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

Answer:

The assignment operator += denotes “Add and Assign”. It refers to add right side operand with left side operand and then assign to left operand. This operator leads to faster operation at the runtime. "+=" operator modifies the object directly, while "=" operator creates a new object and binds the variable to it.

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?

Answer:

The minimum number of lines required to write the above Python expression in most programming languages is 4 lines, while 2 lines for initial value assignment of variables a and b and next 2 lines for reassignment of values i.e., a = a+b and b = a.

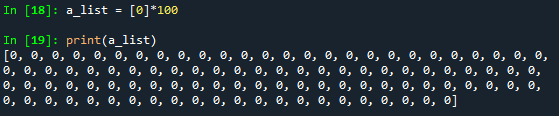
Q3. In Python, what is the most effective way to set a list of 100 integers to 0?

Answer:

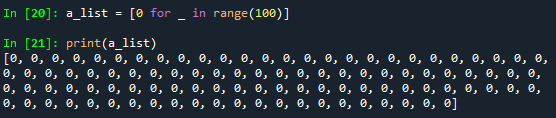
The most efficient way to set a list of 100 integers to 0 in Python is with the repetition operator (\*) or with a list comprehension.

Example:

With repetition operator:

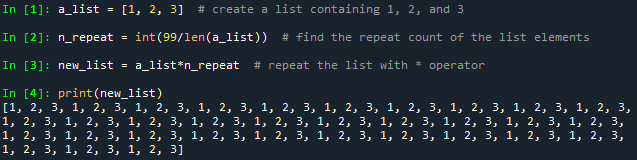
****

**With list comprehension:**

****

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.

Answer:



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

Answer:

# Method-1

a = [[2, 4, 16, 8, 100], [3, 6, 19, 112, 15], [14, 8, 12, 116, 20]]

print(a)

# Method-2

for record in a:

print(record)

# Method-3

for i in range(len(a)) :

for j in range(len(a[i])) :

print(a[i][j], end=" ")

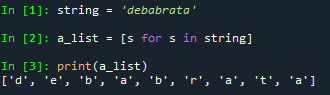
print()

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

Answer:

List comprehension with string is possible as string is an iterable object.

Example:



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

Answer:

Get support with a user-written Python programme from the command line:

1. Start Windows command prompt or terminal of Linux/Mac operating system

2. Go to the directory containing python program file (.py)

3. Type "python <file name>.py"; press enter to run the file

Get support with a user-written Python programme from inside Python IDLE:

1. Open Python IDLE.

2. Click on file, then go to Open and then select the file with \*.py extension that we would like to open

3. Press F5 key to run the program

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++?

Answer:

In Python, a function is an instance of an object type. Functions can be used as an arguments to other functions. Functions can be stored in variables. It can pass as a parameter to another function. It can return from other functions. It can store in data structures such as hash tables or lists.

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

Answer:

Wrappers for functions are called Decrators. Wrappers, wrapper functions, and decorators are all the same. They can be thought of as functions that modify the functionality of another function. They help make our code shorter and more "Pythonic".

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

Answer:

A generator function is a special kind of function that returns a lazy iterator. These are objects that can be iterated over like lists. However, unlike lists, deferred iterators do not store their contents in memory. If the function is a generator function, then instead of returning a value, it returns an iterator object that can be iterated over to get the value.

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?

Answer:

A generator is a function written as a regular function, but using the yield keyword instead of the return keyword to return a value. Hence, we can use the keyword 'yield' instead of 'return' to make our function a generator.

Q12. Identify at least one benefit of generators.

Answer:

A return statement sends a specific value back to the caller, while a yield statement can produce a set of values. If we want to iterate over a sequence, but don't want to store the entire sequence in memory, we should use a generator. Another advantage is that instead of calculating a series of values ​​in advance, the generator will calculate one value and pause its activity and wait for the next instruction. It is also memory efficient.