Fundamentals of Python Programming

Homework Assignments

Module: Generators. Higher order functions. Decorators. Modules. API testing

Part 2

Task 1

Create a function that returns all Fibonacci numbers in a range. The function takes the beginning and the end of the range as parameters. Use the generator mechanism within the function.

Fibonacci numbers are a sequence of numbers where each successive number is the sum of the previous two numbers. The sequence begins with 0 and 1, and the next numbers are determined by the formula:

F(0) = 0

F(1) = 1

F(n) = F(n-1) + F(n-2), for n > 1.

The first few Fibonacci numbers are: 0, 1, 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, ...

You can read more about the Fibonacci numbers.

Task 2

Create a function that returns the sum of the values of lists elements. The function accepts two lists. Use the generator mechanism inside the function.

For example, if we have the following lists:

1 3 4 2

8 3 5 9

The result would be:

9 6 9 11

If the passed lists have different lengths the missing elements must be considered equal to zero.

Task 3

To solve this task, be sure to use the mechanism of higher order functions. Create a function that squares or cubes the list values. A user determines what to do (square or cube).

The function's signature is:

```python
def calculate(list_to_work, function_to_call)
```

- list_to_work — a list with elements.
- function_to_call — a function to modify values (function to square or cube the values).

Task 4

Every year your company submits financial reports to different government agencies. Depending on the organization, the reporting formats are different. Using the decorator mechanism, solve the reporting issue for the organizations.