Q1. Is an assignment operator like += only for show? Is it possible that it would lead to faster results at the runtime?

Ans : The += operator is used to add two numbers together. The result of the addition will be stored in the variable that is on the left side of the += symbol. In this example, the result of adding y (15) to x (25) is stored in the variable x .

Q2. What is the smallest number of statements you'd have to write in most programming languages to replace the Python expression a, b = a + b, a?

Ans :  (b) abc Explanation: In Python, the "+" operator acts as a concatenation operator between two strings.

Q3. In Python, what is the most effective way to set a list of 100 integers to 0?

Ans : In Python, a list is created by placing elements inside square brackets [] , separated by commas. A list can have any number of items and they may be of different types (integer, float, string, etc.). A list can also have another list as an item. This is called a nested list.

Q4. What is the most effective way to initialise a list of 99 integers that repeats the sequence 1, 2, 3? S If necessary, show step-by-step instructions on how to accomplish this.

Ans : With the use of hashing we can finding the size of longest increasing sequence with consecutive integers in time complexity of O(n).

This problem can be solved easily by the concept of[LIS](https://www.geeksforgeeks.org/longest-increasing-subsequence/)where each next greater element differ from earlier one by 1. But this will take O(n^2) time complexity.  
With the use of hashing we can finding the size of longest increasing sequence with consecutive integers in time complexity of O(n).

We create a hash table.. Now for each element arr[i], we perform hash[arr[i]] = hash[arr[i] – 1] + 1. So, for every element we know longest consecutive increasing subsequence ending with it. Finally we return maximum value from hash table.

Q5. If you're using IDLE to run a Python application, explain how to print a multidimensional list as efficiently?

Ans : To print out the entire two dimensional array we can use python for loop as shown below. We use end of line to print out the values in different rows.

Q6. Is it possible to use list comprehension with a string? If so, how can you go about doing it?

Ans : List comprehension in Python is an easy and compact syntax for creating a list from a string or another list. It is a very concise way to create a new list by performing an operation on each item in the existing list. List comprehension is considerably faster than processing a list using the for loop.

Q7. From the command line, how do you get support with a user-written Python programme? Is this possible from inside IDLE?

Ans : If you’ve recently downloaded Python onto your computer, then you may have noticed a new program on your machine called **IDLE**. You might be wondering, “What is this program doing on my computer? I didn’t download that!” While you may not have downloaded this program on your own, IDLE comes bundled with every Python installation. It’s there to help you get started with the language right out of the box. In this tutorial, you’ll learn how to work in Python IDLE and a few cool tricks you can use on your Python journey!

Q8. Functions are said to be “first-class objects” in Python but not in most other languages, such as C++ or Java. What can you do in Python with a function (callable object) that you can't do in C or C++?

Ans : Python is a general-purpose, high-level programming language. Python is considered to be cleaner and more direct, with emphasis code readability.

Some differences between C++ and Python:

* Python uses Garbage Collection whereas C++ does not.
* C++ is a statically typed language, while Python is a dynamically typed language.
* Python is easier to use than C++.
* Python is run through an interpreter, whilst C++ is pre-compiled.
* Hence, C++ is faster than Python.
* C++ supports pointers and incredible memory management.
* Python supports very fast development and rapid, continuous language development.
* Python has less

Q9. How do you distinguish between a wrapper, a wrapped feature, and a decorator?

Ans : Wrappers around the functions are also knows as decorators which are a very powerful and useful tool in Python since it allows programmers to modify the behavior of function or class. Decorators allow us to wrap another function in order to extend the behavior of the wrapped function, without permanently modifying it.

Q10. If a function is a generator function, what does it return?

Ans : Simply speaking, a generator is a function that returns an object (iterator) which we can iterate over (one value at a time).

Q11. What is the one improvement that must be made to a function in order for it to become a generator function in the Python language?

Ans : If a function contains at least one yield statement (it may contain other yield or return statements), it becomes a generator function. Both yield and return will return some value from a function.

Q12. Identify at least one benefit of generators.

Ans : It allows you more mobility: generator sets allow you to have a power supply in areas lacking an electrical network. Therefore, they eliminate obstacles when it comes to carrying out projects and they contribute to the development of remote or disadvantaged areas.