

17SEP PYTHON TUTORIAL SOLUTION

(Adarsh Tiwari)

Problem 1: Top-Scoring Students by Subject using csv (comma-separated values) file

Approach:

- Read data from CSV file containing subject, name, and marks.
- Use a dictionary to track the highest marks and topper(s) for each subject.
- Update the dictionary when a higher or equal mark is found.
- Finally, print the toppers of each subject.

Code:

```
import csv

subject_toppers = {}

with open("marks.csv", newline="") as f:
    reader = csv.DictReader(f)
    for row in reader:
        subject = row["subject"]
        name = row["name"]
        marks = int(row["marks"])

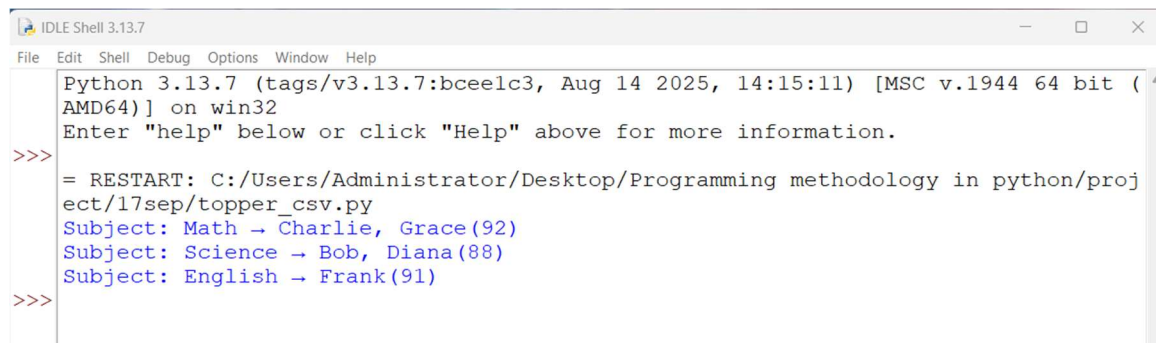
        if subject not in subject_toppers:
            subject_toppers[subject] = (marks, [name])

        else:
            max_marks, students = subject_toppers[subject]
            if marks > max_marks:
                subject_toppers[subject] = (marks, [name])

            elif marks == max_marks:
                students.append(name)
                subject_toppers[subject] = (max_marks, students)

for subject, (marks, students) in subject_toppers.items():
    print("Subject:", subject, "→", end=" ")
    for i in range(len(students)):
        if i == len(students) - 1:
            print(students[i], "(", marks, ")", sep="")
        else:
            print(students[i] + ", ", end=" ")
```

Sample Run:



```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.

>>> = RESTART: C:/Users/Administrator/Desktop/Programming methodology in python/project/17sep/topper_csv.py
      Subject: Math → Charlie, Grace(92)
      Subject: Science → Bob, Diana(88)
      Subject: English → Frank(91)
>>>
```

Problem 2: Word Frequency Filter

Approach:

- Read the contents of a text file and convert all words to lowercase.
- Split the text into words and count frequencies using a dictionary.
- Take user input as threshold value.
- Print words whose frequency is greater than or equal to the threshold.

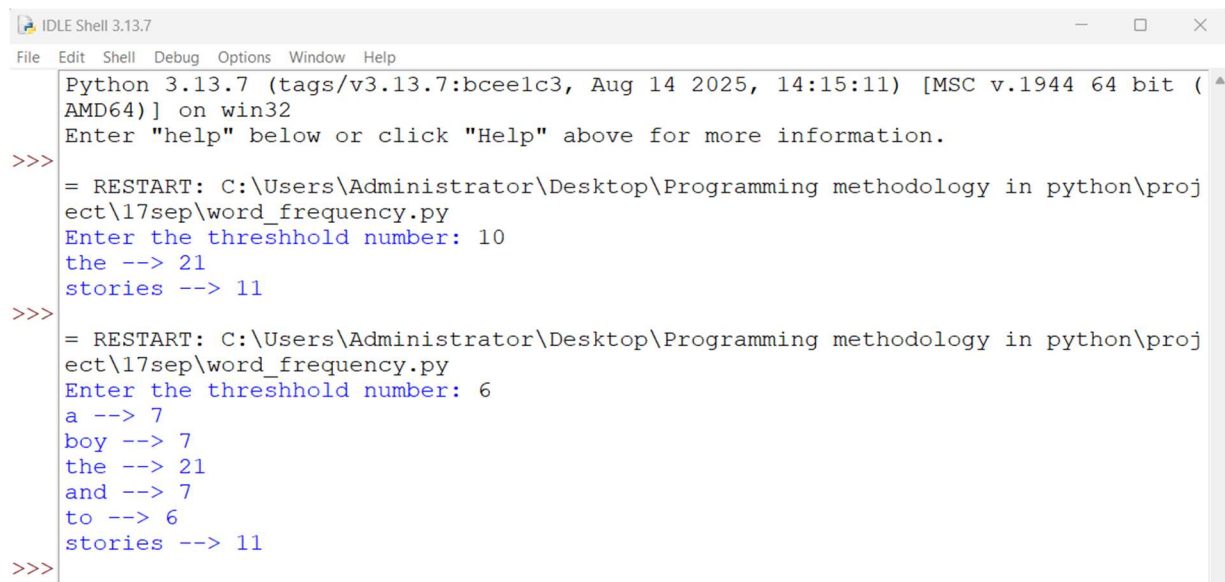
Code:

```
with open ('C://Users//Administrator//Desktop//Programming methodology
in python//project//17sep//story.txt','r') as f:
    r = f.read().lower()
    s = r.split()
    d = {}
    n = int(input("Enter the threshold number: "))

    for i in range(len(s)):
        d[s[i]] = d.get(s[i],0)+1

    for word in d:
        if d[word] >= n:
            print(word, '-->', d[word])
```

Sample Run:



```
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
= RESTART: C:\Users\Administrator\Desktop\Programming methodology in python\project\17sep\word_frequency.py
Enter the threshold number: 10
the --> 21
stories --> 11
>>>
= RESTART: C:\Users\Administrator\Desktop\Programming methodology in python\project\17sep\word_frequency.py
Enter the threshold number: 6
a --> 7
boy --> 7
the --> 21
and --> 7
to --> 6
stories --> 11
>>>
```

Problem 3: Library Borrow Checker

Approach:

- Read the current library stock from a text file.
- Take user input for the book to borrow.
- Check if the book exists and has available copies.
- If available, issue the book and decrease its stock.
- Update the file with the new stock values.

Code:

```
with open ('C://Users//Administrator//Desktop//Programming methodology
in python//project//17sep//library.txt', 'r') as f:
    r = f.read().strip()

    if r:
        d = eval(r)
    else:
        d={}

user = input('Enter the book name: ').lower().strip()

if user in d:
    if d[user] != 0:
        print(user, 'book issued to you.')
        d[user] = d[user] - 1
        print('Library updated.')
    else:
```

```

        print('Out of stock.')
else:
    print('Not found.')

with open ('C:/Users/Administrator/Desktop/Programming methodology in
python/project/17sep/library.txt', 'w') as f:
    f.write(str(d))

```

Sample Run:

```

Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (
AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>>
= RESTART: C:/Users/Administrator/Desktop/Programming methodology in python/proj
ect/17sep/library.py
Enter the book name: b
b book issued to you.
Library updated.
>>>
= RESTART: C:/Users/Administrator/Desktop/Programming methodology in python/proj
ect/17sep/library.py
Enter the book name: b
Out of stock.
>>>
= RESTART: C:/Users/Administrator/Desktop/Programming methodology in python/proj
ect/17sep/library.py
Enter the book name: f
Not found.
>>>

```

Problem 4: Unique Numbers & Statistics

Approach:

- Take an input string and extract all numeric characters.
- Store unique numbers in a set.
- Convert the set to a list and compute sum, count, average, maximum, and minimum.
- Display all calculated values.

Code:

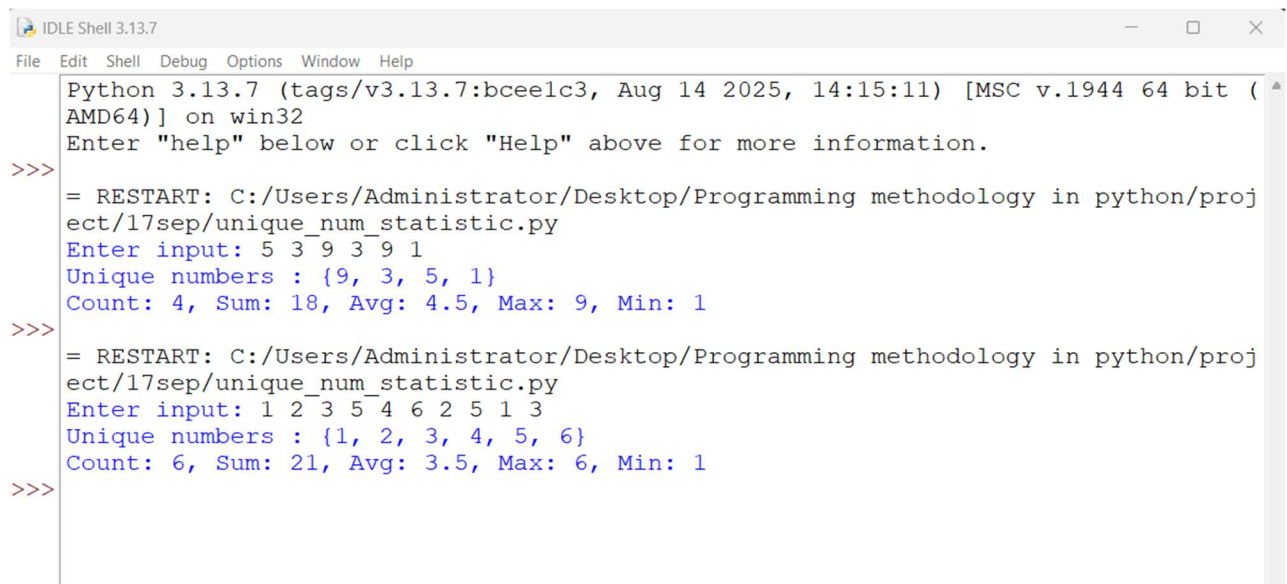
```
s = set()
n = input('Enter input: ')
for i in n:
    if i.isnumeric():
        s.add(int(i))

print('Unique numbers :',s)
l = list(s)
count = 0
suml = 0
maxl = l[0]
minl = l[0]

for i in l:
    suml += i
    count += 1
    if i >= maxl:
        maxl = i
    if i <= minl:
        minl = i

print('Count:', count, end=', ')
print('Sum:', suml, end=', ')
print('Avg:', suml/count, end=', ')
print('Max:', maxl, end=', ')
print('Min:', minl)
```

Sample Run:



```
IDLE Shell 3.13.7
File Edit Shell Debug Options Window Help
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> = RESTART: C:/Users/Administrator/Desktop/Programming methodology in python/project/17sep/unique_num_statistic.py
Enter input: 5 3 9 3 9 1
Unique numbers : {9, 3, 5, 1}
Count: 4, Sum: 18, Avg: 4.5, Max: 9, Min: 1
>>> = RESTART: C:/Users/Administrator/Desktop/Programming methodology in python/project/17sep/unique_num_statistic.py
Enter input: 1 2 3 5 4 6 2 5 1 3
Unique numbers : {1, 2, 3, 4, 5, 6}
Count: 6, Sum: 21, Avg: 3.5, Max: 6, Min: 1
>>>
```

Problem 5: Student Attendance Manager

Approach:

- Maintain a master list of student roll numbers.
- Read present roll numbers from a file.
- Find absent students by comparing with master list.
- Print absent roll numbers and write them to a new file.

Code:

```
master = [1,2,3,4,5,6,7,8,9,10]

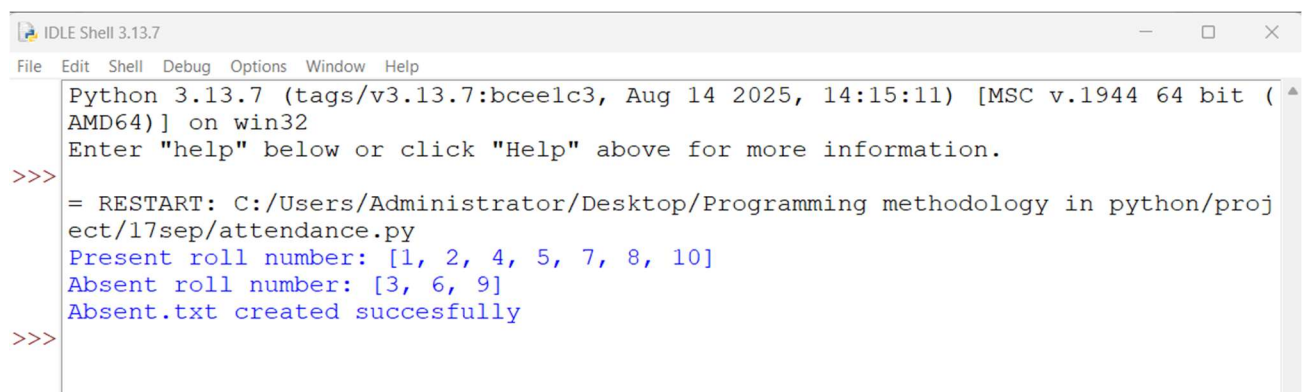
with open('attendance.txt','r') as f:
    present = []
    for line in f:
        present.append(int(line.strip()))
    print("Present roll number:", present)

absent = [roll_no for roll_no in master if roll_no not in present]
print("Absent roll number:", absent)

with open('absent.txt','w') as f:
    for roll_no in absent:
        f.write(str(roll_no))
        f.write('\n')

print('Absent.txt created succesfully')
```

Sample Run:



```
IDLE Shell 3.13.7
File Edit Shell Debug Options Window Help
Python 3.13.7 (tags/v3.13.7:bceelc3, Aug 14 2025, 14:15:11) [MSC v.1944 64 bit (AMD64)] on win32
Enter "help" below or click "Help" above for more information.
>>> = RESTART: C:/Users/Administrator/Desktop/Programming methodology in python/project/17sep/attendance.py
Present roll number: [1, 2, 4, 5, 7, 8, 10]
Absent roll number: [3, 6, 9]
Absent.txt created succesfully
>>>
```