**Name of Student: Raheel Kotwal**

**Roll Number: 45**

**Experiment Number: 5.1**

**Title: Employee Class**

**Theory:** The Code section below has two different snippets of code, one being the my\_module.py file and another being the main.py file..

**Code:**

*def* add():

a=*int*(input("Enter the 1st no. : "))

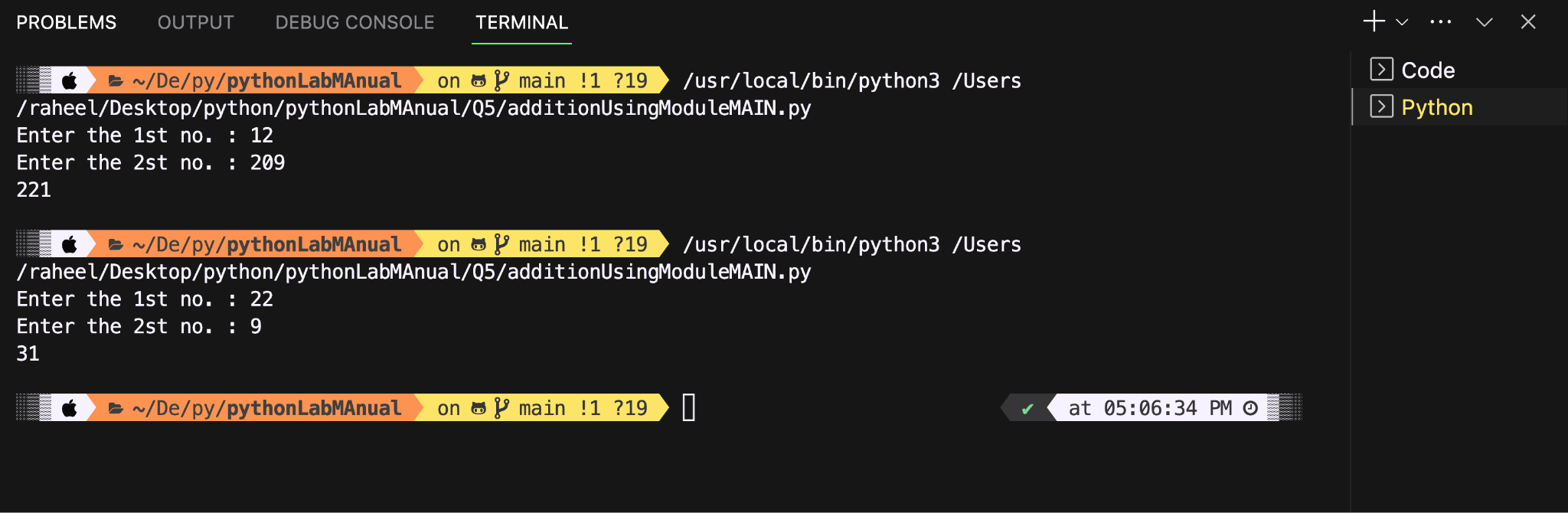
b=*int*(input("Enter the 2st no. : "))

return a+b

**import my\_module**

**print(my\_module.add())**

**Output:(screenshot):**



**Conclusion:**

**The code works perfectly as seen above using two cases. Where user input is taken for both numbers and calculated using m\_module file.**

**Name of Student: Raheel Kotwal**

**Roll Number: 45**

**Experiment Number: 5.2**

**Title: Bank**

**Theory:**  The code below uses a module bank.py and creates an atm-like system such and is used in bankMAIN.py.

**Code:**

**Bank.py:**

**import time**

**import random**

***def* get\_account\_number():**

**while True:**

**account\_no = *int*(input("\nEnter your card number: \n"))**

**if 10000000 <= account\_no <= 99999999:**

**return account\_no**

**else:**

**print("Account number should be of 8 mnumbers :) \n\n")**

***def* withdraw\_money(*balance*):**

**while True:**

**money = *float*(input("\nEnter the money you want to withdraw ₹"))**

**if money > balance:**

**time.sleep(1)**

**print("\nYour balance is lower than the amount you want to withdraw")**

**elif money < 100:**

**print("Minimal amount should be ₹100")**

**else:**

**return money**

***def* deposit\_money(*balance*):**

**money = *float*(input("\nEnter the amount you want to deposit ₹"))**

**return balance + money**

***def* transfer\_money(*balance*, *account\_no*):**

**while True:**

**money = *float*(input("\nEnter the amount you want to transfer : ₹"))**

**ac = *float*(input("\nEnter the account you want to transfer to "))**

**if ac == account\_no:**

**print("\nCan't send money to yourself, can you? \n")**

**elif not (10000000 <= ac < 99999999):**

**print("\nAccount no. should be of 8 digits\n")**

**elif money > balance or money < 100:**

**if money > balance:**

**print("Not enough money in your account \n")**

**else:**

**print("Minimal transfer amount is ₹100\n")**

**else:**

**time.sleep(2)**

**balance -= money**

**print(*f*"\nTransferred amount ₹ {money*:.3f*} To Account with account no: {*int*(ac)}\n")**

**print(*f*"Your bank now has ₹ {balance*:.3f*}")**

**return balance**

bankMAIN.py

import random

from bank import \*

c = random.randint(1000, 10000)

account\_no = get\_account\_number()

print("\n\nChecking your card status please wait... \n")

time.sleep(2)

print("\nWELCOME TO ATM ")

while True:

print("\n\nWhat would you like to do?\n1. Withdrawal\n2. Check balance\n3. Deposit money\n4. Transfer money \n5. Cancel \n")

n = *int*(input())

if n == 1:

c -= withdraw\_money(c)

time.sleep(2)

elif n == 2:

print(*f*"\nYour account has ₹{c*:.3f*}\n")

elif n == 3:

c = deposit\_money(c)

print("Successfully deposited ")

elif n == 4:

c = transfer\_money(c, account\_no)

elif n == 5:

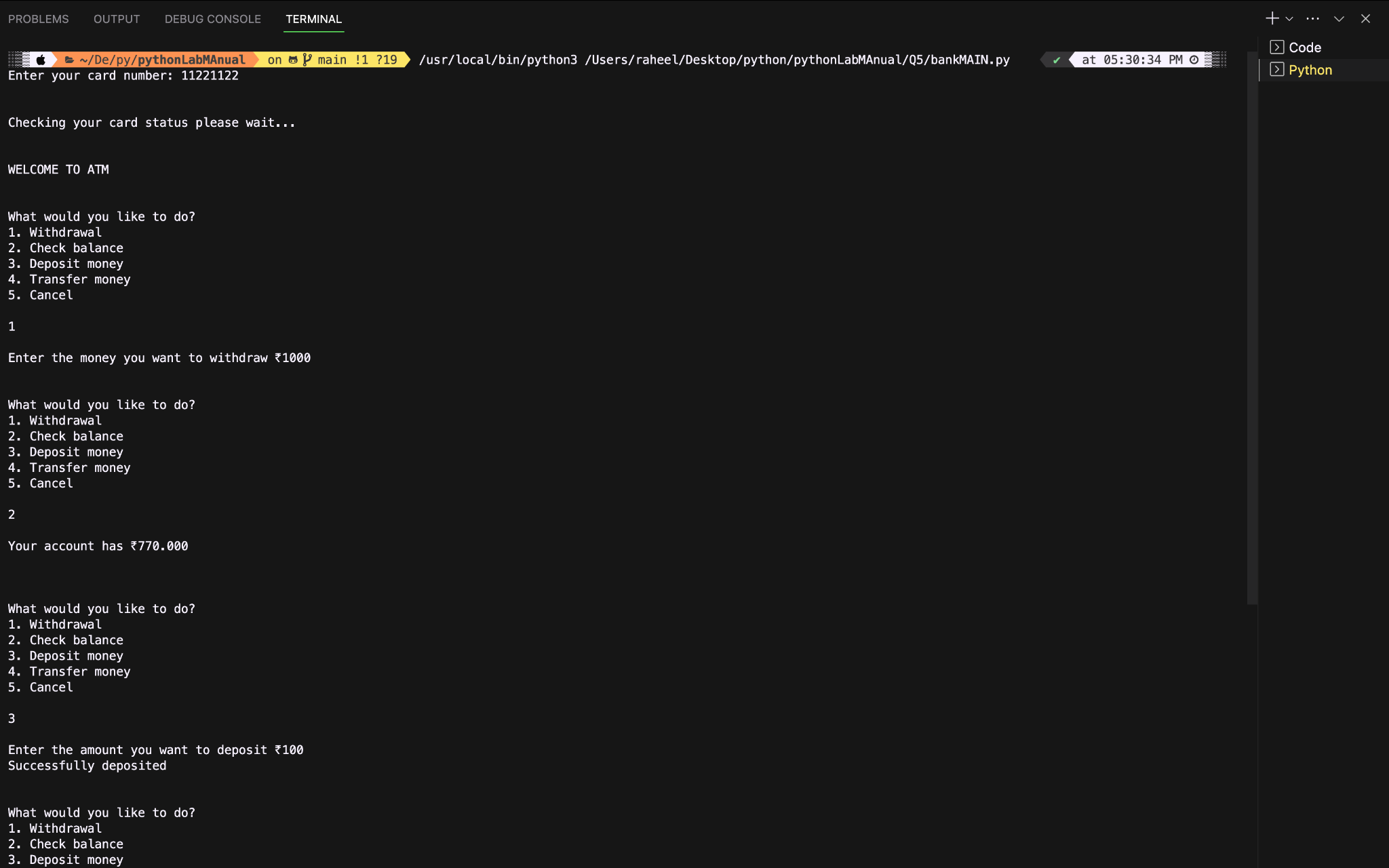
print("\nTHANK YOU...\n\n")

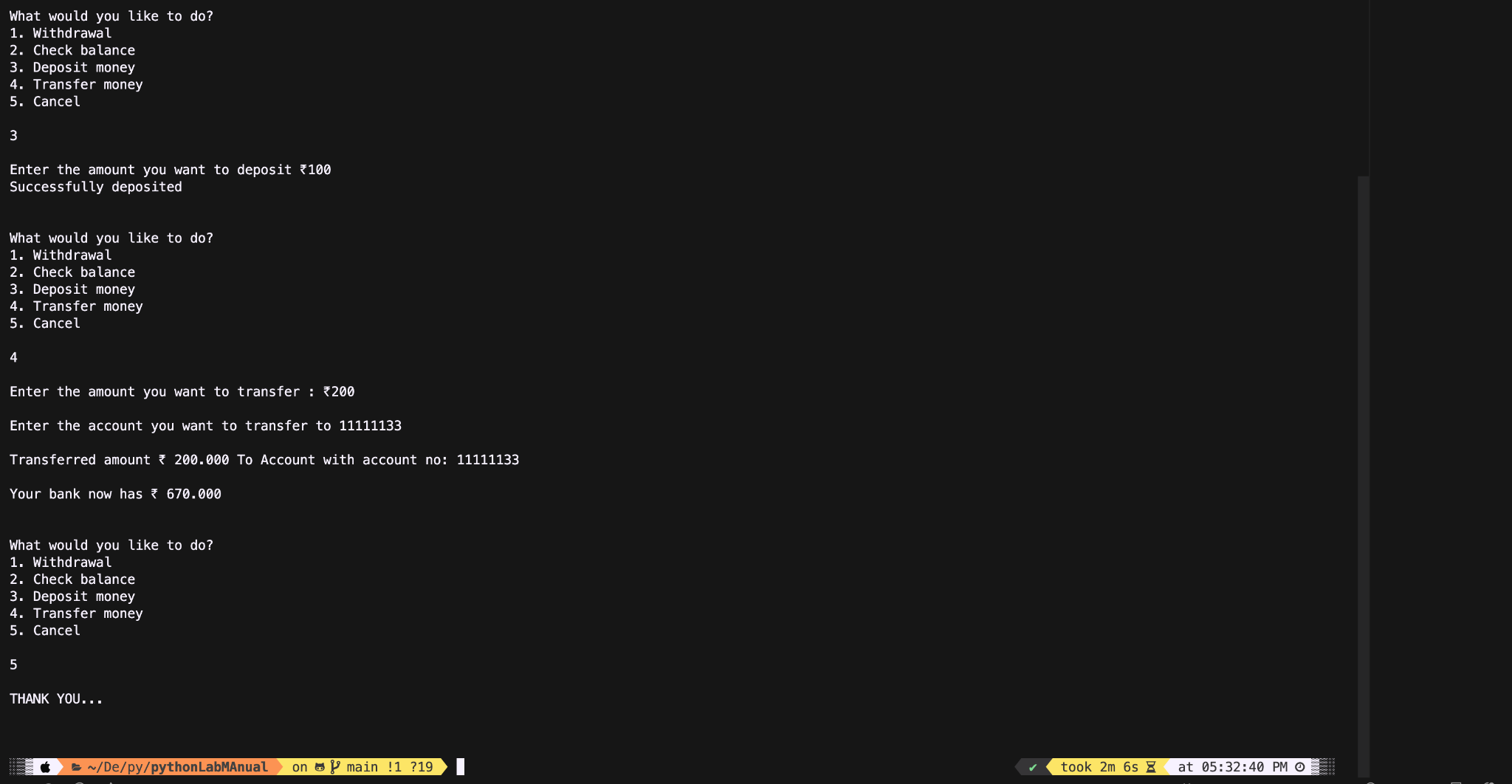
exit()

else:

print("\nInvalid option.\n")

**Output:(screenshot):**





**Conclusion:**

**This code imports from bank.py and variable c has the amount of cash which, in current case is generated at random initially.**

**Name of Student: Raheel Kotwal**

**Roll Number: 45**

**Experiment Number: 4.3**

**Title:**  Car Package

**Theory:** The code below demonstrates the usage of packages in python. Three files: audi.py, bmw.py and nissan.py, which make a package named as cars. These are then used in carsMAIN.py

**Code:**

**carsMAIN.py**

**from cars.bmw import BMW**

**from cars.audi import Audi**

**from cars.nissan import Nissan**

**bmw\_car = BMW(*model*="X5")**

**bmw\_car.start\_engine()**

**bmw\_car.drive()**

**print()**

**audi\_car = Audi(*model*="A4")**

**audi\_car.start\_engine()**

**audi\_car.drive()**

**print()**

**nissan\_car = Nissan(*model*="Altima")**

**nissan\_car.start\_engine()**

**nissan\_car.drive()**

**audi.py**

*class* Audi:

*def* \_\_init\_\_(*self*, *model*):

*self*.model = model

*def* start\_engine(*self*):

print(*f*"Audi {*self*.model} engine started.")

*def* drive(*self*):

print(*f*"Driving the Audi {*self*.model}.")

Bmw.py

*class* BMW:

*def* \_\_init\_\_(*self*, *model*):

*self*.model = model

*def* start\_engine(*self*):

print(*f*"BMW {*self*.model} engine started.")

*def* drive(*self*):

print(*f*"Driving the BMW {*self*.model}.")

Nissan.py

*class* Nissan:

*def* \_\_init\_\_(*self*, *model*):

*self*.model = model

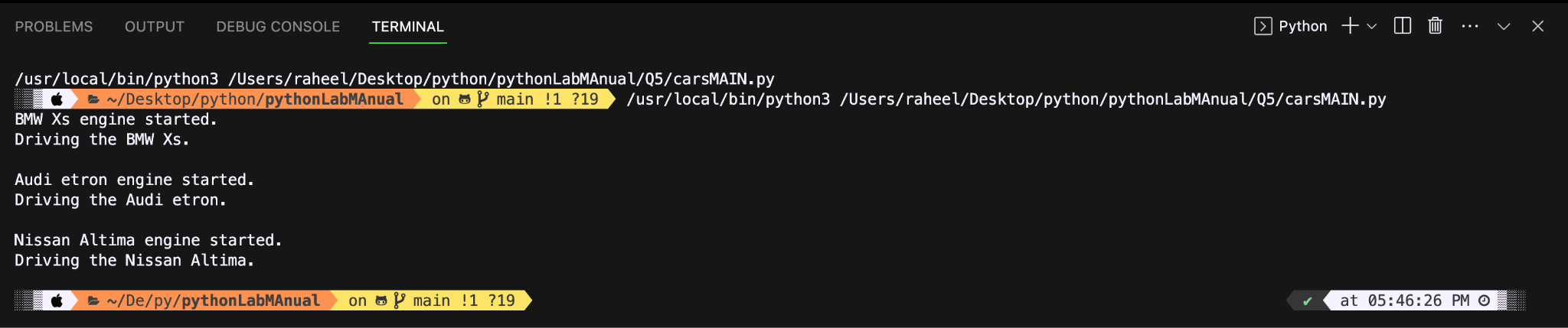
*def* start\_engine(*self*):

print(*f*"Nissan {*self*.model} engine started.")

*def* drive(*self*):

print(*f*"Driving the Nissan {*self*.model}.")

**Output:(screenshot):**



**Conclusion:**

**The code successfully demonstrates usage of packages.**