1) . What is the difference between enclosing a list comprehension in square brackets and

parentheses?

ANS The square brackets tell Python that this is a list comprehension, producing a list. If you use curly braces, you'll get either a set or a dict back, and if you use regular parentheses, you'll get a generator expression

2) What is the relationship between generators and iterators?

ANS IteratorGeneratorIterators are used mostly to iterate or convert other objects to an iterator using iter()

function.Generators are mostly used in loops to generate an iterator by returning all the values in the loop without affecting the iteration of the loop

3) What are the signs that a function is a generator function?

ANS A function with yield statements is a generator functionMeaning, when you call a generator function, it doesn't run the function. Instead, it gives you back a generator object. If you loop over that generator object, it will run the function until a yield statement is reached

4) What is the purpose of a yield statement?

ANS The Yield keyword in Python is similar to a return statement used for returning values or objects in Python. However, there is a slight difference. The yield statement returns a generator object to the one who calls the function which contains yield, instead of simply returning a value

5) What is the relationship between map calls and list comprehensions? Make a comparison and

contrast between the two.

ANS List comprehension is more concise and easier to read as compared to map. List comprehension are used when a list of results is required as map only returns a map object and does not return any list. Map is faster in case of calling an already defined function (as no lambda is required).

FINISH\*\*\*\*