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

\*

‘hello’

-87.8

-

/

+

6

Answer:

\*, -, + , / 🡪 are arithmetic operations which helps in performing addition, multiplication , division and subtraction.

‘hello’ 🡪 is a string datatype. Strings are stored in between single/double quotes.

-87.8 🡪 is a negative float value. Here -ve sign gives more value to 87.8 float value saying that the value is negative.

2. What is the difference between string and variable?

spam

‘spam’

Answer:

Spam 🡪 variable

‘spam’ 🡪 string data type

String 🡪 string is one of the data type. Mostly it is represented in between single quotes/double quotes. Eg: ‘spam’ (or) “spam”

Variable🡪 Variable usually stores the values of different datatype like list,string,tuples and other data types. Eg: spam is variable which it can store value . spam = ‘This is spam mail’. Here in spam variable we are storing value “This is spam mail”.

3. Describe three different data forms.

Answer:

There are different types of data types available in python and below are details.

|  |  |
| --- | --- |
| 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 |

String 🡪 string is one of the data type. Mostly it is represented in between single quotes/double quotes. Eg: ‘spam’ (or) “spam”

Int 🡪 x = 20. Here we are storing value of 20 in variable x. When we check type of of x we get output as <class ‘int’>

List 🡪 x = ["apple", "banana", "cherry"]. Here we stored three different values in list format in x. Basically we can store any datatype in list which would be an added advantage and which made it widely used in programming.

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

Answer:

An expression is a combination of values, variables, operators, and calls to functions. Expressions need to be evaluated. If you ask Python to print an expression, the interpreter evaluates the expression and displays the result.

string is an expression. An expression is anything that has "a value". Like 3, 'Hello world', 1+1, math.sqrt(9), etc. Function names are expressions too.

eval() gives you the value of the expression that you give to it as a string. If you say eval('1+1') it returns 2. So it returns the same that would be returned if you simply write: 1+1.

5. In this chapter, assignment statements such as spam = 10 were added. What’s the

difference between a declaration and an expression?

Answer:

* Spam = 10 🡪 here we are assigning 10 to a variable spam. This is referred as statement too. Statements are like instructions for python interpreter that can perform certain execution. Other types of statements are for , if , while etc.
* An expression is a combination of values, variables, operators, and calls to functions. Expressions need to be evaluated. If you ask Python to print an expression, the interpreter evaluates the expression and displays the result.

string is an expression. An expression is anything that has "a value". Like 3, 'Hello world', 1+1, math.sqrt(9), etc. Function names are expressions too.

eval() gives you the value of the expression that you give to it as a string. If you say eval('1+1') it returns 2. So it returns the same that would be returned if you simply write: 1+1.

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

bacon = 22

bacon + 1

Answer: 23

Because here bacon is having value 22 which is considered as int datatype. So indirectly we are refereeing to the value 22 which is stored in bacon which displays result as 23(bacon =22 🡪 bacon+1 => 22+1 🡺23).

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

‘spam’+’spamspam’

‘spam’\* 3

Answer:

‘spam’+’spamspam’ 🡪'spamspamspam'

‘spam’\* 3 🡪'spamspamspam'

Observed that answer would be same for both scenarios.

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

Answer:

Eggs🡪 started with a alphabet

100 🡪 which is a number

 According to rules of python , a variable name must start with a letter or the underscore character. A variable name cannot start with a number. A variable name can only contain alpha-numeric characters and underscores (A-z, 0-9, and \_ )

9. Which of the following three functions may be used to convert a value to an integer, a

floating-point number, or a string?

Answer: Incomplete question but trying to provide possible scenarios.

As per my assumption, we are trying to discuss about type forecasting i.e., to convert from one data type to other data type.

Syntax to convert into int datatype 🡪 int() 🡪 ex: int(9.5) 🡪 which gives output as 9 as it doesn’t consider float value.

Syntax to convert into float datatype 🡪 float() 🡪 ex: float(9) 🡪 which gives output as 9.0 as it adds .0 to the int value.

Syntax to convert into string datatype 🡪 str() 🡪 ex: str(9.0) 🡪 which gives output as ‘9.0’ as we know that string can be represented in single/double quotes.

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

‘I have eaten‘+ 99 + ‘ burritos. ‘

Answer:

It throws a Type Error which says tries to concatenate only strings and not int. Here 99 is int data type and rest other sentence is in string format. So to concatenate 99 with rest other sentence we need to convert 99 int value to string value as shown : ‘99’ (99 should be in single/double quotes so that python interpreter consider it as string instead of int).