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

Ans : The difference between the two kinds of expressions is that the List comprehension is enclosed in square brackets [] while the Generator expression is enclosed in plain parentheses ()

1. What is the relationship between generators and iterators?

Ans :

|  |  |
| --- | --- |
| Iterator | Generator |
| Iterators are used mostly to iterate or convert other objects to an iterator using iter() function. | Generators are mostly used in loops to generate an iterator by returning all the values in the loop without affecting the iteration of the loop |

1. What are the signs that a function is a generator function?

Ans : In a generator function, a yield statement is used rather than a return statement.

1. What is the purpose of a yield statement?

Ans : Yield is a keyword in Python that is used to return from a function without destroying the states of its local variable and when the function is called, the execution starts from the last yield statement. Any function that contains a yield keyword is termed as generator. Hence, yield is what makes a generator. yield keyword in Python is less known off but has a greater utility which one can think of.

1. 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).