

Assignment : Functions Assignment

1. What is the difference between a function and a method in Python?

Ans- Function: Independent piece of code that can be called anywhere in the program. Defined using def keyword.

Method: A function associated with an object. It's called an instance of a class. Defined inside a class.

2. Explain the concept of function arguments and parameters in Python?

Ans- Parameters are variables listed in a function's definition.

Ex- `def numbers(a,b)`

Arguments are values passed to the function when it's called.

Ex- `res = numbers(5,6)`

3. What are the different ways to define and call a function in Python?

Ans- Standard Function, Lambda Function, Default Arguments, Keyword Arguments.

4. What is the purpose of the `return` statement in a Python function?

Ans- The return statement in a Python function is used to send a value back to where the function was called from and to stop the function's execution.

5. What are iterators in Python and how do they differ from iterables?

Ans- Iterators: Objects that can be iterated upon (traversed through one element at a time) using the `next()` function. They implement `__iter__()` and `__next__()` method.

Iterables: Objects that can return an iterator using the `iter()` function. Examples include lists, tuples, and strings.

6. Explain the concept of generators in Python and how they are defined.

Ans- Special types of iterators that allow you to iterate through a sequence of values lazily, one at a time, without storing the entire sequence in memory. They are created using the `yield` keyword.

7. What are the advantages of using generators over regular functions?

Ans- Memory Efficiency: Generate values on-the-fly without storing the entire sequence in memory.

Lazy Evaluation: Produce items only when needed, which can improve performance for large datasets.

8. What is a lambda function in Python and when is it typically used?

Ans- A lambda function in Python is an anonymous (unnamed) function defined using the `lambda` keyword. It can have any number of arguments but only a single expression.

9. Explain the purpose and usage of the `map()` function in Python.

Ans- The Map function in Python applies a given function to each item of an iterable (like a list) and returns a map object (an iterator) with the result. Purpose Transform each element in an iterable according to a specified function.

10. What is the difference between `map()`, `reduce()`, and `filter()` functions in Python?

Ans- `map()`: transforms each item in an iterable using a given function.

`Reduce`: Reduces an iterable to a single value by applying a function cumulatively.

`Filter`: Selects items from an iterable that meet a certain condition.

11. Using pen & Paper write the internal mechanism for sum operation using reduce function on this given list: [47, 11, 42, 13];
Ans-

//_

Describe the internal mechanism of the reduce function of given list [47, 11, 42, 13]

1) initial step:-
input list [47, 11, 42, 13]

operation:- sum element of the custom function
 $\lambda x, y: x+y$

2) reduce:-
Start with first two element in the given list
for function applied $47+11=58$
Next step between previous result and third element of the list
 $58+42=100$
Final combine result with previous result and fourth element of the list
 $100+13=113$ (Final result)