1. Is the Python Standard Library included with PyInputPlus?

**Ans.** No, PyInputPlus is not included in the Python Standard Library. It is a third-party library that you need to install separately in order to use it.

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

**Ans.** The reason why PyInputPlus is often imported using "import pyinputplus as pypi" is because the `as pypi` part provides an alias for the module, making it easier and quicker to reference in the code. It is a common convention to use short and concise alias names for commonly used modules, especially when the original name is lengthy or difficult to type repeatedly. By using `pypi` as the alias, the programmer can access all the functions and classes within the PyInputPlus module by simply typing `pypi` instead of the full `pyinputplus`.

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

**Ans.** `inputInt()` and `inputFloat()` are both functions in Python used to take input from the user, but there is a difference between them.

`inputInt()` is used to take integer input from the user. When using `inputInt()`, the user is expected to enter an integer value, and if the user enters any other data type like a string or a float, a `ValueError` will be raised.

On the other hand, `inputFloat()` is used to take floating-point input from the user. In this case, the user can enter a decimal value, and the input will be automatically converted to a float. If the user enters a string or an integer, a `ValueError` may be raised, unless the string can be converted to a float.

To summarize, if you want to take integer input from the user, use `inputInt()`, and if you want to take floating-point input, use `inputFloat()`.

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

**Ans.** To ensure that the user enters a whole number between 0 and 99 using PyInputPlus, you can use the `inputInt` function and specify the minimum and maximum values allowed as arguments. Here's an example:

import pyinputplus as pyip

response = pyip.inputInt(prompt='Enter a whole number between 0 and 99: ', min=0, max=99)

In this example, the `inputInt` function is used to prompt the user to enter an integer value. The `min` argument is set to 0, meaning that the smallest allowed value is 0. The `max` argument is set to 99, meaning that the largest allowed value is 99. The function will continue to prompt the user until a valid integer within the specified range is entered.

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

**Ans.** The "allowRegexes" and "blockRegexes" are parameters in a function or method, and they are transferred as keyword arguments. Keyword arguments are a type of argument in which the name of the argument is specified when it is passed to a function or method. In this case, "allowRegexes" and "blockRegexes" are the names of the keyword arguments that are being passed to the function or method. The values assigned to these keyword arguments determine the behaviour of the function or method when it is executed.

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

**Ans.** "inputStr(limit=3)" is a function in a programming language that allows a user to input a string (a sequence of characters) from the keyboard, with a limit of 3 attempts. If a blank input is entered three times, the function will terminate and return an error or a default value, depending on how it is programmed. The purpose of setting a limit of 3 attempts is to prevent an infinite loop if the user repeatedly enters a blank input, and to provide a way to handle such cases in the code.

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

**Ans.** "inputStr(limit=3, default='hello')" means that if an empty input is entered three times in a row, the function will return the string "hello" as the default value. The "limit" parameter sets the maximum number of consecutive empty inputs that can be entered before the default value is returned. In this case, the limit is set to 3, so if an empty input is entered three times in a row, the function will return "hello".