

# Assignment II

## Programming with Python

### Introduction

This is the second of the two assignments for this course. It will be graded as pass/fail and you and your group of up to 3 students will need to submit it before the start of the last lecture. Please hand in one assignment per group via the e-mail address [vlcek@beyondsimulations.com](mailto:vlcek@beyondsimulations.com).

In the assignment, you will practice the concepts you learned in the second part of the course. You will find a lot of information online and you are welcome to use generative AI to help you with the assignment. However, you are not allowed to copy the code from other groups and you have to indicate where and how you used AI to help you. Try to use comments to structure and to explain your code. Furthermore, use descriptive variable names and format your code nicely in order to make it more readable.

### Redact secret information

In this exercise, you will create a program that is able to redact secret information in a text. The program should be able to redact the following: zip codes, names, email addresses and phone numbers. The program should ask the user for a filename and then read the file and redact the secret information. The program should then print the redacted text to the console and write it to a new file called `redacted.txt`. You can find a file with secret information in the git repository under `assignments/secret-text.txt`.

```
# Secret information redactor
# TODO: Create a program that is able to redact secret information in a
text.
# YOUR CODE HERE
```

### Dice roll simulator

In this exercise, you will create a program that is able to simulate dice rolls and visualizes their distribution. The program should ask the user for the number of sides on the dice and the number of dices to roll. Then, the program should simulate the dice rolls 10000 times and visualize the distribution of the dice rolls using a histogram.

```
# Dice roll simulator
# TODO: Create a program that is able to simulate a dice roll.
# YOUR CODE HERE
```

## Future Self Predictor

In this exercise, you will create an interactive program that predicts a users future based on his or her answers. The program should ask 5 questions about users where user can either add the answers in the terminal freely or choose from a number of options. It should then generate 3 entertaining predictions, and save them to a separate file with the user name as filename. The predictions should be a 5-year, 10-year, and 30-year prediction of where the user is going to be in life. Note, the program should be able to handle user input errors gracefully.

```
# Future Self Predictor
# TODO: Create a program that is able to predict the future.
# YOUR CODE HERE
```

## Dashboards

In this exercise, you will create a program that visualizes a data set of your choice in an interactive dashboard. For this exercise, you can choose any data set of your interest. The data should be visualized in a dashboard with at least two plots. You can use one of the dashboard libraries we discussed in the lecture.

### Note

The dashboard should be programmed in a separate file I can call for evaluation.

```
# Dashboards
# TODO: Create a program that visualizes a data set of your choice in a
dashboard.
# YOUR CODE HERE
```