1.What is the relationship between def statements and lambda expressions ?

A. def statements are used to define named functions with multiple statements and can be called anywhere in the program, whereas lambda expressions are used to define small and simple anonymous functions that are usually used only once as arguments to higher-order functions.

2. What is the benefit of lambda?

A.Lambda expression in python include their concise syntax, support for anonymous functions, support for functional programming, improved code readability and flexibility.

3. Compare and contrast map, filter, and reduce.

A. map, filter and reduce are three built-in functions in python that are commonly used in functional programming to process sequences of data. Map is used to apply a function to each element of a sequence and return a new sequence with the results, filter is used to filter a sequence based on a condition and return a new sequence with filtered elements and reduce is used to apply a function to the elements of a sequence in a cumulative way and return a single result.

4. What are function annotations, and how are they used?

A. function annotations are a feature in python3 that allow you to add metadata to function arguments and return values. Annotations are optional, but they can be useful documentation for other developers and code analysis tools. Annotations can also be used with variable annotations in module-level code, class-level code and local variable assignments.

5. What are recursive functions, and how are they used?

A.A recursive function is a function that calls itself during its execution. Recursive functions are used to solve problems that can be broken down into smaller, simpler versions of the same problem.

Recursive functions can be useful for solving problems that have a recursive structure, such as tree traversal or sorting algorithms like quicksort and merge sort. However, they can be less efficient than non-recursive solutions, as each recursive call adds a new layer to the call stack and requires additional memory.

Its also important to be careful when using recursive functions as its easy to accidentally create an infinite loop if the base case is not defined correctly or the recursion is not properly constrained.

6. What are some general design guidelines for coding functions?

A. Below are some general design guidelines for coding functions

a. Keep your functions short and focused

b. Use clear and descriptive names for your functions.

c. Write functions that are reusable

d. Use default arguments and keyword arguments sparingly.

e. Avoid side effects,

f. Write docstrings for your functions,

g. write test cases for your functions,

h. Avoid deeply nested code ,

i. Don’t repeat yourself(DRY),

j. Use comments sparingly.

7. Name three or more ways that functions can communicate results to a caller.

A. Return values,

Global variables,

Out parameters,

exceptions,

call backs,

yield,

print statements