1. What are the differences between operators and values in the following?

Sol:

**Operators** are used to perform the operations (Addition, multiplication , modulus ,assignment etc) on values or variables.

Values are the one on which operations are performed.

\* multiplication operator

'hello' –string value

-87.8 negative float value

- subtraction operator

/ division operator

+ addition operator

6 integer value

1. What is the difference between string and variable?

Sol: **STRING** – A string is a derived datatype that holds a sequence of characters. Strings are immutable. This means that once defined, they cannot be changed. Many Python methods, such as replace(), join(), or split() modify strings. However, they do not modify the original string. They create a copy of a string which they modify and return to the caller. It is usually written within quotes.

**VARIABLE** - Variables are used to store data, they take memory space based on the type of value we assigning to them.

A variable can be a string data type or float data type and many more depending on the type of data.

Spam - variable

'spam' - string

1. Describe three different data types.

Variables can store data of different types, and different types can do different things. Python has the following data types built-in by default, in these categories:

|  |  |
| --- | --- |
| Text Type: | str |
| Numeric Types: | int, float, complex |
| Sequence Types: | list, tuple, range |
| Mapping Type: | dict |
| Set Types: | set, frozenset |
| Boolean Type: | bool |
| Binary Types: | bytes, bytearray, memoryview |

1. What makes up an expression? What are the functions of all expressions?

An **expression** is a combination of one or more operands, zero or more operators, and zero or more pairs of parentheses.

There are three kinds of expressions:

A **arithmetic expression** evaluates to a single arithmetic value.

A **character expression** evaluates to a single value of type character.

A **logical** or **relational** **expression** evaluates to a single logical value.

1. In this chapter, assignment statements such as spam = 10 were added. What's the difference between a declaration and an expression?

Declaration of a variable is when you try to create a variable along with its type.

However, Python is solely OOP language and it does not need any variable declaration. In python, a value to variable is assigned and it comes into existence.

Expression: represent something, like a number, a string, or an instance of a class. Any value is an **expression**.

Spam=10 is expression.

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

bacon = 22

bacon + 1

22 is the answer

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

'spam' + 'spamspam' ------🡪 spamspamspam

'spam' \* 3 -----🡪spamspamspam

8. Why is it that eggs is a true variable name but 100 is not?

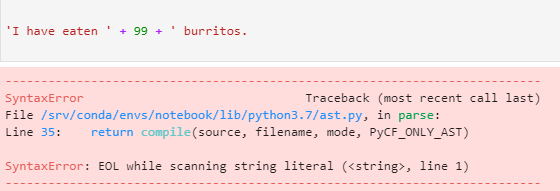
There are few rules that you have to follow while naming the variables in Python.  
\* The name of the variable must always start with either a letter or an underscore (\_). For example: \_str, str, num, \_num are all valid name for the variables.  
\* The *name of the variable cannot start with a number*. For example: 9num is not a valid variable name.  
\* The name of the variable cannot have special characters such as %, $, # etc, they can only have alphanumeric characters and underscore (A to Z, a to z, 0-9 or\_).  
\* Variable name is case sensitive in Python which means num and NUM are two different variables in python.

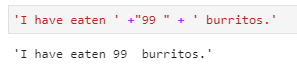
9. Which of the following three functions may be used to convert a value to an integer, a floating-point number, or a string?

Sol: int(value) ,float(value), str(value)

10. What is the error caused by this expression? What would you do about it?

'I have eaten ' + 99 + ' burritos.'



--- putting 99 in string format since 99 is integer and it cant be concatenated with string.