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

**Ans: When we enclose it with square brackets, this is called list comprehension and returns a list object. However, when we enclose it with parentheses, this is called Generator comprehension and returns a generator object.**

2) What is the relationship between generators and iterators?

**Ans:**

**Iterator is an object that can be iterated over. It’s next() and iter()functions can be overridden. Generators provide us means to create iterators in an easier manner without creating a new iterator class. Instead of that we can define a generator function that uses yield keyword. The state of the local variables in iterator have to be maintained explicitly, however this doesn’t need to be worried in generators.**

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

**Ans: A generator function must have yield keyword instead of a return keyword.**

4) What is the purpose of a yield statement?

**Ans: Yield is a keyword in Python that returns the execution back to the calling function without destroying the state of its local variables. Yield keyword is used in generators.**

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

**Ans: Map function allows us to apply a function to a set of values and return a map object which is an iterator. On the other side, list comprehension allows us to create a list by applying some operations.**

**In list comprehension we can put some conditions to extract data but the same can’t be done with map function.**

**The operation being performed is visible in list comprehension but now visible in map function unless we open the function definition.**