**Assignment\_8**

Q1. Is the Python Standard Library included with PyInputPlus?

Ans: PyInputPlus is not a part of the Python Standard Library, so you must install it separately using Pip

Q2. Why is PyInputPlus commonly imported with import pyinputplus as pypi?

Ans: pypi is alias of PyInputPlus.

The as pyip code in the import statement saves us from typing pyinputplus each time we want to call a PyInputPlus function. Instead we can use the shorter pyip name

Q3. How do you distinguish between inputInt() and inputFloat()?

Ans: inputInt() : Accepts an integer value, and returns int value

inputFloat() : Accepts integer/floating point value and returns float value

Both takes additional parameters ‘min’, ‘max’, ‘greaterThan’ and ‘lessThan’ for bounds

Q4. Using PyInputPlus, how do you ensure that the user enters a whole number between 0 and 99?

Ans: In the inputint function we can set the min = 0 and max =99 to ensure user enters number between 0 and 99

pyip.inputInt(min = 0, max =99)

180

Number must be at maximum 99.

3

Output : 3

Q5. What is transferred to the keyword arguments allowRegexes and blockRegexes?

Ans: We can also use regular expressions to specify whether an input is allowed or not. The allowRegexes and blockRegexes

keyword arguments take a list of regular expression strings to determine what the PyInputPlus function will accept or

reject as valid input.

Q6. If a blank input is entered three times, what does inputStr(limit=3) do?

Ans: it will throw RetryLimitException exception.

response = pyip.inputStr(limit=3)

Q7. If blank input is entered three times, what does inputStr(limit=3, default='hello') do?

Ans: When you use limit keyword arguments and also pass a default keyword argument, the function returns the default value instead of raising an exception

response = pyip.inputStr(limit=3,default='hello')

response

Blank values are not allowed.

Blank values are not allowed.

Blank values are not allowed.

Output :

'hello'