

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 ()

2) What is the relationship between generators and iterators?

Ans: 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.

Iterators : An 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. As lazy evaluation is implemented, it requires only 1 memory location to process the value and when we are using a large dataset then, wastage of RAM space will be reduced the need to load the entire dataset at the same time will not be there.

3) 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.

4) 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.

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).