

```
In [1]: ▶ from textblob import TextBlob #for spelling correction
import re #for spelling correction
import string #for punctuation correction
from language_tool_python import LanguageTool #for grammar correction
from tkinter import * #for GUI
```

Spelling Correction Function ¶

```
In [2]: ▶ def correct_spelling(sentence):
    # Tokenize the sentence into words
    words = sentence.split()

    corrected_words = []
    for word in words:
        # Preserve the case of the first letter using regular expressions
        if re.match(r'[A-Z]', word): # Check if the word starts with an uppercase letter
            corrected_word = str(TextBlob(word.lower()).correct()).capitalize()
        else:
            corrected_word = str(TextBlob(word).correct())

        corrected_words.append(corrected_word)

    return ' '.join(corrected_words)
```

Punctuation Correction Function

```
In [3]: ▶ def correct_punctuation(sentence):
    punctuation_marks = set(string.punctuation) - {'"'}

    corrected_sentence = []
    words = sentence.split()

    for word in words:
        if word[-1] in punctuation_marks:
            if len(word) > 1 and word[-2] not in string.whitespace:
                corrected_sentence.append(word[:-1] + " " + word[-1])
            else:
                corrected_sentence.append(word)
        else:
            corrected_sentence.append(word)

    return ' '.join(corrected_sentence)
```

Grammar Correction Function

```
In [4]: ▶ def correct_grammar(sentence):  
        tool = LanguageTool('en-US')  
        matches = tool.check(sentence)  
        return tool.correct(sentence), matches
```

Auto Correct Function

```
In [5]: ▶ def auto_correct_with_mistakes(sentence):  
        corrected_spelling = correct_spelling(sentence)  
        corrected_punctuation = correct_punctuation(corrected_spelling)  
        corrected_grammar, grammar_mistakes = correct_grammar(corrected_punctu  
        mistakes = {  
            "Spelling": corrected_spelling != sentence,  
            "Punctuation": corrected_punctuation != corrected_spelling,  
            "Grammar": grammar_mistakes  
        }  
        return corrected_grammar, mistakes
```



```

In [6]: ▶ from tkinter import *
from tkinter import filedialog
from docx import Document
import PyPDF2

class main:
    def __init__(self, root):
        self.root = root
        root.geometry("1400x533")
        root.maxsize(1400,533)
        root.minsize(1400,533)

        root.title("Auto Correct System")

        Header_frame = Frame(root, bg="lavender", height=150)
        title_label = Label(Header_frame, text="    Auto Correct System    "
                            font=('Garuda', 30, 'bold', 'italic'), pady=10)

        Center_frame = Frame(root, bg="medium slate blue", height=450)

        Center_block1 = Frame(Center_frame, bg="medium slate blue", width=
uploadfilebut_label = Label(Center_block1, text="Upload File for t
                            font=('Garuda', 10, 'bold', 'italic'), justify
fileupload_button = Button(Center_block1, text="Choose a File", bg
                            width=12, height=1, fg="white", activeb
                            )
writetext_label = Label(Center_block1, text="Write text for auto c
                            font=('Garuda', 10, 'bold', 'italic'), justify
writetext_Entry = Text(Center_block1, font=('Garuda',10), width=97
checktext_button = Button(Center_block1, text="Correct Text", bg=
                            width=12, height=1, fg="white", activeb

        Center_block2 = Frame(Center_frame, bg="orchid", width=690, height
correct_label = Label(Center_block2, text="Aut Corrected Text:", b
                            font=('Garuda', 10, 'bold', 'italic'), justify
correcttext_label = Label(Center_block2, bg="white", font=('Garuda
                            anchor="nw")

    def selectFile():
        file_path = filedialog.askopenfilename(initialdir='D://', titl
                            filetypes=[("Text Files", "*.txt"), ("Word
if file_path.endswith('.docx'):
    try:
        doc = Document(file_path)
        whole_text = ""
        for paragraph in doc.paragraphs:
            whole_text += paragraph.text + "\n"
        print(f"Original Sentence: {whole_text}")
        Testtext(whole_text)
    except Exception as e:
        print(f"Error reading the Word document: {e}")
elif file_path.endswith('.txt'):
    try:
        with open(file_path, 'r') as file:
            content = file.read()

```

```

        print(content)
    except Exception as e:
        print(f"Error reading the text file: {e}")
    elif file_path.endswith('.pdf'):
        try:
            with open(file_path, 'rb') as file:
                pdf_reader = PyPDF2.PdfReader(file)
                num_pages = len(pdf_reader.pages)
                for page_num in range(num_pages):
                    page = pdf_reader.getPage(page_num)
                    print(page.extract_text())
        except Exception as e:
            print(f"Error reading the PDF file: {e}")
    else:
        print("Unsupported file format. Please provide a .docx or

def Testtext(data):
    corrected_sentence, mistakes = auto_correct_with_mistakes(data)
    correcttext_label.config(text=corrected_sentence)

    print(f"Auto-corrected Sentence: {corrected_sentence} \n")

    print("\nDetected Mistakes:")
    for mistake_type, mistake_detail in mistakes.items():
        if mistake_type == "Grammar":
            print(f"Grammar Mistakes:")
            for mistake in mistake_detail:
                print(f" - {mistake}")
        else:
            print(f"{mistake_type} mistake: {mistake_detail}")

def gettext():
    sent = writetext_Entry.get("1.0", "end-1c")

    Testtext(sent)

fileupload_button['command'] = selectFile
checktext_button['command'] = gettext

Header_frame.pack(side=TOP, fill=BOTH)
title_label.pack(pady=15)

Center_frame.pack(side=TOP, fill=BOTH)

Center_block1.pack(side=LEFT, fill=BOTH)
uploadfilebut_label.pack(side=TOP, pady=(20,0), padx=10, anchor="w")
fileupload_button.pack(side=TOP, pady=10, padx=10, anchor="w")
writetext_label.pack(side=TOP, pady=(10,0), padx=10, anchor="w")
writetext_Entry.pack(side=TOP, padx=10, pady=5, expand=1)
writetext_Entry.config(height=15)
checktext_button.pack(side=TOP, pady=15, padx=10, anchor="w")

Center_block2.pack(side=LEFT, fill=BOTH, padx=(10,0))
correct_label.pack(side=TOP, pady=(20,0), padx=20, anchor="w")

```

```

correcttext_label.pack(side=TOP, padx=20, pady=10)
correct_label.pack(side=TOP, pady=(20,0), padx=20, anchor="w")

root = Tk()
obj = main(root)
mainloop()

```

Downloading LanguageTool 5.7: 100%|██████████| 225M/225M [04:07<00:00, 907kB/s]

Unzipping C:\Users\ABDULL~1\AppData\Local\Temp\tmp9qp6c02l.zip to C:\Users\Abdullah Maroof\.cache\language_tool_python.

Downloaded <https://www.languagetool.org/download/LanguageTool-5.7.zip> (<https://www.languagetool.org/download/LanguageTool-5.7.zip>) to C:\Users\Abdullah Maroof\.cache\language_tool_python.

Auto-corrected Sentence: Delete my sole

Detected Mistakes:

Grammar Mistakes:

- Offset 0, length 6, Rule ID: UPPERCASE_SENTENCE_START

Message: This sentence does not start with an uppercase letter.

Suggestion: Delete

delete my sole

^^^^^^

Grammar mistake: [Match({'ruleId': 'UPPERCASE_SENTENCE_START', 'message': 'This sentence does not start with an uppercase letter.', 'replacements': ['Delete'], 'offsetInContext': 0, 'context': 'delete my sole', 'offset': 0, 'errorLength': 6, 'category': 'CASING', 'ruleIssueType': 'typographical', 'sentence': 'delete my sole'})]

```
In [8]: > sentence = "Hee wrotes an eemportant meesage. He do not go to school yeste
corrected_sentence, mistakes = auto_correct_with_mistakes(sentence)

print(f"Original Sentence: {sentence}")
print(f"Auto-corrected Sentence: {corrected_sentence}")

print("\nDetected Mistakes:")
for mistake_type, mistake_detail in mistakes.items():
    if mistake_type == "Grammar":
        print(f"Grammar Mistakes:")
        for mistake in mistake_detail:
            print(f" - {mistake}")
    else:
        print(f"{mistake_type} mistake: {mistake_detail}")
```

Original Sentence: Hee wrotes an eemportant meesage. He do not go to scho
ol yesterday!

Auto-corrected Sentence: He wrote an important message. He does not go to
school yesterday !

Detected Mistakes:

Spelling mistake: True

Punctuation mistake: True

Grammar Mistakes:

- Offset 29, length 2, Rule ID: COMMA_PARENTHESIS_WHITESPACE

Message: Don't put a space before the full stop.

Suggestion: .

He wrote an important message . He do not go to school yesterday !

^^

- Offset 35, length 2, Rule ID: HE_VERB_AGR

Message: The pronoun 'He' is usually used with a third-person or a past t
ense verb.

Suggestion: does; did

He wrote an important message . He do not go to school yesterday !

^^

In []: >

In []: >