Practical-6

February 11, 2025

0.1 Practical 6 :-

Name: Sarthak Sanay
Enrollment No: 230031101611051

0.1.1 Problem Statement 1 :-

Online Shopping Cart: Imagine you're developing an online shopping platform. Create a Python program that simulates a user's shopping cart. - Allow the user to add product names and prices to their cart. - Display the current items in the cart. - Allow the user to remove items from the cart. - Calculate the total price and display the total number of items in the cart.

[7]:
```
# Approach using list of tuples, where each product name and price is inserted as a tuple into the list cart.

cart = [] # initialize empty list of tuples

while True:

    print("\nSHOPPING CART OPERATIONS :- \n1. Add product to cart. \n2. Display items in cart. \n3. Remove item from cart. \n4. Calculate cart total. \n0. Exit the program.")
    ch = int(input("Enter choice: "))

    if ch == 1:
        product_name = input("\nEnter product name to add: ")
        product_price = float(input("Enter product price: "))
        cart.append((product_name, product_price))
        print(f"{product_name} added to cart successfully!\n")

    elif ch == 2:
        if len(cart) == 0:
            print("\nYour cart is empty!\n")
        else:
            print("\nItems in your cart are :-")
            for item in cart:
                print(f"{item[0]}\t\tRs. {item[1]:.2f}")
            print() # to add a newline
```

```
    elif ch == 3:
        product_name = input("\nEnter product name to remove: ")
        product_in_cart = False
        for item in cart:
            if item[0] == product_name:
                product_in_cart = True
                cart.remove(item)
                # break # to remove only first occurence of the product
        if product_in_cart:
            print(f"{product_name} removed from cart!\n")
        else:
            print(f"{product_name} does not exist in the cart!\n")

    elif ch == 4:
        total_price = 0
        for item in cart:
            total_price += item[1]
        print("\nTotal no. of items in cart:\t", len(cart))
        print(f"Total price of items in cart:\t Rs. {total_price:.2f}\n")

    elif ch == 0:
        print("\nExited the program successfully!")
        break

    else:
        print("\nEnter correct choice.\n")
```

```
SHOPPING CART OPERATIONS :-
1. Add product to cart.
2. Display items in cart.
3. Remove item from cart.
4. Calculate cart total.
0. Exit the program.

Enter choice:  1

Enter product name to add:  iPhone 16
Enter product price:  79900

iPhone 16 added to cart successfully!


SHOPPING CART OPERATIONS :-
1. Add product to cart.
2. Display items in cart.
3. Remove item from cart.
4. Calculate cart total.
```

```
0. Exit the program.

Enter choice:  1

Enter product name to add:  Apple Airpods
Enter product price:  12900

Apple Airpods added to cart successfully!


SHOPPING CART OPERATIONS :-
1. Add product to cart.
2. Display items in cart.
3. Remove item from cart.
4. Calculate cart total.
0. Exit the program.

Enter choice:  1

Enter product name to add:  Apple Watch
Enter product price:  29900

Apple Watch added to cart successfully!


SHOPPING CART OPERATIONS :-
1. Add product to cart.
2. Display items in cart.
3. Remove item from cart.
4. Calculate cart total.
0. Exit the program.

Enter choice:  2


Items in your cart are :-
iPhone 16               Rs. 79900.00
Apple Airpods           Rs. 12900.00
Apple Watch             Rs. 29900.00


SHOPPING CART OPERATIONS :-
1. Add product to cart.
2. Display items in cart.
3. Remove item from cart.
4. Calculate cart total.
0. Exit the program.

Enter choice:  4


Total no. of items in cart:      3
```

```
Total price of items in cart:    Rs. 122700.00


SHOPPING CART OPERATIONS :-
1. Add product to cart.
2. Display items in cart.
3. Remove item from cart.
4. Calculate cart total.
0. Exit the program.

Enter choice:  3

Enter product name to remove:  Apple Watch

Apple Watch removed from cart!


SHOPPING CART OPERATIONS :-
1. Add product to cart.
2. Display items in cart.
3. Remove item from cart.
4. Calculate cart total.
0. Exit the program.

Enter choice:  2


Items in your cart are :-
iPhone 16               Rs. 79900.00
Apple Airpods           Rs. 12900.00


SHOPPING CART OPERATIONS :-
1. Add product to cart.
2. Display items in cart.
3. Remove item from cart.
4. Calculate cart total.
0. Exit the program.

Enter choice:  4


Total no. of items in cart:      2
Total price of items in cart:    Rs. 92800.00


SHOPPING CART OPERATIONS :-
1. Add product to cart.
2. Display items in cart.
3. Remove item from cart.
4. Calculate cart total.
```

```
0. Exit the program.

Enter choice:  0


Exited the program successfully!
```

0.1.2 Problem Statement 2 :-

Student Grade Analyzer: As a teacher, you have a list of student names and scores (out of 100) for a test. - Write a Python program that calculates the average score and identifies students who scored above the average.

[12]:
```python
# list of tuples containing student_name and student_marks as tuple pair
list = [("Sarthak Sanay", 89),
        ("Arnav Singh", 75),
        ("Kunal Mishra", 65),
        ("Rahul Singal", 52),
        ("Abhishek Tripathi", 94),
        ("Hardik Sharma", 44),
        ("Arjun Kumar", 33),
        ("Aman Yadav", 86),
        ("Vaibhav Raj", 96),
        ("Harsh Rathore", 88)]

# to find average marks of students
total = 0
for marks in list:
    total += marks[1]
average_marks = total / len(list)

# printing names of students who have scored above the average marks
print("Average marks:", average_marks)
print(f"\nStudents who have scored above {average_marks} are :-\n")
for student in list:
    if student[1] > average_marks:
        print(student[0])
```

```
Average marks: 72.2

Students who have scored above 72.2 are :-

Sarthak Sanay
Arnav Singh
Abhishek Tripathi
Aman Yadav
Vaibhav Raj
Harsh Rathore
```

0.1.3 Problem Statement 3 :-

Temperature Converter: You're building a weather app. Create a Python program that converts temperatures between Celsius and Fahrenheit. - Prompt the user to enter a temperature value and a unit (C or F). - Calculate and display the converted temperature. - Example: If the user enters 32 C, the program should output 89.6 F.

[1]:
```
print("TEMPERATURE CONVERTER :- \nEnter 'EXIT' to terminate the program.\n")

while True:
    user_temp = input("Enter temperature value and a unit (C or F): ")

    if user_temp == "EXIT":
        print("\nExited the program successfully!")
        break

    temperature = float(user_temp[ :user_temp.find(" ")])
    unit = user_temp[(user_temp.find(" ")) + 1: ]
    unit = unit.upper()

    if unit == 'C':
        temp_fahr = (temperature * (9/5)) + 32
        print(f"Temperature in Fahrenheit: {temp_fahr:.2f} F\n")

    elif unit == 'F':
        temp_cel = (temperature - 32) * (5/9)
        print(f"Temperature in Celsius: {temp_cel:.2f} C\n")

    else:
        print("Enter correct unit for temperature.\n")
```

```
TEMPERATURE CONVERTER :-
Enter 'EXIT' to terminate the program.

Enter temperature value and a unit (C or F):  24 C
Temperature in Fahrenheit: 75.20 F

Enter temperature value and a unit (C or F):  31 C
Temperature in Fahrenheit: 87.80 F

Enter temperature value and a unit (C or F):  33 F
Temperature in Celsius: 0.56 C

Enter temperature value and a unit (C or F):  20 F
```

```
Temperature in Celsius: -6.67 C


Enter temperature value and a unit (C or F):  EXIT


Exited the program successfully!
```

0.1.4 Problem Statement 4 :-

Vowel Counter: You're developing a text analysis tool. Write a Python program that reads a sentence from the user. - Count the number of vowels (a, e, i, o, u) in the sentence. - Display the total count of each vowel.

[2]:
```python
print("Text Analysis Tool :-\n")

sentence = input("Enter a sentence: ")
sentence = sentence.upper()

# dictionary having key-value pair with vowels as keys, and the values being␣
 ↪each of their specific count
vowels = {'A':0, 'E':0, 'I':0, 'O':0, 'U':0}

# checks if a char in the str is vowel, and increments the particular vowel␣
 ↪accordingly where required
for char in sentence:
    if char in vowels:
        vowels[char] += 1

# to print the total no. of vowels as well as the count of each vowel
print("\nCount of each vowel is as follows:-")
total_vowels = 0
for vowel, count in vowels.items():
    print(f"{vowel}: {count}")
    total_vowels += count
print("\nTotal count of vowels in the sentence:", total_vowels)
```

```
Text Analysis Tool :-


Enter a sentence:  The quick brown fox jumps over the lazy dog


Count of each vowel is as follows:-
A: 1
E: 3
I: 1
O: 4
U: 2
```

```
Total count of vowels in the sentence: 11
```

0.1.5 Problem Statement 5 :-

Bookstore Inventory: As a bookstore manager, you create a list of book titles and their corresponding quantities in stock. - Write a Python program that asks if the user is a manager or a normal user. - Then allow the manager to create and update the list of books. - Allow other users to search for a book title and check its availability. - Prompt the user to enter a book title. - If the book is in stock, display the quantity available; otherwise, show an appropriate message.

[2]:
```
# Approach using two lists (one to maintain the book titles, and the other one to maintain the corresponding quantities)

print("BOOKSTORE INVENTORY :- \nEnter 'EXIT' to terminate the program.\n")

# added few books as example
book_titles = ["Harry Potter", "Famous Five", "Merchant of Venice"]
book_quantity = [12, 5, 3]

while True:
    user_type = input("Enter M for 'Manager' or U for 'User' \nEnter choice: ")
    user_type = user_type.upper()

    if user_type == 'M':
        ch = int(input("Enter 1 to add a new book. \nEnter 2 to remove a book. \nEnter 3 to update quantity. \nEnter choice: "))

        if ch == 1:
            book_add = input("\nEnter book title: ")
            quantity_add = int(input("Enter quantity: "))
            book_titles.append(book_add)
            book_quantity.append(quantity_add)
            print(f"Book '{book_add}' with {quantity_add} quantity successfully added to inventory.\n")

        elif ch == 2:
            book_remove = input("\nEnter book title: ")
            book_found = False
            for index, book in enumerate(book_titles):
                if book == book_remove:
                    book_titles.remove(book_remove)
                    book_quantity.pop(index)
                    book_found = True
                    break
            if book_found:
                print(f"Book '{book_remove}' removed from inventory!\n")
            else:
                print(f"Book '{book_remove}' not found in inventory!\n")
```

```
        elif ch == 3:
            book_update = input("\nEnter book title: ")
            book_found = False
            for index, book in enumerate(book_titles):
                if book == book_update:
                    quantity_update = int(input("Enter new quantity value: "))
                    book_quantity[index] = quantity_update
                    book_found = True
                    break
            if book_found:
                print(f"Quantity of book '{book_update}' updated successfully!
↪\n")
            else:
                print(f"Book '{book_update}' not found in inventory!\n")

        else:
            print("\nEnter correct choice.\n")


    elif user_type == 'U':
        book_user = input("\nEnter a book title: ")
        book_found = False
        for index, book in enumerate(book_titles):
            if book == book_user:
                print(f"Book '{book_user}' is available in the inventory!")
                print(f"Quantity: {book_quantity[index]}\n")
                book_found = True
                break
        if book_found == False:
            print(f"Book '{book_user}' is not available in the inventory!\n")

    elif user_type == "EXIT":
        print("\nExited the program successfully!")
        break

    else:
        print("\nEnter correct choice.\n")
        continue
```

```
BOOKSTORE INVENTORY :-
Enter 'EXIT' to terminate the program.


Enter M for 'Manager' or U for 'User'
Enter choice:  M
Enter 1 to add a new book.
```

```
Enter 2 to remove a book.
Enter 3 to update quantity.
Enter choice:  1

Enter book title:  Oliver Twist
Enter quantity:  4

Book 'Oliver Twist' with 4 quantity successfully added to inventory.


Enter M for 'Manager' or U for 'User'
Enter choice:  M
Enter 1 to add a new book.
Enter 2 to remove a book.
Enter 3 to update quantity.
Enter choice:  1

Enter book title:  Wings of Fire
Enter quantity:  8

Book 'Wings of Fire' with 8 quantity successfully added to inventory.


Enter M for 'Manager' or U for 'User'
Enter choice:  U

Enter a book title:  Oliver Twist

Book 'Oliver Twist' is available in the inventory!
Quantity: 4


Enter M for 'Manager' or U for 'User'
Enter choice:  M
Enter 1 to add a new book.
Enter 2 to remove a book.
Enter 3 to update quantity.
Enter choice:  3

Enter book title:  Oliver Twist
Enter new quantity value:  10

Quantity of book 'Oliver Twist' updated successfully!


Enter M for 'Manager' or U for 'User'
Enter choice:  U

Enter a book title:  Oliver Twist

Book 'Oliver Twist' is available in the inventory!
Quantity: 10
```

```
Enter M for 'Manager' or U for 'User'
Enter choice:  M
Enter 1 to add a new book.
Enter 2 to remove a book.
Enter 3 to update quantity.
Enter choice:  2

Enter book title:  Oliver Twist

Book 'Oliver Twist' removed from inventory!


Enter M for 'Manager' or U for 'User'
Enter choice:  U

Enter a book title:  Oliver Twist

Book 'Oliver Twist' is not available in the inventory!


Enter M for 'Manager' or U for 'User'
Enter choice:  Wings of Fire


Enter correct choice.


Enter M for 'Manager' or U for 'User'
Enter choice:  U

Enter a book title:  Wings of Fire

Book 'Wings of Fire' is available in the inventory!
Quantity: 8


Enter M for 'Manager' or U for 'User'
Enter choice:  EXIT


Exited the program successfully!
```