EEE110 Computer Programming

Files and Exceptions & Lists and Tuples

Dr Kasım Zor

April 14, 2022

Files

Writing Data to a File

```python
def main():
    # Open a file named philosophers.txt.
    outfile = open('philosophers.txt', 'w')

    # Write the names of three philosphers to the file.
    outfile.write('John Locke\n')
    outfile.write('David Hume\n')
    outfile.write('Edmund Burke\n')

    # Close the file.
    outfile.close()

main()
```

Files

Reading Data from a File

```python
def main():
    # Open a file named philosophers.txt.
    infile = open('philosophers.txt', 'r')

    # Read the file's contents.
    file_contents = infile.read()

    # Close the file.
    infile.close()

    # Print the data that was read into memory.
    print(file_contents)

main()
```

Files

Reading One Line at a Time

```python
def main():
    infile = open('philosophers.txt', 'r')

    line1 = infile.readline()
    line2 = infile.readline()
    line3 = infile.readline()

    infile.close()

    print(line1)
    print(line2)
    print(line3)

main()
```

Files

Writing Names

```python
def main():
    print('Enter the names of three friends.')
    name1 = input('Friend #1: ')
    name2 = input('Friend #2: ')
    name3 = input('Friend #3: ')

    myfile = open('friends.txt', 'w')

    myfile.write(name1 + '\n')
    myfile.write(name2 + '\n')
    myfile.write(name3 + '\n')

    myfile.close()
    print('The names were written to friends.txt.')
main()
```

Files

Reading a String and Stripping the Newline from It

```python
def main():
    infile = open('philosophers.txt', 'r')
    line1 = infile.readline()
    line2 = infile.readline()
    line3 = infile.readline()

    line1 = line1.rstrip('\n')
    line2 = line2.rstrip('\n')
    line3 = line3.rstrip('\n')
    infile.close()

    print(line1)
    print(line2)
    print(line3)
main()
```

Files

Writing Numeric Data

```python
def main():
    outfile = open('numbers.txt', 'w')

    num1 = int(input('Enter a number: '))
    num2 = int(input('Enter another number: '))
    num3 = int(input('Enter another number: '))

    outfile.write(str(num1) + '\n')
    outfile.write(str(num2) + '\n')
    outfile.write(str(num3) + '\n')

    outfile.close()
    print('Data written to numbers.txt')

main()
```

Files

Reading Numeric Data

```python
def main():
    infile = open('numbers.txt', 'r')

    num1 = int(infile.readline())
    num2 = int(infile.readline())
    num3 = int(infile.readline())

    infile.close()

    total = num1 + num2 + num3

    print('The numbers are:', num1, num2, num3)
    print('Their total is:', total)

main()
```

Files

Using Loops for Writing to a File

```python
def main():
    num_days = int(input('For how many days do ' + \
                         'you have sales? '))

    sales_file = open('sales.txt', 'w')

    for count in range(1, num_days + 1):
        sales = float(input('Enter the sales for day #' + \
                            str(count) + ': '))
        sales_file.write(str(sales) + '\n')

    sales_file.close()
    print('Data written to sales.txt.')

main()
```

Files

Using while Loops for Reading from a File

```python
def main():
    sales_file = open('sales.txt', 'r')

    line = sales_file.readline()

    while line != '':
        amount = float(line)
        print(format(amount, '.2f'))
        line = sales_file.readline()

    sales_file.close()

main()
```

Files

Using for Loops for Reading from a File

```python
def main():
    sales_file = open('sales.txt', 'r')

    for line in sales_file:
        amount = float(line)
        print(format(amount, '.2f'))

    sales_file.close()

main()
```

Files

Question 1. Working with Files (15 minutes)

Adana Demirspor is a football club founded in 1940 by local railway workers in Adana and the club has been still competing in Turkish Super League to participate in the European Cup for the football season 2022-2023.

Assume that result of a football match is evaluated as three points for a win, one point for a draw, and no points for a defeat.

Design a Python program that allows user to enter the result (in points) of each match of Adana Demirspor. The results shall be saved to a file. Next, the program shall read the contents of the file, display the results (in points), and then show the total points that Adana Demirspor gets.

Files

Solution 1/2

```python
def write():
    num_games = int(input('How many games did Adana Demirspor
    write_file = open('ADS.txt', 'w')
    print('Enter the results (in points) for each game.')
    for count in range(1, num_games + 1):
        result = int(input('Game #' + str(count) + ': '))
        write_file.write(str(result) + '\n')
    write_file.close()
    print('The results have been saved to ADS.txt.')

write()
```

Files

Solution 2/2

```python
def read():
    read_file = open('ADS.txt', 'r')
    total = 0
    count = 0
    print('The results in ADS.txt file:')
    for line in read_file:
        pts = int(line)
        count += 1
        print('Game #' + str(count) + ': ', pts)
        total += pts
    print('Total points that Adana Demirspor got:', total)
    read_file.close()

read()
```

Exceptions

Division

```python
def main():
    num1 = 1 #int(input('Enter a number: '))
    num2 = 0 #int(input('Enter another number: '))
    result = num1 / num2
    print(num1, 'divided by', num2, 'is', result)
main()
## Error in py_call_impl(callable, dots$args, dots$keywords):
##
## Detailed traceback:
##   File "<string>", line 1, in <module>
##   File "<string>", line 4, in main
```

Exceptions

Division

```python
def main():
    num1 = 1 #int(input('Enter a number: '))
    num2 = 0 #int(input('Enter another number: '))

    if num2 != 0:
        result = num1 / num2
        print(num1, 'divided by', num2, 'is', result)
    else:
        print('Cannot divide by zero.')

main()
## Cannot divide by zero.
```

Lists and Tuples

Assigning the Elements of a List

```python
NUM_DAYS = 5
def main():
    sales = [0] * NUM_DAYS
    index = 0
    print('Enter the sales for each day.')
    while index < NUM_DAYS:
        print('Day #', index + 1, ': ', sep='', end='')
        sales[index] = float(input())
        index += 1

    print('Here are the values you entered:')

    for value in sales:
        print(value)
main()
```

Lists and Tuples

Finding Items in Lists

```python
def main():
    prod_nums = ['V475', 'F987', 'Q143', 'R688']

    search = input('Enter a product number: ')

    if search in prod_nums:
        print(search, 'was found in the list.')
    else:
        print(search, 'was not found in the list.')

main()
```

Lists and Tuples

List Methods: append()

```python
def main():
    name_list = []
    again = 'Y'
    while again.upper() == 'Y':
        name = input('Enter a name: ')
        name_list.append(name)
        print('Do you want to add another name?')
        again = input('y = yes, anything else = no: ')
        print()
    print('Here are the names you entered.')

    for name in name_list:
        print(name)
main()
```

Lists and Tuples

List Methods: index()

```python
def main():
    food = ['Pizza', 'Burgers', 'Chips']
    print('Here are the items in the food list:')
    print(food)
    item = input('Which item should I change? ')

    try:
        item_index = food.index(item)
        new_item = input('Enter the new value: ')
        food[item_index] = new_item
        print('Here is the revised list:')
        print(food)
    except ValueError:
        print('That item was not found in the list.')
main()
```

Lists and Tuples

List Methods: insert()

```python
def main():
    names = ['James', 'Kathryn', 'Bill']
    print('The list before the insert:')
    print(names)

    names.insert(0, 'Joe')

    print('The list after the insert:')
    print(names)

main()
```

Lists and Tuples

List Methods: remove()

```python
def main():
    food = ['Pizza', 'Burgers', 'Chips']
    print('Here are the items in the food list:')
    print(food)
    item = input('Which item should I remove? ')

    try:
        food.remove(item)
        print('Here is the revised list:')
        print(food)

    except ValueError:
        print('That item was not found in the list.')
main()
```

Lists and Tuples

Question 2. Using List Elements in a Math Expression (15 minutes)

Megan owns a small neighborhood coffee shop, and she has six employees who work as baristas (coffee bartenders). All of the employees have the same hourly pay rate. Megan has asked you to design a program that will allow her to enter the number of hours worked by each employee, then display the amounts of all the employees' gross pay. You determine the program should perform the following steps:

1. For each employee: get the number of hours worked and store it in a list element.
2. For each list element: use the value stored in the element to calculate an employee's gross pay. Display the amount of the gross pay.

Lists and Tuples

Solution 2

```python
NUM_EMPLOYEES = 6
def main():
    hours = [0] * NUM_EMPLOYEES
    for index in range(NUM_EMPLOYEES):
        print('Enter the hours worked by employee ', \
                index + 1, ': ', sep='', end='')
        hours[index] = float(input())
    pay_rate = float(input('Enter the hourly pay rate: '))
    for index in range(NUM_EMPLOYEES):
        gross_pay = hours[index] * pay_rate
        print('Gross pay for employee ', index + 1, ': $', \
                format(gross_pay, ',.2f'), sep='')
main()
```

Lists and Tuples

Question 3. Processing a List (15 minutes)

Dr. LaClaire gives a series of exams during the semester in her chemistry class. At the end of the semester, she drops each student's lowest test score before averaging the scores. She has asked you to design a program that will read a student's test scores as input and calculate the average with the lowest score dropped. Here is the algorithm that you developed:

- Get the student's test scores.
- Calculate the total of the scores.
- Find the lowest score.
- Subtract the lowest score from the total. This gives the adjusted total.
- Divide the adjusted total by 1 less than the number of test scores. This is the average.
- Display the average.

Lists and Tuples

Solution 3 - Part 1/3

```python
def main():

    scores = get_scores()

    total = get_total(scores)

    lowest = min(scores)

    total -= lowest

    average = total / (len(scores) - 1)

    print('The average, with the lowest score dropped', \
          'is:', average)
```

Lists and Tuples

Solution 3 - Part 2/3

```python
def get_scores():
    test_scores = []

    again = 'y'

    while again == 'y':
        value = float(input('Enter a test score: '))
        test_scores.append(value)

        print('Do you want to add another score?')
        again = input('y = yes, anything else = no: ')
        print()

    return test_scores
```
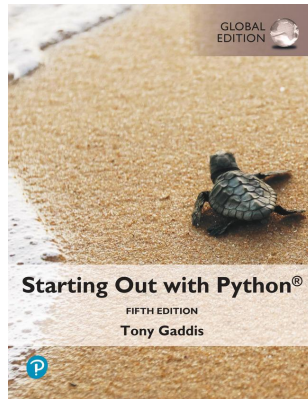

Lists and Tuples

Solution 3 - Part 3/3

```python
def get_total(value_list):
    total = 0.0

    for num in value_list:
        total += num

    return total

main()
```

References

Aforementioned contents are adapted from the book:

- 'Starting out with Python' written by Tony Gaddis.