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

**ANS**

Square brackets are [lists](http://docs.python.org/tutorial/datastructures.html#more-on-lists) while parentheses are [tuples](https://docs.python.org/tutorial/datastructures.html#tuples-and-sequences).

A list is mutable, meaning you can change its contents:

while tuples are not

The other main difference is that a tuple is hashable, meaning that you can use it as a key to a dictionary, among other things.

2) What is the relationship between generators and iterators?

**ANS**

A python generator is an iterator  
Generator in python is a subclass of Iterator. To prove this, we use the issubclass() function.

Python iterator is an iterable  
Iterator in python is a subclass of Iterable.

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

**ANS**

It is as easy as defining a normal function, but with a yield statement instead of a return statement. 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**

In its simplest form, 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.

**ANS**

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).