

## Computational Finance and FinTech – Exercises 2

**Exercise 1.** Write a program that finds all numbers that are divisible by 7 but are not multiples of 5 between 2000 and 3200 (both included). The numbers obtained should be printed in a comma-separated sequence on a single line.

**Exercise 2.** Write a function that computes the present value of a growing annuity:

$$PV = \frac{C}{r - g} \left( 1 - \frac{(1 + g)^n}{(1 + r)^n} \right),$$

where  $r$  is the discount rate,  $g$  is the growth rate,  $C$  is the amount paid and  $n$  is the number of periods (years).

Use the function to calculate the PV of an annuity that pays 100 annually for ten years using a discount rate of 5%.

**Exercise 3.** Using the function from the previous exercise, build a mortgage calculator that prints out the monthly mortgage payment given an annual percentage rate (note:  $r = \text{APR}/12$ ), the loan amount and the number of years (note:  $n = \text{years} \cdot 12$ ).

Use the function to calculate the monthly payment on a loan of €100,000 at an APR of 2.5%. Verify that the PV is indeed €100,000 by calculating the sum of the discounted loan payments.

**Exercise 4.** Write a Python program to combine two dictionaries adding values for common keys.

Example: Let

```
d1 = {'a': 100, 'b': 200, 'c': 300}
d2 = {'a': 300, 'b': 200, 'd': 400}
```

Sample output:

```
{'a': 400, 'b': 400, 'd': 400, 'c': 300}
```

**Exercise 5.** Generate a random walk path starting in 0 by accumulating normally distributed random numbers. Use a **Data Frame** object indexed by days (365 days) to accommodate the path. Plot the result.

Extend the **DataFrame** of random numbers to accommodate more paths and fill it with 5 paths. Find out what the seed of a random number generator is and how you can set the seed in Python. Why would this be useful?

Hint: Use `numpy.random`.