

**INT 344**: Natural Language Processing

**Submitted by:**

**Shubham Jain: 11701185**

**Jilvin Thomas Abraham: 11704962**

**CA 3: PROJECT**

**TOPIC:** SPELL CHECKER

**Submitted To:**

**Ishaan Sir**

**School of Computer Science &**

**Engineering Lovely Professional**

**University, Phagwara**

1. **INTRODUCTION**

Spell checker has many applications in the real world. The most common tool that we use known as the Microsoft Word has an inbuilt spell checker in it whenever you type an incorrect spelling of any word the application suggest the correct word. In this project we have created a simple spell checker application we have used the levenshtein algorithm to suggest the similar words the application is built with python, tkinter module is used for the GUI part what the application does is very simple you input a word if the spelling is correct it returns the meaning of the word if it is misspelled it suggests similar words. We have many algorithms for suggesting similar words in the project we have used edit minimum distance technique and have kept a threshold of 1 that is if the algorithms returns a value less than or equal to 1 the word is taken for suggestion and considered similar. Also if there is a need to add new word to the dictionary that can also be done by the adding to dictionary feature that we have provided.

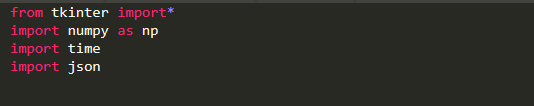
**Requirements**:

A dictionary full of words as keys and meanings as values in Jason format.

A python editor

1. **CODE IMPLEMENTATION**

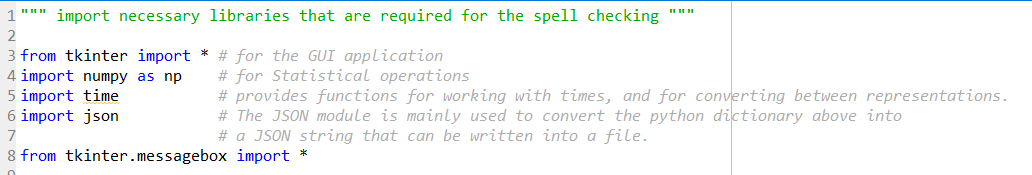
Required libraries to be imported



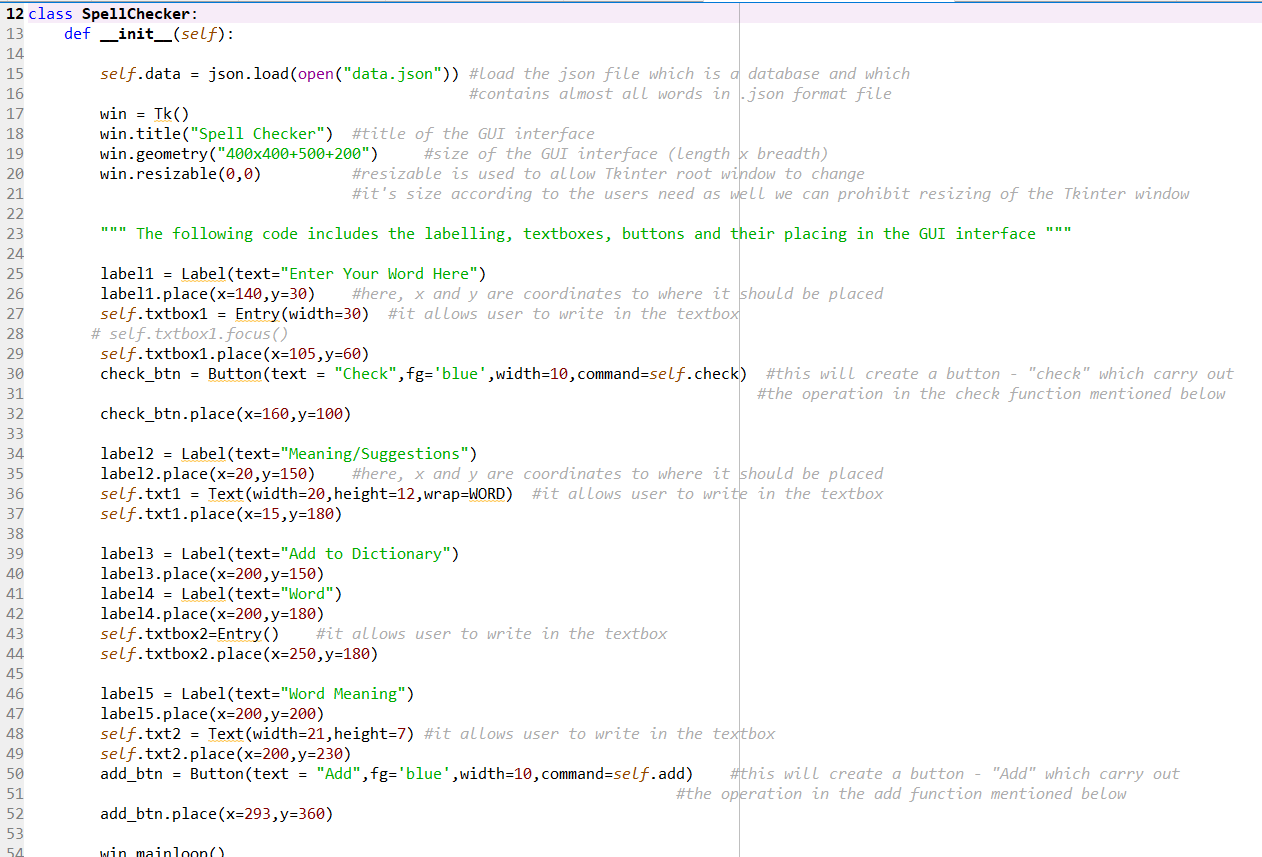
1. Tkinter library: Tkinter is the standard GUI library for Python. Python when combined with Tkinter provides a fast and easy way to create GUI applications. Tkinter provides a powerful object-oriented interface to the Tk GUI toolkit.
2. Numpy library: NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays. **NumPy** is a **Python** package which stands for 'Numerical **Python**'. It is the core **library** for scientific computing, which contains a powerful n-dimensional array object, provide tools for integrating C, C++ etc. It is also useful in linear algebra, random number capability etc.
3. Time library: There is a popular **time** module available in Python which provides functions for working with times, and for converting between representations. The function *time.time()* returns the current system time in ticks since 12:00am, January 1, 1970(epoch).
4. JSON library: Use the **import** function to **import** the **JSON module**. The **JSON module** is mainly used to convert the **python** dictionary above into a **JSON** string that can be written into a file. While the **JSON module** will convert strings to **Python** datatypes, normally the **JSON** functions are used to read and write directly from **JSON** files.

**Explanation of Code using Comment Lines:**

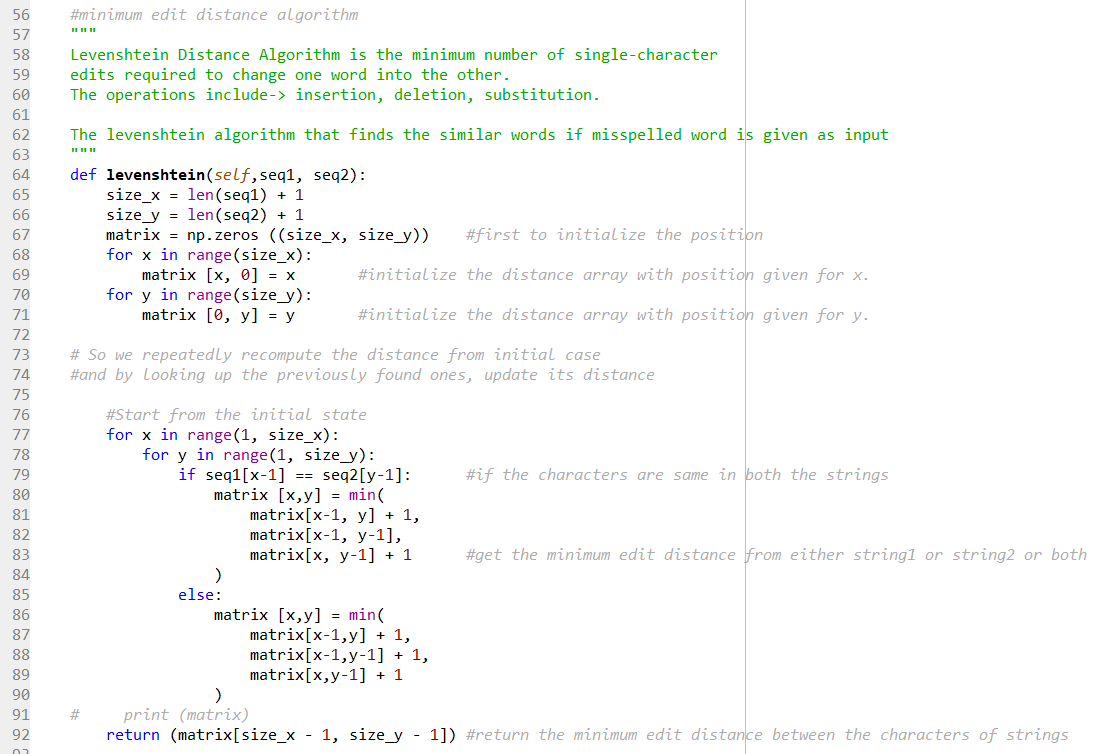
1. Necessary Libraries:



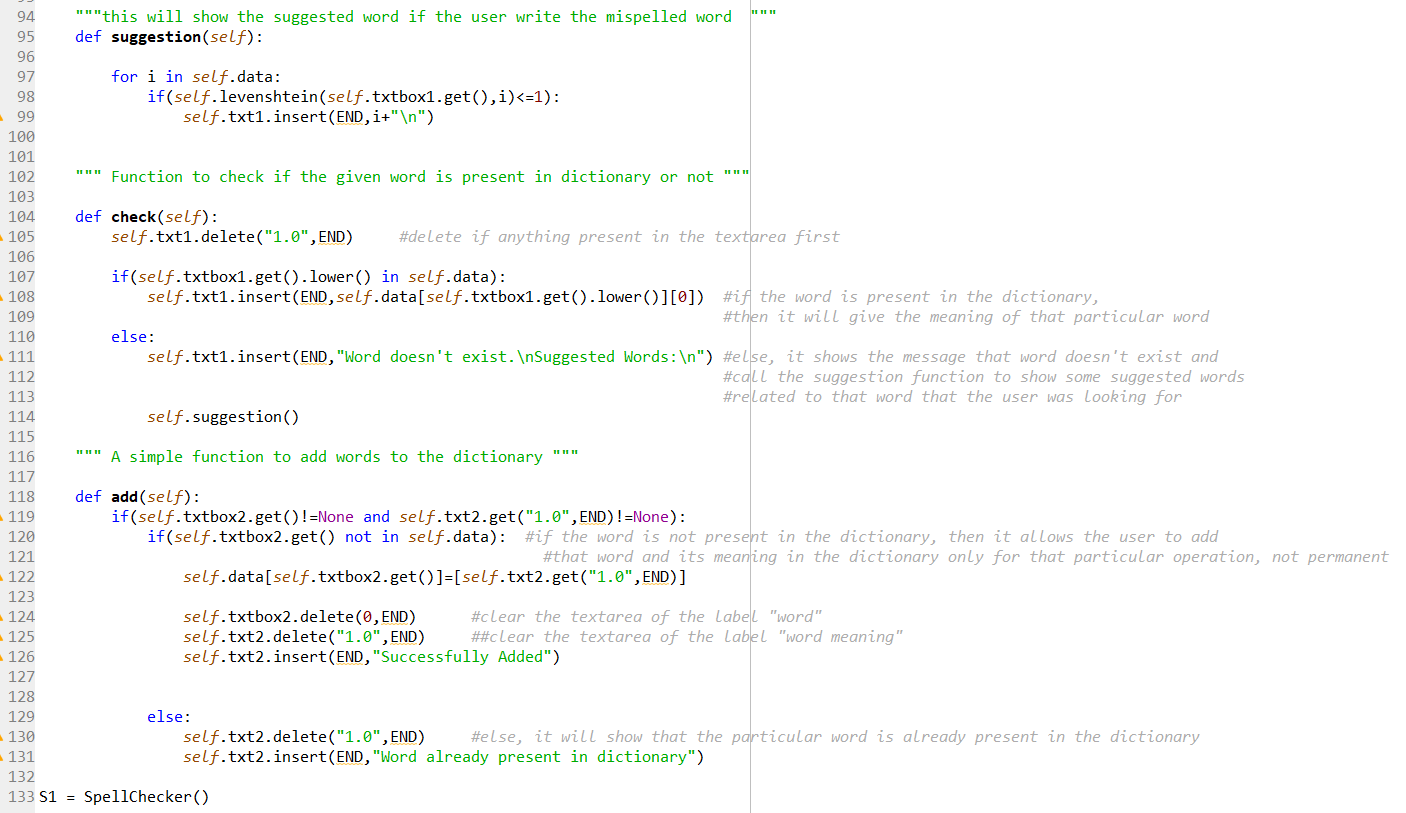
1. GUI Interface:



1. Levenshtein Minimum Distance Algorithm:

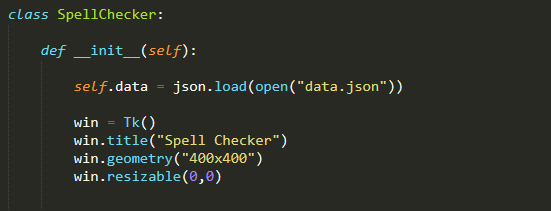


1. Function related to GUI like Suggestion, Check and Add Functions:



**Explanation of Each Module:**

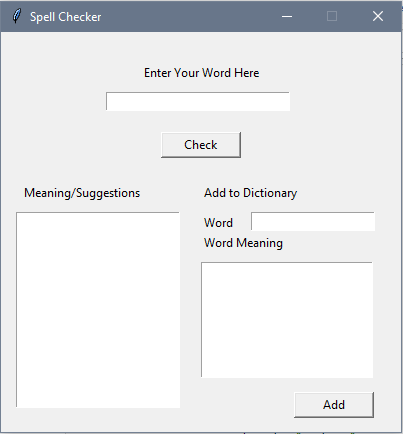
A class implementation SpellChecker, loading the dictionary from Jason format .



The design of the application is coded in the following manner

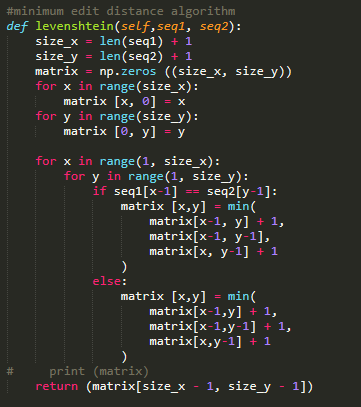


1. **Graphical user interface of the application:**

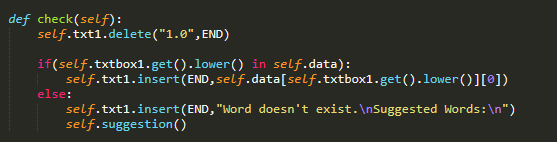


1. **Explanation of Levenshtein Algorithm:**

i. The levenshtein algorithm that finds the similar words if misspelled word is given as input:



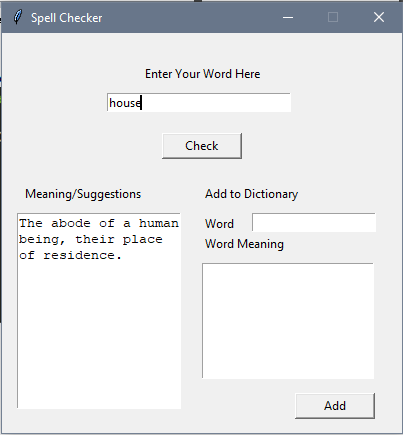
**ii. Function to check if the given word is present in dictionary or not**



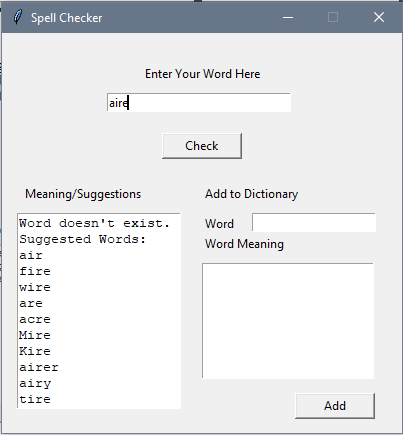
**iii. A simple function to add words to the dictionary**



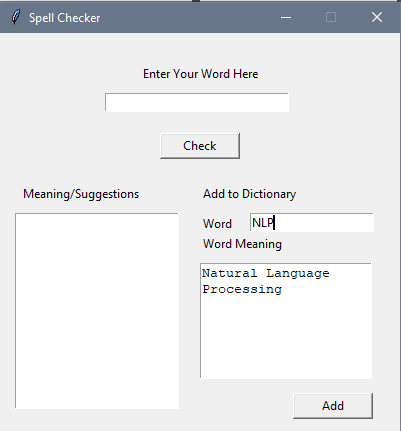
1. **RESULT VISUALIZATION**
2. When given the correct word the meaning of the word is displayed



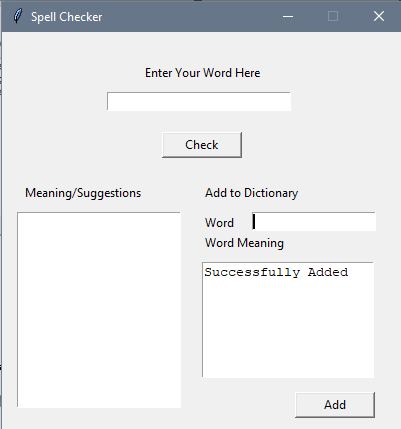
1. When wrong or incorrect word is given suggestions are shown



1. When you want to add a word to the dictionary fill the fields and click add



Result:



1. **GITHUB REPOSITORY LINKS:**

**SHUBHAM JAIN:** <https://github.com/Shubham11Jain/Spell-Checker>

**JILVIN THOMAS ABRAHAM :**