Q1. Is it permissible to use several import statements to import the same module? What would the goal be? Can you think of a situation where it would be beneficial?

Ans : Nothing, if a module has already been imported, it's not loaded again. You will simply get a reference to the module that has already been imported (it will ...

Q2. What are some of a module's characteristics? (Name at least one.)

Ans : Characteristics of Modules

* Modules contain instructions, processing logic, and data.
* Modules can be separately compiled and stored in a library.
* Modules can be included in a program.
* Module segments can be used by invoking a name and some parameters.
* Module segments can be used by other modules.

Q3. Circular importing, such as when two modules import each other, can lead to dependencies and bugs that aren't visible. How can you go about creating a program that avoids mutual importing?

Ans : A circular dependency occurs when two classes depend on each other. For example, class A needs class B, and class B also needs class A. Circular dependencies can arise in Nest between modules and between providers. While circular dependencies should be avoided where possible, you can't always do so.

Q4. Why is \_ \_all\_ \_ in Python?

Ans : In the \_\_init\_\_.py file of a package \_\_all\_\_ is a list of strings with the names of public modules or other objects. Those features are available to wildcard imports. As with modules, \_\_all\_\_ customizes the \* when wildcard-importing from the package.

Q5. In what situation is it useful to refer to the \_ \_name\_ \_ attribute or the string '\_ \_main\_ \_'?

Ans : The value of \_\_name\_\_ attribute is set to “\_\_main\_\_” when module is run as main program. Otherwise, the value of \_\_name\_\_ is set to contain the name of the module. We use if \_\_name\_\_ == “\_\_main\_\_” block to prevent (certain) code from being run when the module is imported.

Q6. What are some of the benefits of attaching a program counter to the RPN interpreter application, which interprets an RPN script line by line?

Ans : An advantage of reverse Polish notation is that it removes the need for parentheses that are required by infix notation. While 3 − 4 × 5 can also be written 3 − (4 × 5), that means something quite different from (3 − 4) × 5.

Q7. What are the minimum expressions or statements (or both) that you'd need to render a basic programming language like RPN primitive but complete— that is, capable of carrying out any computerised task theoretically possible?

Ans : An assignment statement always has a single variable on the left hand side. The value of the expression (which can contain math operators and other variables) on the right of the = sign is stored in the variable on the left.