**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.**

\* *-> expression*

'hello' *-> value*

-87.8 *-> value*

- *-> expression*

/ *-> expression*

* *-> expression*

6 *-> value*

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

*Variables are symbols that can be used to store data in a program. You can think of them as an empty box that can be filled with some data or value. Strings are data, so we can use them to fill up a variable.*

*For example, myVar = 6*

*myVar2 = ‘python’*

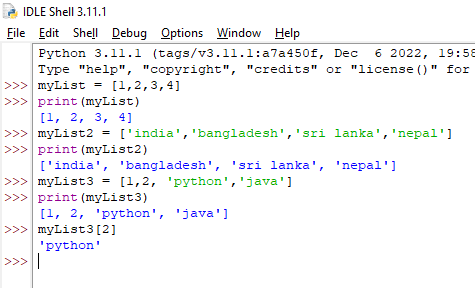
*Both myVar and myVar2 are variables but the value stored by myVar2 is a string ‘python’*

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

***LIST:*** *I) It is an ordered sequence of data written inside a pair of square brackets [] and separated by comma.*

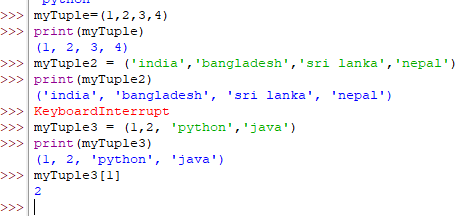
*ii) It can hold different type of data simultaneously.*

*iii) It follows zero based indexing*

**

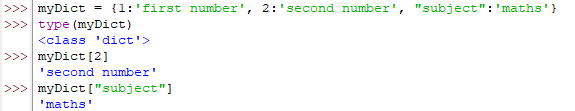
***TUPLE:*** *i) It’s also ordered sequence of data like list but written inside a pair of parenthesis*

*ii) Tuple is immutable.*

**

***DICTIONARY:*** *i) It’s an unordered sequence of data in key-value pairs*

*ii) These are written inside curly braces in the form key:value format*

**

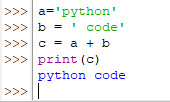
**4. What is an expression made up of? What do all expressions do?**

*A python expression is made up of two things:*

*i) Operators - special symbols that represents some sort of computations or operations, e.g. +, -*

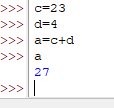
*ii) Operands - the values on which operators act on to perform any computation or operation*

*For example, in the below snippet, operator is + for performing concatenation on operands a and b*

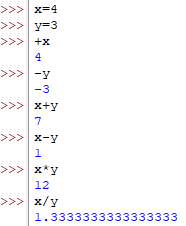
**

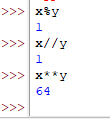
*Expressions are sequences of operators and operands to perform some computation or operation and generate some value accordingly. Expressions get evaluated based on the precedence of the operators by the Python interpreter. There are various operators and different types of data. Hence there are different kinds of expressions:*

***Constant expression -***

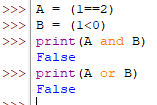
**

***Arithmetic expression***

******

**

***Logical Expressions***

******

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

*iExpression ultimately evaluates to a value always but a statement may or may not produce a value, rather it executes something what it says. For example, the above statement just assigns a value to the variable spam.*

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

**bacon = 22**

**bacon + 1**

*23*

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

**'spam' + 'spamspam'**

**'spam' \* 3**

*'spamspamspam'*

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

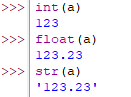
*As per the rules of variable names, the name must start with a letter or a underscore character but can;t start with number. That’s why 100 is invalid variable name.*

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

*int()*

*float()*

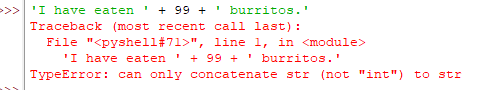
*str()*

**

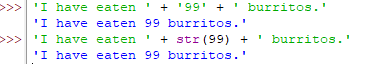
**10. Why does this expression cause an error? How can you fix it?**

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

*The Python interpreter can’t concatenate two different type of data, i.e. string and integer, That’s why it gives us objection:*

**

*If we enclose the integer 99 inside a quote or explicitly convert it into str, the issue will be fixed:*

**