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

ANS) Square brackets are lists while parentheses are tuples.

**2) What is the relationship between generators and iterators?**

ANS) An iterator does not make use of local variables, all it needs is iterable to iterate on. A generator may have any number of 'yield' statements. You can implement your own iterator using a python class; a generator does not need a class in python.

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

ANS) If a function contains at least one yield statement (it may contain other yield or return statements), it becomes a generator function. Both yield and return will return some value from a function.

**4) What is the purpose of a yield statement?**

ANS) The yield keyword is used to create a generator function. Generator functions act as iterators which can be looped through with a foreach loop. The value given by the yield keyword is used as a value in one of the iterations of the loop.

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

ANS) Map VS List Comprehension

* List comprehension is more concise and easier to read as compared to map.
* List comprehension allows filtering. In map, we have no such facility. For example, to print all even numbers in range of 100, we can write [n for n in range(100) if n%2 == 0]. There is no alternate for it in 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.
* List comprehension is faster than map when we need to evaluate expressions that are too long or complicated to express
* Map is faster in case of calling an already defined function (as no lambda is required).