1. Is the Python Standard Library included with PyInputPlus?

**No, PyInputPlus is not a part of Python Standard Library, it needs to be installed explicitly using the command !pip install PyInputPlus**

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

**can import the module with import pyinputplus as pyip so that you can enter a shorter name when calling the module's functions.**

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

**inputInt() function Accepts an integer value. This also takes additional parameters min, max, greaterThan and lessThan for bounds. And it always returns an int.**

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

**PyInputPlus module provides a function called as inputInt() which only returns only integer values. inorder to restrict the input between 0 and 99, i'ii use parameters like min & max to ensure that user enters the values between the defined range only.**

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

we can use **allowRegexes** and **blockRegexes** keyword arguments to take list of regular expression strings to determine what the pyinputplus function will reject or accept valid input.

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

he statement **inputStr(limit=3)** will throw two exceptions **ValidationException** and **RetryLimitException**. The first exception is thrown because blank values are not allowed by inputStr() function by default. it we want to consider blank values as valid input, we have to set **blank=True**.

The second exception is occured because we have reached the max limit we have specified by using **limit** parameter. inorder to avoid this exception we can use **default** parameter to return a default value when max limit is reached.

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

Since the default parameter is set to **hello**. after blank input is entered three times instead of raising **RetryLimitException** exception. the function will return **hello** as response to the calling function