) or an underscore (\_). They cannot start with a number. This is why eggs is a valid variable name because it starts with a letter.

However, 100 start with a number, which is not allowed.

**9. What three functions can be used to get the integer, floating-point number, or string version of a value?**

In Python, you can use the following three functions to convert values to different types:

* Integer Conversion: int()

The int () function is used to convert a value to an integer.

E.g.: value = 4.56

integer\_value = int (value) ## Converts the float value 4.56 to the integer 4

* Floating-Point Conversion: float()

The float () function is used to convert a value to a floating-point number.

E.g.: value = 4

float\_value = float (value) # Converts the integer 4 to the floating-point number 4.0

* String Conversion: str()

The str () function is used to convert a value to a string.

E.g.: number = 42

string\_number = str (number) # Converts the integer 42 to the string "42"

**10. Why does this expression cause an error? How can you fix it? 'I have eaten ' + 99 + ' burritos.'**

The expression 'I have eaten ' + 99 + ' burritos.' causes an error because we are trying to concatenate a string ('I have eaten ') with an integer (99) using the + operator. In Python, you can concatenate strings using the + operator, but you cannot directly concatenate a string with a non-string without converting it to a string first.

To fix it implement this code:

'I have eaten ' + str (99) + ' burritos.'