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

**Ans.** Enclosing a list comprehension in square brackets returns a list where as enclosing a list comprehension in parentheses returns a generator object.

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

**Ans.** An iterator is an object which contains a countable number of values and it is used to iterate over iterable objects like list, tuples etc.

Using an iterator- iter() keyword is used to create an iterator containing an iterable object. next() keyword is used to call the next element in the iterable object.

Generator is a way of creating iterators in a simple way where it uses the keyword "yield" instead of returning it in a defined function. Generators are implemented using a function. Here, the yield function returns the data without affecting or exiting the function. It will return a sequence of data in an iterable format where we need to iterate over the sequence to use the data as they won’t store the entire sequence in the memory By using next() function we can iterate the output of generator function.

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

**Ans.** If a function contains yield statement then it becomes a generator function.

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

**Ans.**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.** The main differences between map calls and list comprehensiosn are:

\*List comprehension is more concise and easy to read as compared to map.

\* List comprehension allows filtering. In map, we have no such facility. For example, to print all odd numbers in range of 50, we can write [n for n in range(50) if n%2 != 0]. There is no alternate for it in map.

\* List comprehension are used when a list of results is required as final output but map only returns a map object. It needs to be explicitly coverted to desired datatype.

\* 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 on a set of values.