**Python Programming for non-Programmers**



Saarland University

**Winter Term 2020/2021**

M.Sc. Alexander Berrang | M.Sc. Martin Scheid | B. Sc. Jessica Lackas

**Project**

**Language Learning App in Command Line**

|  |  |
| --- | --- |
| Issue: | 14.12.2020, 10:00 am |
| Submission deadline: | 28.02.2021, 11:59 pm |
|  |  |

**Project Information:**

For your final Project you will be programming a German language learning app in Python Command Line. Don't worry if you don't speak German. We will provide a list of words and translations your app needs to support. You will work in groups of 2 or 3 people. For a passing grade you only need to implement the basic functionality of the app. For a higher grade you also need to implement the advanced functionality. The grading criteria are explained below.

**Data:**

A file with 100 words from the categories nouns, verbs, and adjectives your app needs to support has been provided to you.

**Tasks:**

Your app is supposed to have 4 different modes, each of which is explained separately below. You also need to implement a way of switching between the different modes. The app needs to be self-explanatory and guide the user through how to use it and what the functionality of the app is. The instructions are under specified on purpose, so that you can come up with creative ideas. We expect each group to have a unique solution, so do not share your code with other groups.

**Mode 1: Dictionary**

Basic: The first mode is very straight forward. The user either types in a German word and the app displays the English translation or the other way around. For this your program needs to be able to read an external file (the one we provide) with the words and translations in it.

**Mode 2: Vocabulary Trainer**

Basic: In this mode your app emulates a traditional vocabulary trainer using the same file as in the dictionary mode. This means the app provides a German word and the user needs to type in the English translation or the other way around. The app then checks whether the input is correct and corrects it if it's not.

Advanced: Store the user's progress in another external file. Words that the user already knows are supposed to be selected less often. Words that are newly added to the external file are supposed to be selected more often.

Advanced: Give the user the option to print out some analytics regarding their progress, meaning how many times they have used the vocab trainer, how many words they’ve already gotten right multiple times, how many words they keep getting wrong, and any other useful information.

**Mode 3: Grammar**

Basic: German grammar is complex, so this mode only checks a very small fragment of it, namely the conjugation of the verb "to be", which is the verb "sein" in German.

I am - ich bin

you are - du bist

he/she/it is - er/sie/es ist

we are- wir sind

you are - ihr seid

they are - sie sind

The user inputs a German text that consists of several sentences only using pronouns and the verb "to be". For example:

Er ist groß. Sie ist klein. Wir sind freundlich. Du bist toll.

(He is tall. She is short. We are friendly. You are great.)

Your program needs to check whether the pronoun matches the conjugation of "sein" in every sentence.

**Mode 4: Mad Libs Game**

Basic: This mode is not really about language learning, rather it‘s a fun addition to your app. Mad Libs is a phrasal template word game. Here is one example for a phrasal template, but you can also come up with your own using (singular) nouns, adjectives, and verbs.

A unicorn is nothing like a \_\_\_\_\_\_(noun). They're \_\_\_\_\_\_(adjective) creatures. Some have a \_\_\_\_\_\_(adjective) mane of hair and others have a \_\_\_\_\_\_(adjective) \_\_\_\_\_\_(noun) on their head. I would love to \_\_\_\_\_\_(verb) a unicorn one day.

This mode does not require any user input. Your program is supposed to print out the mad lib filling in the blanks with random words from the dictionary mode. Make sure to use the correct part of speech (noun, verb, adjective).

Advanced: If possible, only select words the user already knows from the vocabulary trainer. If the user hasn't used the vocab trainer yet or doesn't know enough words to fill the blanks use random words.

**Grading**

You will be graded based on the following criteria:

- Does your app work? How many of the modes are usable?

- Is your app user friendly? (To the degree that's possible in command line)

- Is your code efficient and elegant? Is it modular and easy to change and extend?

- Is your code documented in a way that makes it easy to read? (This means putting comments into your code)