```
import sqlite3
import unicodedata

#A function which removes unwanted articles and diacritical marks from the words.

def simplifyWord(string):

    for i in ["la ", "le ","l'","l'","to "," ","un ","une "]:
        string = string.replace(i, "")

    #Gets rid of diacritical marks.
    string = ''.join(c for c in unicodedata.normalize('NFD', string) if unicodedata.category(c) != 'Mn')

    return string

def main():

    text_files = ["diversite", "marginalise","criminels"]

    word_class_dic = {"a":"adjective", "n":"noun", "v":"verb", "p":"phrase"}

    #connect to the database.
    with sqlite3.connect('database.db') as connection:

        #Create a cursor.
        cur = connection.cursor()

        french_primary_key = 0
        english_primary_key = 0

        #Clear any data from the last compilation of the database.
        cur.execute("DELETE FROM FrenchWords")
        cur.execute("DELETE FROM EnglishWords")
        cur.execute("DELETE FROM FrenchGender")
        cur.execute("DELETE FROM Translations")
        connection.commit()

        #Goes through each vocab file.
        for text_file in text_files:

            topic = text_file

            #Opens the file.
            with open(f"{text_file}.txt","r",encoding="utf-8") as file:

                #Goes through each line in the file.
                for line in file.readlines():

                    #Creates a list of each component.
                    line = line.replace("\n","").split(",")

                    #This chunk splits the line into its components.
                    french_word = line[0]
                    french_word_simplified = simplifyWord(french_word)
                    french_word_length = len(french_word_simplified)
                    if "-" not in line[1]:
                        english_words = (line[1],)
                    else:
                        english_words = line[1].split("-")
                    word_class = line[2]

                    #Incriments the french_primary_key so that it's unique.
                    french_primary_key += 1

                    cur.execute(f'''
                    INSERT INTO FrenchWords (FrenchWordID, FrenchWord, WordClass, Topic, WordLength, FrenchWordForC
                    VALUES ({french_primary_key}, "{french_word}", "{word_class_dic[word_class]}", "{topic}", {fre

                    #If the word is a noun or an adjecticve then note the gender.
                    if word_class=="n" or word_class=="a":
                        gender = line[3]

                        cur.execute(f'''
                        INSERT INTO FrenchGender (FrenchWordID, Gender)
                        VALUES ({french_primary_key}, "{gender}")''')
```

```
                    #Goes through each English word (usually there's only one but sometimes there's two).
                    for word in english_words:

                        english_word_simplified = simplifyWord(word)
                        english_word_length = len(english_word_simplified)

                        #Checks to see if the English word already exists within the database
                        found = False
                        english_word_id = english_primary_key
                        for word_2 in cur.execute('''SELECT EnglishWord, EnglishWordID FROM EnglishWords''').fetcha
                            if word_2[0] == word:
                                found = True
                                english_word_id = word_2[1]

                        #If the English word is not already in the database then add it.
                        if not found:

                            english_primary_key+=1
                            english_word_id = english_primary_key

                            cur.execute(f'''
                            INSERT INTO EnglishWords (EnglishWordID, EnglishWord, WordLength, EnglishWordForCrosswo
                            VALUES ({english_primary_key}, "{word}", {english_word_length}, "{english_word_simplifi

                        cur.execute(f'''
                        INSERT INTO Translations (FrenchWordID, EnglishWordID)
                        VALUES ({french_primary_key}, {english_word_id})''')

                #Commit the changes.
                connection.commit()


def displayDatabase():

    with sqlite3.connect('database.db') as connection:
        cur = connection.cursor()

        print()

        for table in cur.execute("SELECT name FROM sqlite_master WHERE type='table';").fetchall():

            print(cur.execute(f"PRAGMA table_info({table[0]});").fetchall())
            print()

        #print("-------------------------------------------------------")

        for table in cur.execute("SELECT name FROM sqlite_master WHERE type='table';").fetchall():

            values = cur.execute(f"SELECT * FROM {table[0]}").fetchall()
            print(f"{table[0]}: {values}\n")

if __name__ == "__main__":
    main()
    #displayDatabase()
```