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

\*

Answer: Expression

'hello'

Answer: string

-87.8

Answer: Integer

-

Answer: Expression

/

Answer: Expression



Answer: Expression

6

Answer: Integer

2. What is the difference between string and variable?

**Answer:**

**String:** String is any value that we assign to a variable using single or double quote.

For Example: x=”Amit”, y=”7896b”

Hence, the values for x and y denotes string and are stored under single or double quote.

**Variable:** A variable is like an identifier that is used to store a value in a memory location and to manipulate if needed.

For example; X= 20 and Y=50

A variable is created when we first assign a value to it in our program. In the above program, two variables are created when we assigned the value 20 to X and the value 50 to Y.

3. Describe three different data types.

**Answer:**

The three data types are the following:

* Numeric
* Boolean
* Set

Numeric: Numeric data type represent the data which has numeric value. Numeric value can be integer, floating number, or even complex numbers. There are 3 types of numeric data types

* Integers – This value is represented by int class. It contains positive or negative whole numbers (without fraction or decimal). In Python there is no limit to how long an integer value can be.
* Float – This value is represented by float class. It is a real number with floating point representation. It is specified by a decimal point. Optionally, the character e or E followed by a positive or negative integer may be appended to specify scientific notation.
* Complex Numbers – Complex number is represented by complex class. It is specified as (real part) + (imaginary part)j. For example – 2+3j

Boolean: The data type with two in-built values ‘True’ or ‘False is called Boolean Data Type. Boolean objects that are equal to True are (true), and those equal to False are (false)

Set: Set is an unordered collection of data type that is iterable and has no duplicate elements. The order of elements in a set is undefined however it may consist of various elements. Sets are unordered and indexing has no meaning in it. We cannot access or change an element of a set using indexing or slicing. Set data type does not support it.

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

**Answer:**

An expression is a combination of operators and operands that is interpreted to produce some other value. An expression is evaluated as per the precedence of its operators. So that if there is more than one operator in an expression, their precedence decides which operation will be performed first. We have many different types of expressions in Python. They are as follows:

**Constant Expressions:** These are the expressions that have constant values only.

**Arithmetic Expressions:** An arithmetic expression is a combination of numeric values, operators, and sometimes parenthesis. The result of this type of expression is also a numeric value. The operators used in these expressions are arithmetic operators like addition, subtraction, etc.

**Integral Expressions:** These are the kind of expressions that produce only integer results after all computations and type conversions.

**Floating Expressions:** These are the kind of expressions which produce floating point numbers as result after all computations and type conversions.

**Relational Expressions:** In these types of expressions, arithmetic expressions are written on both sides of relational operator (> , < , >= , <=). Those arithmetic expressions are evaluated first, and then compared as per relational operator and produce a boolean output in the end. These expressions are also called Boolean expressions.

**Logical Expressions:**These are kinds of expressions that result in either *True*or *False.*It basically specifies one or more conditions. For example, (10 == 9) is a condition if 10 is equal to 9. As we know it is not correct, so it will return False.

**Bitwise Expressions:**These are the kind of expressions in which computations are performed at bit level.

**Combinational Expressions:**We can also use different types of expressions in a single expression, and that will be termed as combinational expressions.

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

**Answer:**

|  |  |
| --- | --- |
| **Statement** | **Expression** |
| An expression evaluates to a value | A statement executes something |
|  |
| The evaluation of a statement does not change state | The execution of a statement changes state |  |
|  |
| Evaluation of an expression always Produces or returns a result value. | Execution of a statement may or may not produces or displays a result value, it only does whatever the statement says. |  |
|  |
| Every expression can’t be a statement. | Every statement can be an expression. |  |
|  |

An Expression always evaluates to a value.

And,

A statement does something, like creating a variable or displaying a value, it only does whatever the statement says.

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

bacon = 22

bacon + 1

**Answer:** The variable bacon contains a numeric value which is when is when processed with a mathematical operator + and other numeric value 1 gives the output as 23.

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

'spam' + 'spamspam'

'spam' \* 3

**Answer:** Both term will have the same value 'spamspamspam'

Graphical user interface, text, application

Description automatically generated

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

**Answer**:

Because the variable cannot begin with a number.

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

**Answer:**

str()

int()

float()

10. Why does this expression cause an error? How can you fix it?

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

**Answer:**

This expression causes and error because in this line 'I have eaten' and 'burritos' are strings, while 99 is treated as integer. In order to fix the error and print 'I have eaten 99 burritos.', 99 needs quotes (“) around it to treat it as a string.