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

**\***

**'hello'**

**-87.8**

**-**

**/**

**+**

**6**

A. In python operators are like symbols used to perform some sort of computation . The values that holds are known as operands.

Eg: a = 10 + 20 where a is operand and + is operator , 10 and 20 are values.

\*, -, /, + are Arthimetic operators that are used in computation process. Lets say

A+B where A = 10 and b = 20 then the result will be as 30 sicce operator + uis used to add

* \* = multiplies value on either side i.e a\*b
* - = subtracts right hand to left hand i.e a - b
* / = division i.e a/b
* + = add two operands i.e a+b

6, ‘hello’, -87.8 are values or expression we can say.

Lets say a = 6, s = ‘hello’, f = -87.8 are values which are referenced by variables

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

**spam**

**'spam'**

A.String is the type of information which store using variable in enclosed punctuate. String is datatype for a sequence of characters, in addition to letters, they can also include numbers, spaces and punctuations.

Variables are references that you create to refer some value in programe.

Example: my\_value = "Hello World". In this my\_value referes as variable and "Hello World" is string

spam = ‘spam’ here, spam which is left hand side is variable and ‘spam’ on right hand side is string

**3. Describe three different data forms.**

A. Three different data forms in python are

a.Numeric

b.Boolean

c.Set

a.Numeric data type repreents numeric data values. It can be float, integer or complex number.

INTEGER: The integer value represented by int class. It can be whole numbers, in python there is no limit to how long integer value can be.

FLOAT: The float value represented by float class. It is real numbers with floating poiunt representation, it has decimal point also

COMPLEX NUMBERS: Complex values are repreented by complex class. It is represented as (realpart) + (imaginary part)

a = 5

print("Type of a" + type(a)) ---------> Type of a <class 'int'>

a = 2.5

print("Type of a" + type(a)) ---------> Type of a <class 'float'>

a = 2 + 3j

print("Type of a" + type(a)) ---------> Type of <class 'complex'>

b.Boolean data type represents two type of values i.e TRUE or FALSE. Boolean object when they are equal then it is TRUE or else FALSE.

in python we should Use T and F for boolean i.e True and False otherwise throws error

a = 10

if a == 10:

print(type(True)) -----> <class 'bool'>

else

print(type(False)) -----> <class 'bool'>

c. Set is unordered collection of data type that is iterable , mutable and has no duplicate elements.

Two operations of sets i.e creating set and accessing set

1. creating set: set can be created using built in set() which is iterable. we can create set using list also

set1 = set("hello")

print("\nSet with the use of String: ") output: Set with the use of String:

{'e', 'h', 'l', 'o'}

print(set1)

set1 = set(["Apple", "For", "Apple"])

print("\nSet with the use of List: ") output : Set with the use of List:

{'For', 'Apple'}

print(set1)

2. Accessing set elements:

Set items cannot be accessed by referring to an index, since sets are unordered the items has no index. But you can loop through the set items using a for loop.

set1 = set(["Apple", "For", "Apple"])

for i in set1:

print(i)

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

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

Eggs = 10 this could be an expression

Functions for expressions:

f(10) = 10

f(false) = false

f(0) or f(false) or f(1) = 1

f(0) and f(false) and f(1) = false

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

A. In python declaration of variable is not required. If there need of variable you think that you have to store integer value then you can directly assign the value to the variable .

Eg: spam = 10 here spam is variable and its value is 10.

Expression represents some value like number, string, or even instance of class. Any value is an expression we can say.

Eg: spam = 10 here it is statement but expression is something that has value i.e 10 which is an simple expression More complex expressions evaluate like adding numbers. Eval() will evaluates the string format in python

Lets say eval(“2+2”) --🡪 4

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

**bacon = 22**

**bacon + 1**

A. bacon = 22

print(bacon + 1)

output: 23

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

**'spam' + 'spamspam'**

**'spam' \* 3**

1. print('spam' + 'spamspam')

output: spamspamspam

print('spam'\*3)

output: spamspamspam

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

Eggs is true variable name because in python we cannot begin name of variable with integer values , since it is case sensitive,

The following are the rules of declaring variable name:

1. A variablename must **start** with a letter or the underscore character
2. A variablename cannot **start** with a number.
3. A variablename 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?**

The three functions are

1. INT: **int()**

a = 2.1

print(“int value “ + int(a))

output: 2

1. FLOAT: **float()**

a = 2

print(“int value “ + float(a))

output: 2.0

1. STRING: **str()**

a = 2.1

print(“int value “ + str(a))

output: 2.1

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

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

1. eggs='I have eaten' + 99 + 'burritos’

print(eggs)

output:

It throws error as int value cannot be convert to string

Improvement of above:

-----------------------------

eggs='I have eaten' + str(99) + 'burritos’

print(eggs)

output:

I have eaten 99 burritod