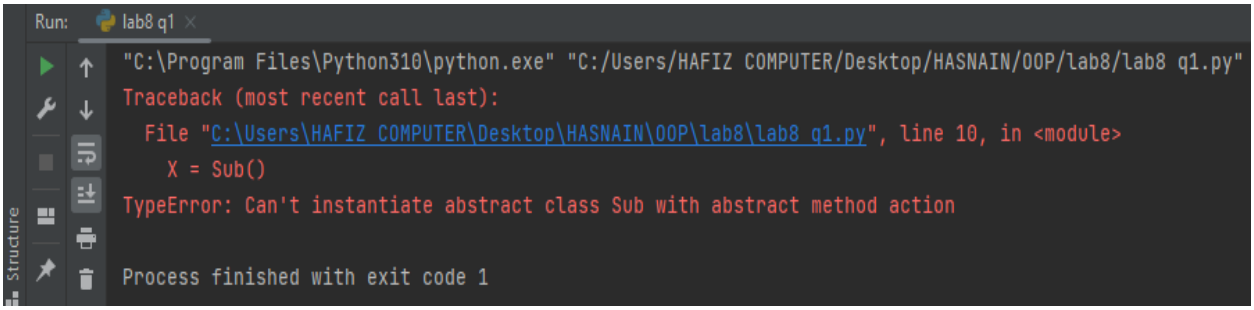
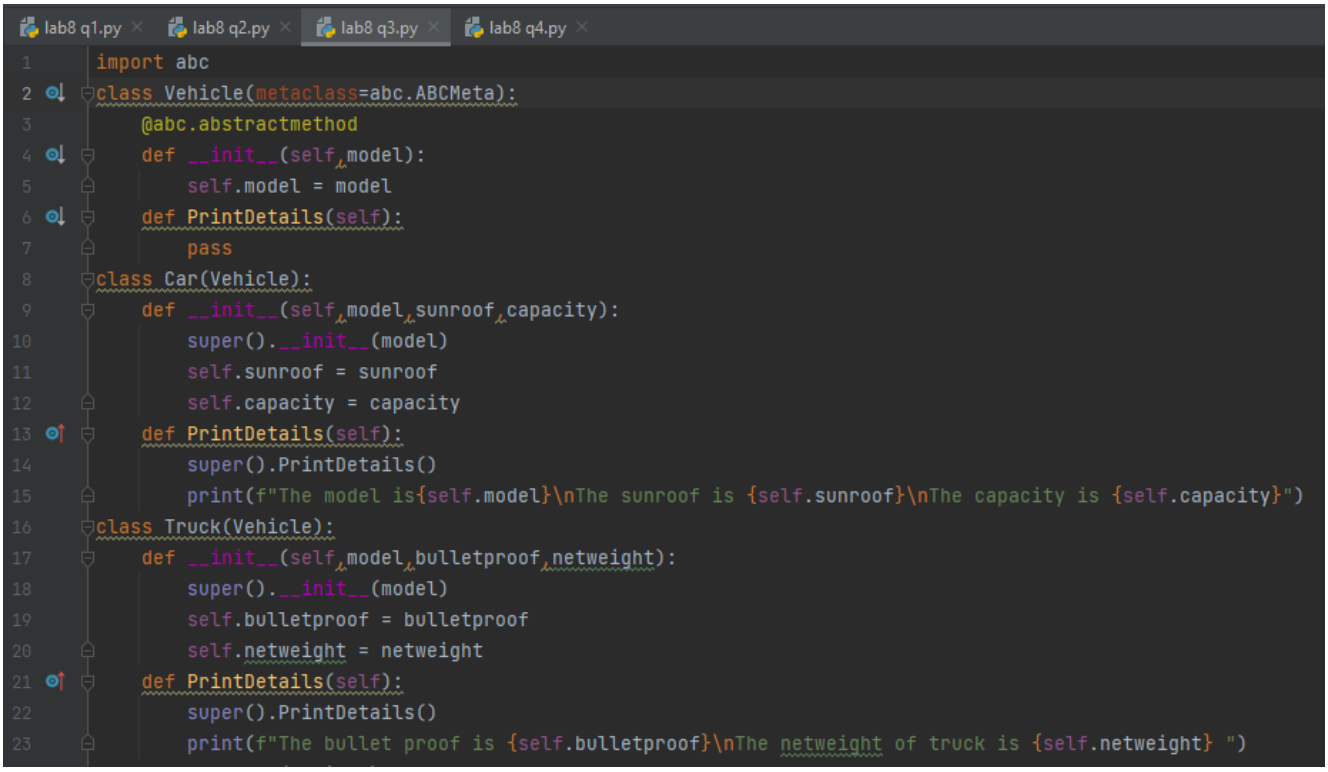

Run:
lab8 q1
"C:\Program Files\Python310\python.exe" "C:/Users/HAFIZ COMPUTER/Desktop/HASNAIN/OOP/lab8/lab8 q1.py"
Traceback (most recent call last):
File "C:\Users\HAFIZ COMPUTER\Desktop\HASNAIN\OOP\lab8\lab8 q1.py", line 10, in <module>
X = Sub()
TypeError: Can't instantiate abstract class Sub with abstract method action
Process finished with exit code 1


lab8 q1.py
lab8 q2.py
from abc import ABCMeta, abstractmethod
class Super(metaclass=ABCMeta):
def delegate(self):
self.action()
@abstractmethod
def action(self):
pass
class Sub(Super):
def action(self):
print("Your are in sub class!!! ")
X=Sub()
X.action()
Super
Run:
lab8 q2
"C:\Program Files\Python310\python.exe" "C:/Users/HAFIZ COMPUTER/Desktop/HASNAIN/OOP/lab8/lab8 q2.py"
Your are in sub class!!!
Process finished with exit code 0


lab8 q1.py
lab8 q2.py
lab8 q3.py
lab8 q4.py
import abc
class Vehicle(metaclass=abc.ABCMeta):
@abc.abstractmethod
def __init__(self,model):
self.model = model
def PrintDetails(self):
pass
class Car(Vehicle):
def __init__(self,model,sunroof,capacity):
super().__init__(model)
self.sunroof = sunroof
self.capacity = capacity
def PrintDetails(self):
super().PrintDetails()
print(f"The model is{self.model}\nThe sunroof is {self.sunroof}\nThe capacity is {self.capacity}")
class Truck(Vehicle):
def __init__(self,model,bulletproof,netweight):
super().__init__(model)
self.bulletproof = bulletproof
self.netweight = netweight
def PrintDetails(self):
super().PrintDetails()
print(f"The bullet proof is {self.bulletproof}\nThe netweight of truck is {self.netweight} ")

lab8 q1.py | lab8 q2.py | lab8 q3.py | lab8 q4.py

```
        self.netweight = netweight
    def PrintDetails(self):
        super().PrintDetails()
        print(f"The bullet proof is {self.bulletproof}\nThe netweight of truck is {self.netweight} ")
class Motorcycle(Vehicle):
    def __init__(self,model,brand,speed):
        super().__init__(model)
        self.brand = brand
        self.speed =speed
    def PrintDetails(self):
        super().PrintDetails()
        print(f"The model of motorcycle is {self.model}\nThe brand of motorcycle is {self.brand}\nThe speed of motorcycle is {self.speed} ")
Unique = Motorcycle("2019","Super Star","70")
Unique.PrintDetails()
```

Vehicle

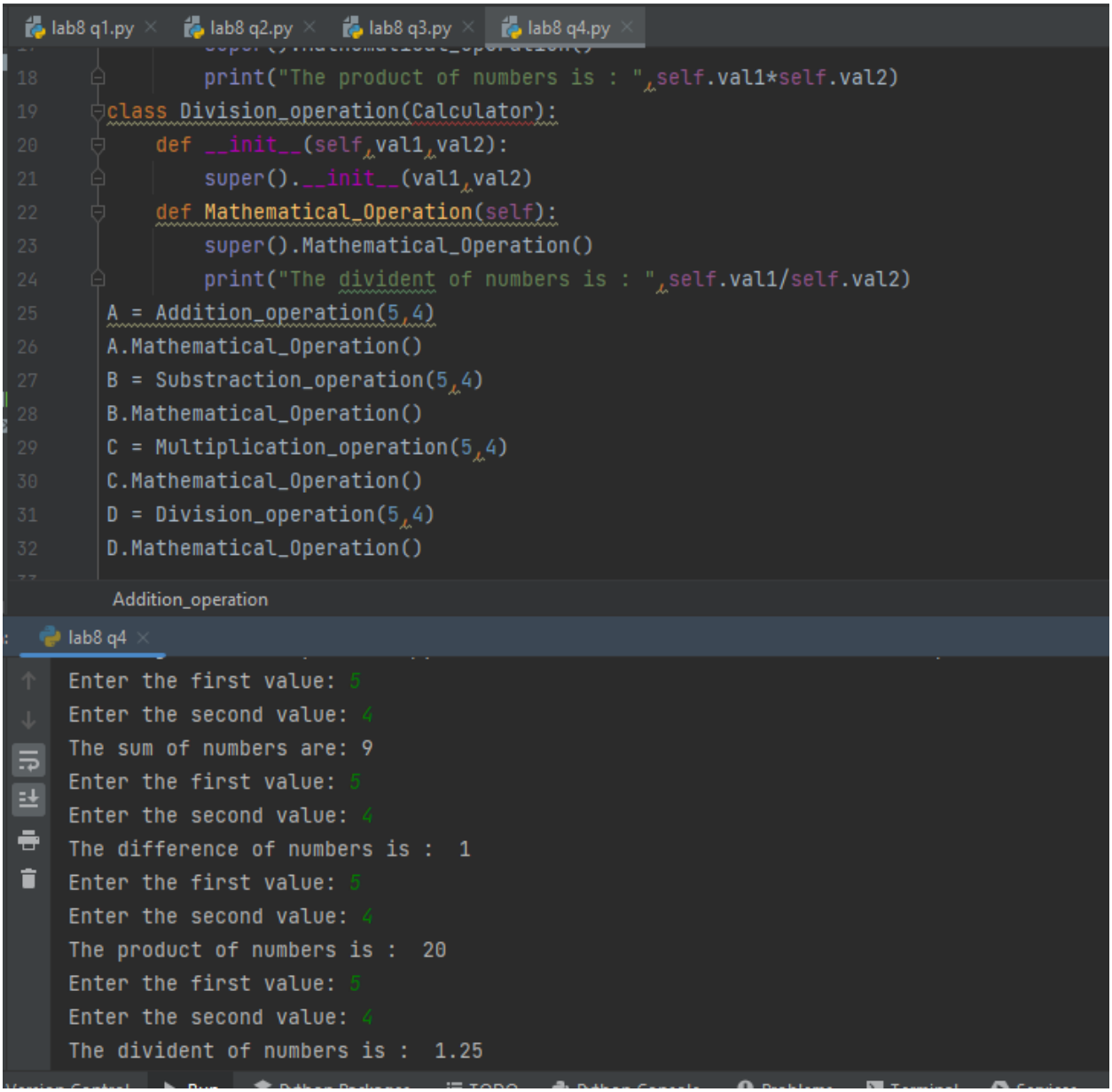
lab8 q3

```
"C:\Program Files\Python310\python.exe" "C:/Users/HAFIZ COMPUTER/Desktop/HASNAIN/OOP/lab8/lab8 q3.py"
The model of motorcycle is 2019
The brand of motorcycle is Super Star
The speed of motorcycle is 70

Process finished with exit code 0
```

lab8 q1.py | lab8 q2.py | lab8 q3.py | lab8 q4.py

```
import abc
class Calculator(metaclass=abc.ABCMeta):
    @abc.abstractmethod
    def __init__(self,val1,val2):
        self.val1 = int(input("Enter the first value: "))
        self.val2 = int(input("Enter the second value: "))
    def Mathematical_Operation(self):
        pass
class Addition_operation(Calculator):
    def __init__(self,val1,val2):
        super().__init__(val1,val2)
    def Mathematical_Operation(self):
        super().Mathematical_Operation()
        print("The sum of numbers are:",self.val1 + self.val2)
class Substraction_operation(Calculator):
    def __init__(self,val1,val2):
        super().__init__(val1,val2)
    def Mathematical_Operation(self):
        super().Mathematical_Operation()
        print("The difference of numbers is : ",self.val1-self.val2)
class Multiplication_operation(Calculator):
    def __init__(self,val1,val2):
        super().__init__(val1,val2)
    def Mathematical_Operation(self):
        super().Mathematical_Operation()
        print("The product of numbers is : ",self.val1*self.val2)
```


lab8 q1.py
lab8 q2.py
lab8 q3.py
lab8 q4.py
print("The product of numbers is : ",self.val1*self.val2)
class Division_operation(Calculator):
def __init__(self,val1,val2):
super().__init__(val1,val2)
def Mathematical_Operation(self):
super().Mathematical_Operation()
print("The divident of numbers is : ",self.val1/self.val2)
A = Addition_operation(5,4)
A.Mathematical_Operation()
B = Substraction_operation(5,4)
B.Mathematical_Operation()
C = Multiplication_operation(5,4)
C.Mathematical_Operation()
D = Division_operation(5,4)
D.Mathematical_Operation()
Addition_operation
lab8 q4
Enter the first value: 5
Enter the second value: 4
The sum of numbers are: 9
Enter the first value: 5
Enter the second value: 4
The difference of numbers is : 1
Enter the first value: 5
Enter the second value: 4
The product of numbers is : 20
Enter the first value: 5
Enter the second value: 4
The divident of numbers is : 1.25