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

Both lambda and def create the same kind of function – they have the same kind of metadata and capabilities. Their technical difference is syntactical: A lambda is an expression producing a function. A def is a statement producing a function.

1. What is the benefit of lambda?

Lambda functions allow you to create small, single-use functions that can save time and space in your code. They ares also useful when you need to call a function that expects a function as an argument for a callback such as Map() and Filter()

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

MAP function to each element of iterator and collects result.

FILTER applies function to each element of iterator and collect those elements for which function returns true. For this function need to return boolean value.

REDUCE applies rolling computation to sequential pair of elements in iterator.

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

Function annotations are arbitrary python expressions that are associated with various part of functions. These expressions are evaluated at compile time and have no life in python's runtime environment.

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

A recursive function is a function in code that refers to itself for execution.

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

* Coding guidelines increase the efficiency of the software and reduces the development time.
* Coding guidelines help in detecting errors in the early phases, so it helps to reduce the extra cost incurred by the software project.
* If coding guidelines are maintained properly, then the software code increases readability and understandability thus it reduces the complexity of the code.
* It reduces the hidden cost for developing the software.

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