محمدمهدی تویسرکانی

تاریخ تحویل : 1403/12/22

تمرین ۱: استخراج کلمات یکتا از یک متن صورت سوال: برنامه‌ای بنویسید که از کاربر یک جمله بگیرد، سپس:

کلماتی که تکراری نیستند را پیدا کند.

آن‌ها را بر اساس طول کلمه مرتب کند.

فقط کلماتی که طولشان بیشتر از ۳ است را نمایش دهد.

In [5]:
```python
def main():
    # Get a sentence from the user
    sentence = input("Please enter a sentence: ")

    # Split the sentence into words and convert to lowercase
    words = sentence.split()
    words = [word.lower() for word in words]  # Normalize to lowercase

    # Find unique words (not repeated)
    unique_words = set(word for word in words if words.count(word) == 1)

    # Filter words longer than 3 characters
    long_words = [word for word in unique_words if len(word) > 3]

    # Sort the words by their length
    sorted_long_words = sorted(long_words, key=len)

    # Display the result
    print("Words longer than 3 characters, sorted by length:")
    for word in sorted_long_words:
        print(word)

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

```
Please enter a sentence:  I am a mechanical engineer.
Words longer than 3 characters, sorted by length:
engineer.
mechanical
```

تمرین ۲: شمارش و مرتب‌سازی حروف صورت سوال: برنامه‌ای بنویسید که از کاربر یک رشته بگیرد و .

هر حرفی که در رشته وجود دارد را بشمارد.

بر اساس تعداد تکرار از بیشترین به کمترین مرتب کند.

حروفی که فقط یک بار آمده‌اند را حذف کند. nd{document}

In [15]:
```python
from collections import Counter

def main():
    # Take input from the user
    user_input = input("Enter a string: ")

    # Remove spaces and convert to lowercase for consistent counting
    cleaned_input = ''.join(user_input.split()).lower()

    # Count the occurrences of each letter
    letter_count = Counter(cleaned_input)

    # Remove letters that occur only once
    filtered_count = {letter: count for letter, count in letter_count.items

    # Sort by number of occurrences (highest to lowest)
    sorted_count = sorted(filtered_count.items(), key=lambda item: item[1],

    # Display results
    print("Letters sorted by occurrence (highest to lowest):")
    for letter, count in sorted_count:
        print(f"{letter}: {count}")

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

```
Enter a string:  Mohammad Mahdi
```

```
Letters sorted by occurrence (highest to lowest):
m: 4
a: 3
h: 2
d: 2
```

تمرین ۳: بازی حدس عدد صورت سوال: برنامه یک عدد تصادفی بین ۱ تا ۵۰ تولید کند. از کاربر بخواهد که حدس بزند .

اگر عدد کمتر بود، بگوید "عدد بزرگ‌تر است!"

اگر عدد بیشتر بود، بگوید "عدد کوچک‌تر است!"

تا زمانی که عدد درست حدس زده نشده، ادامه دهد.

In [6]:
```python
# import random

# Generate a random number between 1 and 50
random_number = random.randint(1, 50)
guessed = False

print("Welcome to the number guessing game!")
print("I have selected a random number between 1 and 50. Can you guess it?"

while not guessed:
```

```
    # Ask the user for their guess
    user_guess = input("Enter your guess: ")

    # Validate the input
    if not user_guess.isdigit():
        print("Please enter a valid number!")
        continue

    user_guess = int(user_guess)

    # Check the user's guess against the random number
    if user_guess < random_number:
        print("The number is bigger!")
    elif user_guess > random_number:
        print("The number is smaller!")
    else:
        print("Congratulations! You guessed the number correctly!")
        guessed = True
```

```
Welcome to the number guessing game!
I have selected a random number between 1 and 50. Can you guess it?

Enter your guess:  32

The number is smaller!

Enter your guess:  24

The number is smaller!

Enter your guess:  18

The number is smaller!

Enter your guess:  14

The number is smaller!

Enter your guess:  8

The number is smaller!

Enter your guess:  5

The number is bigger!

Enter your guess:  6

Congratulations! You guessed the number correctly!
```

تمرین ۴: فیلتر کردن و چرخش لیست صورت سوال: از کاربر یک لیست از اعداد بگیرد .

فقط اعداد زوج را نگه دارد.

لیست را یک واحد به چپ بچرخاند.

جمع اعداد جدید را نمایش دهد.

In [10]:

```
def main():
    # Get a list of numbers from the user
    user_input = input("Enter a list of numbers separated by spaces: ")
    numbers = list(map(int, user_input.split()))

    # Keep only even numbers
```

```
    even_numbers = [num for num in numbers if num % 2 == 0]

    # Rotate the list one unit to the left
    if even_numbers:  # Ensure the list is not empty
        rotated = even_numbers[1:] + even_numbers[:1]
    else:
        rotated = even_numbers  # No even numbers, keep it empty

    # Calculate the sum of the new numbers
    total_sum = sum(rotated)

    # Display the results
    print("Even numbers:", even_numbers)
    print("Rotated list:", rotated)
    print("Sum of the new numbers:", total_sum)

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

```
Enter a list of numbers separated by spaces:  5 8 12 17 23 28 32 35

Even numbers: [8, 12, 28, 32]
Rotated list: [12, 28, 32, 8]
Sum of the new numbers: 80
```

تمرین ۵: برعکس کردن کلمات جمله صورت سوال: برنامه‌ای بنویسید که از کاربر یک جمله بگیرد و تمام کلمات آن را برعکس کند اما ترتیب کلمات را تغییر ندهد.

In [16]:

```
def reverse_words(sentence):
    # Split the sentence into words
    words = sentence.split()
    # Reverse each word
    reversed_words = [word[::-1] for word in words]
    # Join the reversed words back into a sentence
    reversed_sentence = ' '.join(reversed_words)
    return reversed_sentence

# Get input from the user
user_input = input("Enter a sentence: ")
# Get the reversed words
result = reverse_words(user_input)
# Display the result
print("Reversed words:", result)
```

```
Enter a sentence:  My name is Mohammad Mahdi and I am a mechanical engineer

Reversed words: yM eman si dammahoM idhaM dna I ma a lacinahcem reenigne
```

تمرین ۶: یافتن میانگین نمرات دانشجویان صورت سوال: از کاربر دو لیست بگیرد:

لیست نام دانشجویان

لیست نمرات هر دانشجو سپس نام یک دانشجو را دریافت کند و میانگین نمرات او را حساب کند.

In [20]:

```python
def main():
    # Ask for the number of students
    num_students = int(input("Enter the number of students: "))

    # Initialize lists for student names and grades
    student_names = []
    student_grades = []

    # Get student names and grades
    for _ in range(num_students):
        name = input("Enter student's name: ")
        grades_input = input(f"Enter grades for {name} (comma separated): "
        grades = list(map(float, grades_input.split(',')))  # Convert grade

        student_names.append(name)
        student_grades.append(grades)

    # Ask for the name of the student to calculate average
    query_name = input("Enter the name of the student to calculate average

    # Find the student and calculate average
    if query_name in student_names:
        index = student_names.index(query_name)
        grades = student_grades[index]

        # Calculate average
        average_grade = sum(grades) / len(grades) if grades else 0
        print(f"The average grade for {query_name} is: {average_grade:.2f}"
    else:
        print(f"Student {query_name} not found.")

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

```
Enter the number of students:  3
Enter student's name:  Ali
Enter grades for Ali (comma separated):  20,18,16,17,18
Enter student's name:  Mohammad Mahdi
Enter grades for Mohammad Mahdi (comma separated):  20,19,18,20,20
Enter student's name:  Amir Ali
Enter grades for Amir Ali (comma separated):  20,19,16,17,17
Enter the name of the student to calculate average grades:  Mohammad Mahdi

The average grade for Mohammad Mahdi is: 19.40
```

In [ ]: