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

**Answer:**

**Enclosing a list comprehension in square brackets will return us a List , whereas enclosing a list comprehension in parentheses will return us a generator object.**

2) What is the relationship between generators and iterators?

**Answer:**

### ****Iterator :****

**Iterator is an object which contains a countable number of values and it is used to iterate over iterable objects like list, tuples, sets, etc. Iterators are implemented using a class and a local variable for iterating is not required here, It follows lazy evaluation where the evaluation of the expression will be on hold and stored in the memory until the item is called specifically which helps us to avoid repeated evaluation. We use iter() or next() for using iterator.**

**e.g**

**x = iter([i for i in range(10) ])**

**next(x)**

### ****Generators****

**It is another 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. Just as iterators, generators also follow lazy evaluation. 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.**

**e.g,**

**def squre\_num(n):**

**for i in range(1, n+1):**

**yield i\*i**

**a = squre\_num(3) #generator object**

**print(a)**

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

**Answer:**

**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?

**Answer:**

**A yield statement looks much like a return statement, except that instead of stopping execution of the function and returning, yield instead provides a value to the code looping over the generator and pauses execution of the generator function.**

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

**Answer:**

* **List comprehension is more concise and easier to read as compared to map.**
* **List comprehension allows filtering. In map, we have no such option.**
* **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).**
* **We need to define a separate function in case of map.**