

# Assignment I

## Programming with Python

### Introduction

This is the first of the two assignments for this course. It will be graded as pass/fail. You and your group of up to 3 students will need to submit it before the start of lecture 7 on scientific programming. 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 first 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.

### Tax refund for donations

In this exercise, you will create a program that is able to calculate the amount of tax a person can get back for a donation. First, research the rules for tax refunds for donations in Germany. Then, create a program that is able to calculate the amount of tax a person can get back for a donation. The program should ask the user for the amount of the donation and the yearly income and then calculate the amount of tax the user can get back. The program should then print the amount of tax to the console. Note, that the program should also handle wrong inputs from the user, e.g. if the user enters a negative donation amount or a non-numeric income.

```
# Tax refund for donations
# TODO: Create a program that is able to calculate the amount of tax a
# person can get back for a donation.
# Your code here
```

#### Tip

Use the `input()` function to get user input in order to let the user interact with your program.

### Text based adventure game

In this exercise, you will create a text based adventure game. You are completely free in the choice of the story, but it must at least include 4 choices, 3 different endings, the choice to restart the game and yourself as a character of the game. Try to use the

concepts of conditionals, loops, and functions to create a game that is both fun and interactive for the user. Note, that the program should also handle wrong inputs from the user, e.g. if the user enters a choice that is not available.

```
# Test based adventure game
# TODO: Create a text based adventure game with at least 3 choices, 2
different endings and the choice to restart the game.
# Your code here
```

## Hangman

In this exercise, you will create a program that is able to play the game Hangman. The program should ask the game master for a secret word. Afterward, it should ask the player to guess a letter and then check if the letter is in the word. The program should then print the word with the guessed letters and the number of tries left. The program should also handle wrong inputs from the user, e.g. if the user enters a non-letter or a letter that is already guessed.

```
# Hangman
# TODO: Create a program that is able to play the game Hangman with a game
master and one player.
# Your code here
```

## Caesar Cipher

In this exercise, you will create a program that is able to encrypt and decrypt messages using the Caesar Cipher. The program should ask the user for a message and a key and whether to encrypt or decrypt the message. The program should then encrypt the message by shifting each letter of the message by the key or decrypt the message by shifting each letter of the message back by the key. The program should then print the encrypted or decrypted message and asks the user whether to continue with the next message or to quit the program. Note, that you only need to encrypt or decrypt letters, keep other characters unchanged.

```
# TODO: Create a program able to encrypt and decrypt messages using the
Caesar Cipher.
# Your code here
```