**Assignment\_No:-5**

1. Create a base class BankAccount with methods like deposit() and withdraw(). Derive a class SavingsAccount that overrides the withdraw() method to impose a limit on the withdrawal amount. Write a program that demonstrates the use of overridden methods and proper access modifiers & return the details.

**Code:-**

class BankAccount {

private double balance;

private double remain;

private double amount;

public BankAccount() {

this.balance = balance;

this.remain = remain;

this.amount = amount;

}

public BankAccount(double balance, double amount) {

this.balance = balance;

this.amount = amount;

}

public double deposit() {

System.*out*.println("Total Present amount: " + balance);

System.*out*.println("Entered deposited amount: " + amount);

if (balance > 0) {

balance = balance + amount;

System.*out*.println("Balance After deposited is: " + this.balance);

System.*out*.println("===========================================");

} else {

System.*out*.println("Invalid amount ");

}

return balance;

}

public double withdraw() {

if (amount > 0 && amount <= balance) {

System.*out*.println("Withdraw " + amount);

} else {

System.*out*.println("Invalid Withdraw");

}

return amount;

}

}

class SavingAccount extends BankAccount {

private double balance;

private double amount;

private double remain;

public SavingAccount() {

this.balance = balance;

this.amount = amount;

this.remain = remain;

}

public SavingAccount(double balance, double amount, double remain) {

this.balance = balance;

this.amount = amount;

this.remain = remain;

}

public double withdraw() {

System.*out*.println("Total Balance : " + this.balance);

if (amount > 0 && amount <= balance) {

System.*out*.println("Withdraw " + amount);

} else {

System.*out*.println("Invalid Withdraw");

}

return amount;

}

public void remain1() {

double remain = balance - amount;

System.*out*.println("remaining balance: " + this.remain);

}

}

public class Ex\_1 {

public static void main(String[] args) {

System.*out*.println("Account Details");

SavingAccount sc = new SavingAccount(150000, 100000, 50000);

BankAccount bc = new BankAccount(15000, 5000);

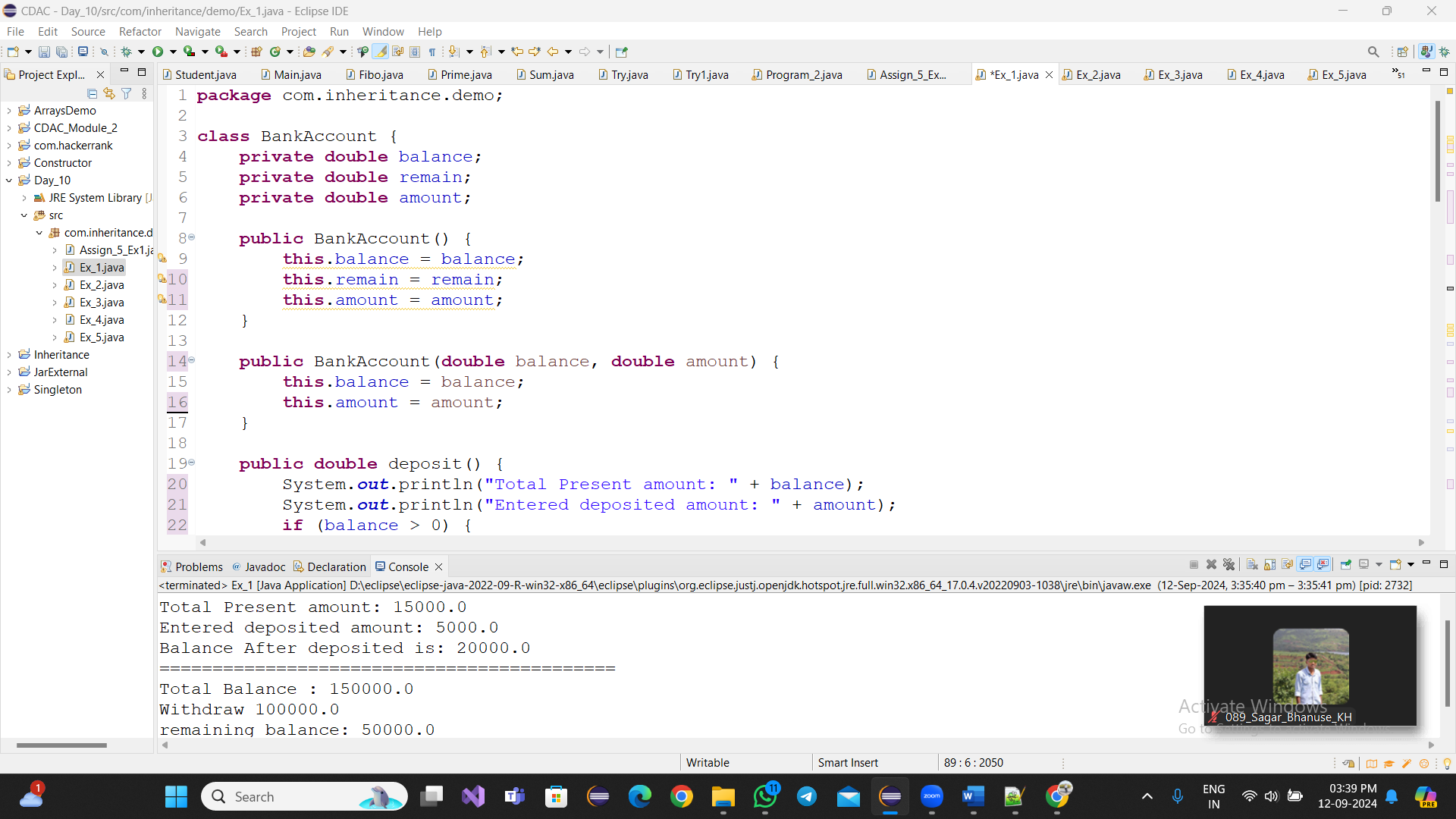
bc.deposit();

sc.withdraw();

sc.remain1();

}

}



1. Create a base class Vehicle with attributes like make and year. Provide a constructor in Vehicle to initialize these attributes. Derive a class Car that has an additional attribute model and write a constructor that initializes make, year, and model. Write a program to create a Car object and display its details.

Code:-

**class** Vehicle {

**private** String make;

**private** **int** year;

**public** Vehicle() {

**this**.make = make;

**this**.year = year;

}

**public** Vehicle(String make, **int** year) {

**this**.make = make;

**this**.year = year;

}

**public** **void** dislay() {

System.***out***.println("make by : " + **this**.make);

System.***out***.println("year : " + **this**.year);

}

}

**class** Car **extends** Vehicle {

**private** String make;

**private** **int** year;

**private** String model;

**public** Car() {

**this**.model = model;

}

**public** Car(String make, **int** year, String model) {

**super** (make,year);

**this**.model = model;

}

**public** **void** display() {

System.***out***.println("make by : " + **this**.make);

System.***out***.println("year : " + **this**.year);

System.***out***.println("model : " + **this**.model);

}

}

**public** **class** Ex\_2 {

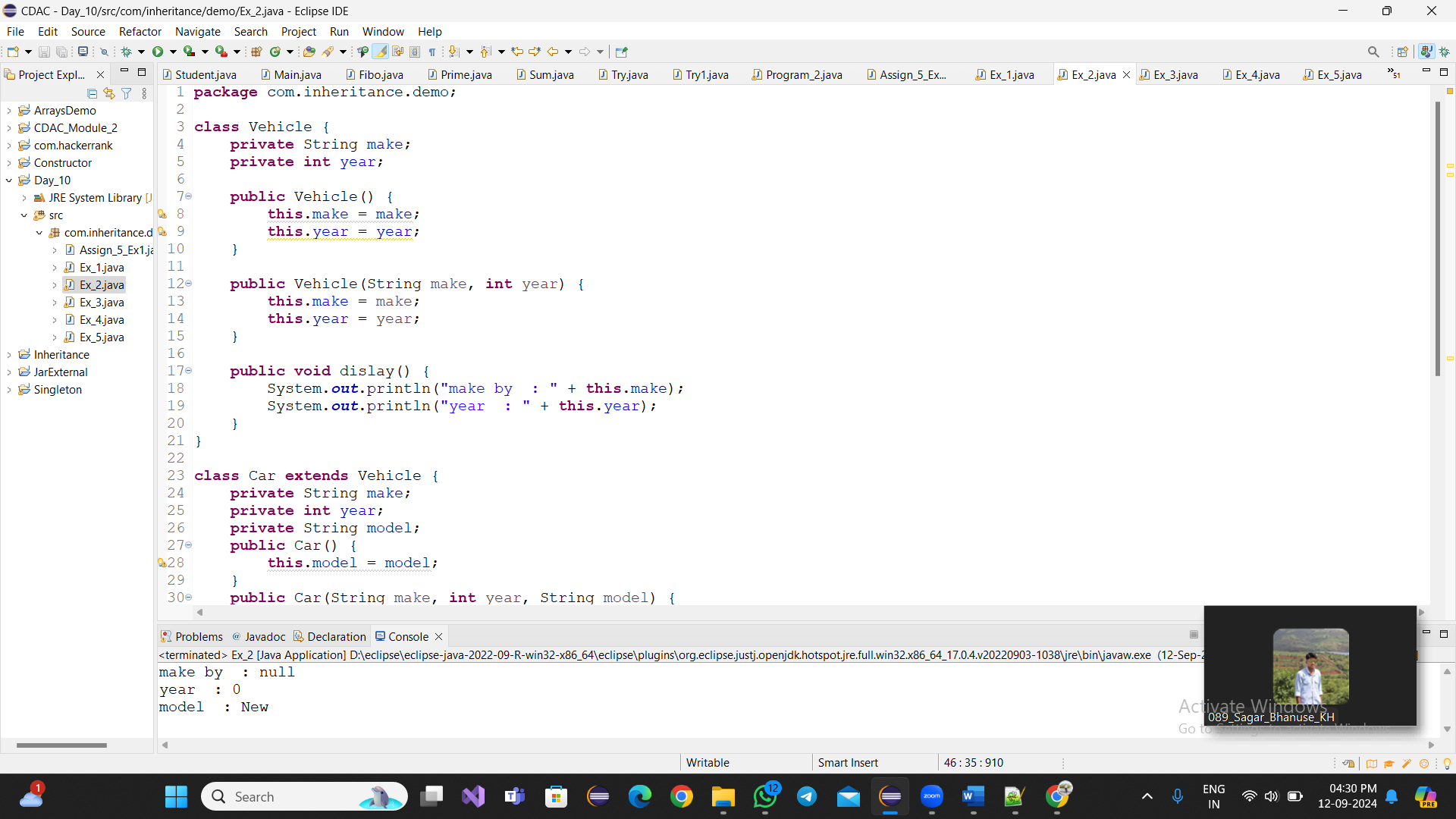
**public** **static** **void** main(String[] args) {

Car cr = **new** Car("TATA", 2011,"New");

cr.dislay();

}

}



1. Create a base class Animal with attributes like name, and methods like eat() and sleep(). Create a subclass Dog that inherits from Animal and has an additional method bark(). Write a program to demonstrate the use of inheritance by creating objects of Animal and Dog and calling their methods.

**Code:-**

**class Animal {**

**private String name;**

**public Animal() {**

**this.name = name;**

**}**

**public Animal(String name) {**

**this.name = name;**

**}**

**public void eat() {**

**System.*out*.println("Name of Animal : " + this.name);**

**System.*out*.println(name + " is...Eating");**

**}**

**public void sleep() {**

**System.*out*.println(this.name + "Sleeping : ");**

**}**

**}**

**class Dog extends Animal {**

**private String name;**

**public Dog(String name) {**

**super(name);**

**}**

**public void bark() {**

**System.*out*.println(name + " is Barking");**

**}**

**}**

**public class Ex\_3 {**

**public static void main(String[] args) {**

**Animal animal = new Animal("Lion");**

**animal.eat();**

**animal.sleep();**

**Dog dg = new Dog("nilu");**

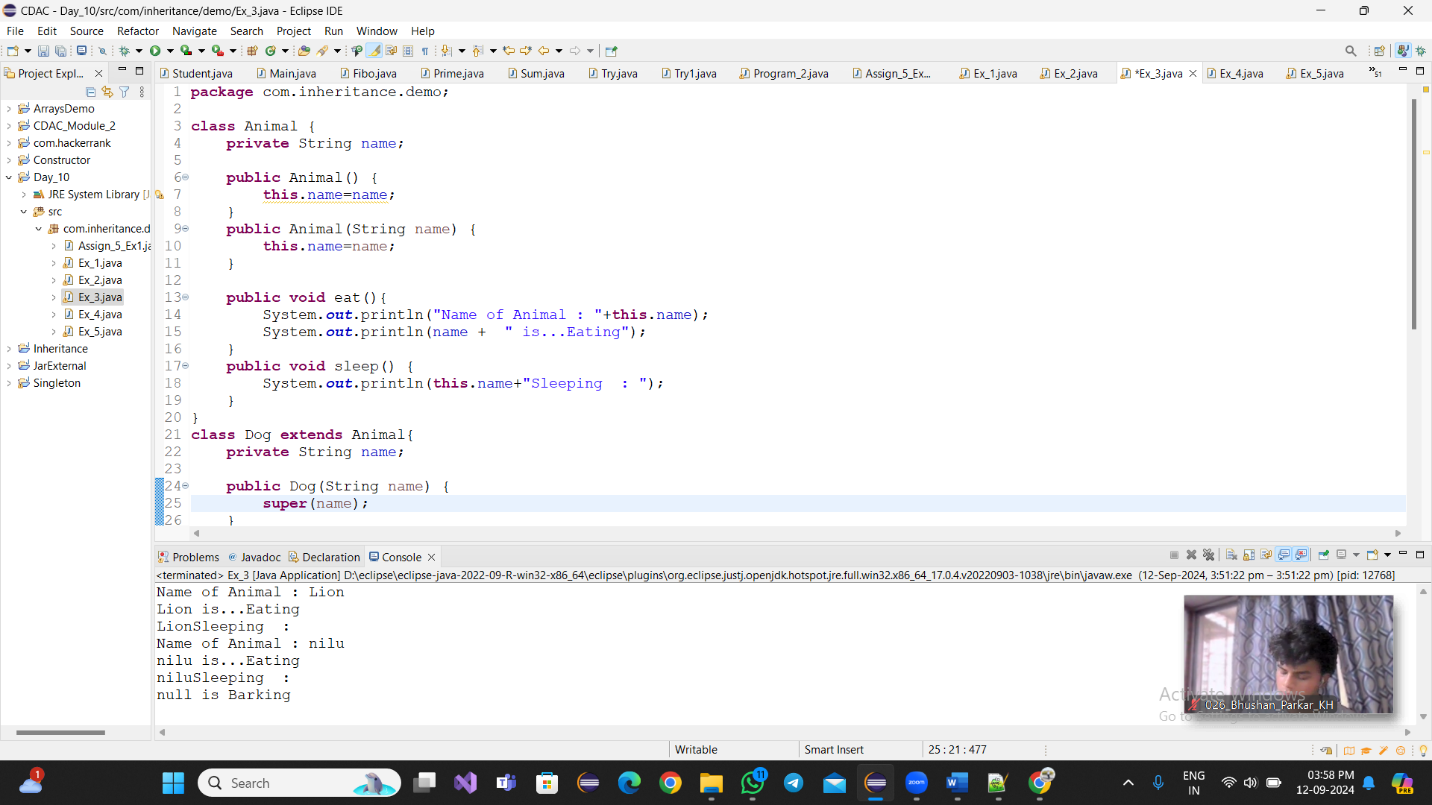
**dg.eat();**

**dg.sleep();**

**dg.bark();**

**}**

**}**

****

1. Build a class Student which contains details about the Student and compile and run its

instance.

**Code:-**

**class Student1 {**

**private String name;**

**private int prn;**

**private String address;**

**private String college;**

**public Student1(String name,int prn,String address,String college) {**

**this.name=name;**

**this.prn=prn;**

**this.address=address;**

**this.college=college;**

**}**

**public void details() {**

**System.*out*.println("Name : "+this.name);**

**System.*out*.println("PRN : "+this.prn);**

**System.*out*.println("Name : "+this.address);**

**System.*out*.println("Name : "+this.college);**

**}**

**}**

**public class Ex\_4 {**

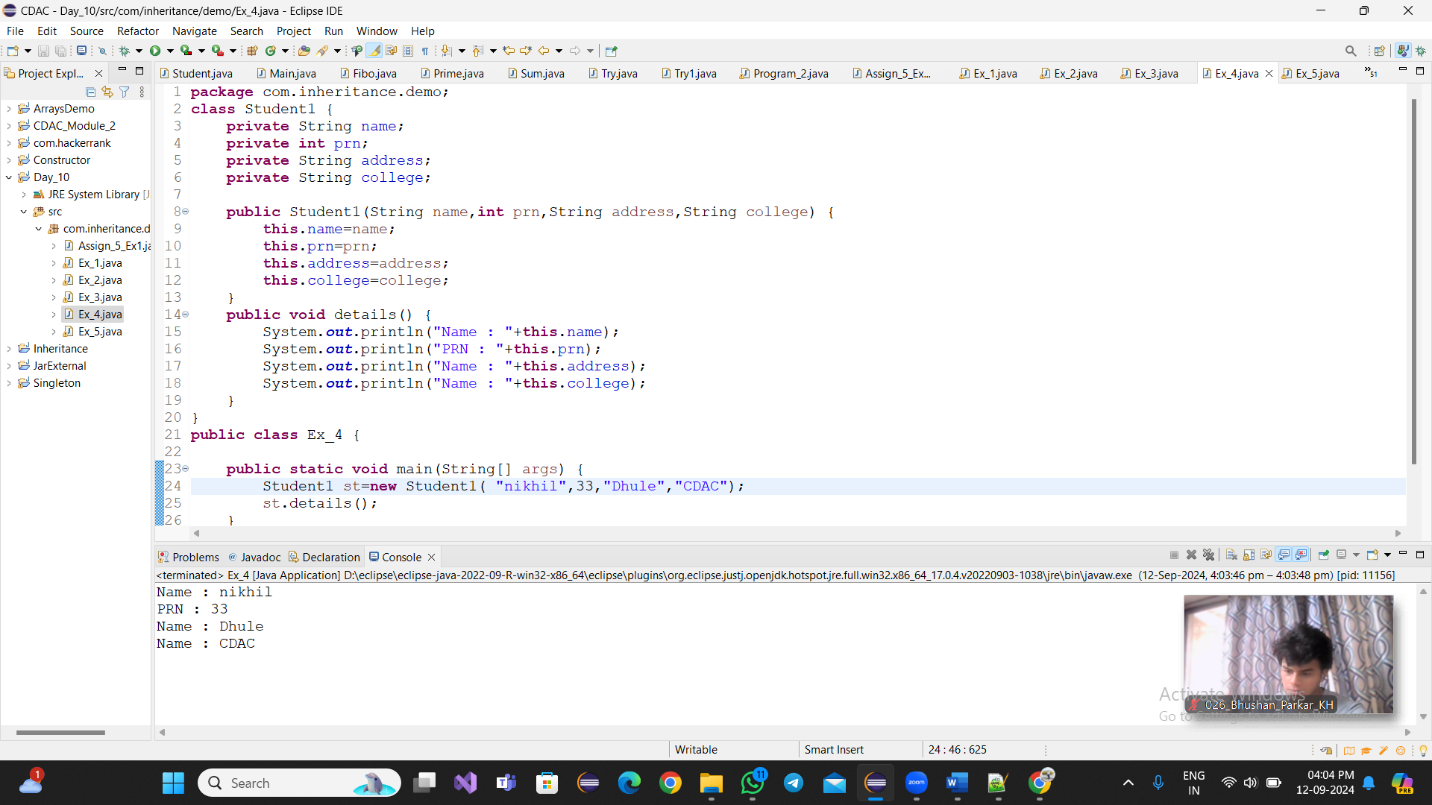
**public static void main(String[] args) {**

**Student1 st=new Student1( "nikhil",33,"Dhule","CDAC");**

**st.details();**

**}**

**}**

****

1. Write a Java program to create a base class Vehicle with methods startEngine() and stopEngine(). Create two subclasses Car and Motorcycle. Override the startEngine() and stopEngine() methods in each subclass to start and stop the engines differently.

**Code:-**

**class Vehicle1{**

**private String name;**

**public Vehicle1() {**

**this.name=name;**

**}**

**public void startEngine() {**

**System.*out*.println("Engine is Start");**

**}**

**public void stopEngine() {**

**System.*out*.println("Engine is Stop");**

**}**

**}**

**class Car1 extends Vehicle{**

**private String name;**

**public Car1(String name) {**

**this.name=name;**

**}**

**public void startEngine() {**

**System.*out*.println("Name of Car : "+this.name);**

**System.*out*.println("Car Engine is Start");**

**}**

**public void stopEngine() {**

**System.*out*.println("car Engine is Stop");**

**}**

**}**

**class Motorcycle extends Vehicle{**

**private String name;**

**public Motorcycle(String name) {**

**this.name=name;**

**}**

**public void startEngine() {**

**System.*out*.println("Name of Motorcycle : "+this.name);**

**System.*out*.println("Motorcycle Engine is Start");**

**}**

**public void stopEngine() {**

**System.*out*.println("Motorcycle Engine is Stop");**

**System.*out*.println("=========================");**

**}**

**}**

**public class Ex\_5 {**

**public static void main(String[] args) {**

**Motorcycle mcy=new Motorcycle("Royal Enfield");**

**mcy.startEngine();**

**mcy.stopEngine();**

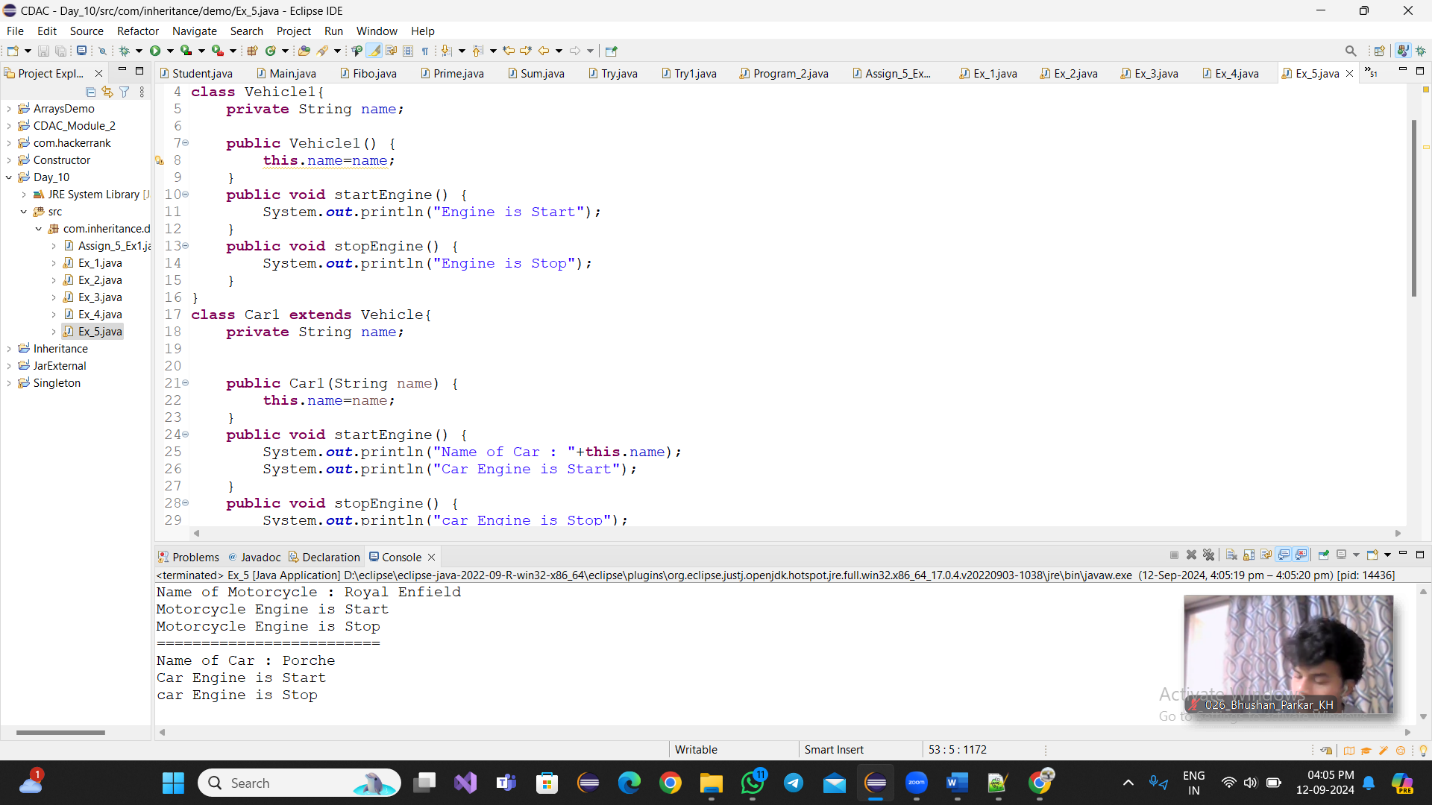
**Car1 cr=new Car1("Porche");**

**cr.startEngine();**

**cr.stopEngine();**

**}**

**}**

****