**1. Create a superclass Person with attributes name and age, and a method display(). Create a subclass Student that adds an attribute studentID. Write a program to create a Student object and display all its attributes.**

**CODE**

**package** pack;

**class** Person {

String name;

**int** age;

**public** Person(String name, **int** age) {

**this**.name = name;

**this**.age = age;

}

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

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

System.***out***.println("Age: " + age);

}

}

**class** Student **extends** Person {

**int** studentID;

**public** Student(String name, **int** age, **int** studentID) {

**super**(name, age);

**this**.studentID = studentID;

}

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

**super**.display();

System.***out***.println("Student ID: " + studentID);

}

}

**public** **class** pan {

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

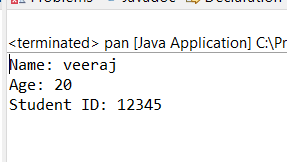
Student student1 = **new** Student("veeraj", 20, 12345);

student1.display();

}

}

**OUTPUT**



**2. Create a superclass Calculator with a method add(int a, int b). Create a subclass AdvancedCalculator that overloads the add method to handle three integers.**

**CODE**

**package** calc;

**class** Calculators

{

**int** add(**int** a, **int** b)

{

**return** a+b;

}

}

**class** Advancedcalcualtor **extends** Calculators

{

**int** add(**int** a,**int** b,**int** c)

{

**return** a+b+c;

}

}

**public** **class** clap {

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

Advancedcalcualtor calculator=**new** Advancedcalcualtor();

System.***out***.println("Sum of two integers:"+ calculator.add(2,3));

System.***out***.println("Sum of three integers:"+calculator.add(2,3,4));

}

}

**OUTPUT**



**3.Create a superclass Vehicle with a method move(). Create subclasses Car and Bike that inherit from Vehicle. Write a program to create objects of Car and Bike and call the move() method on each**.

**CODE**

**package** veer;

**class** Vehicle {

**public** **void** move() {

System.***out***.println("Vehicle is my.");

}

}

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

**public** **void** move() {

System.***out***.println("car is my.");

}

**class** Bike **extends** Vehicle {

**public** **void** move() {

System.***out***.println("Bike is my.");

}

**public** **class** veeraj {

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

Car car = **new** Car();

car.move();

Bike bike = **new** Bike();

bike.move();

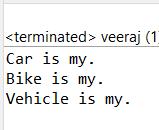
Vehicle Vechile = **new** Vehicle();

Vechile.move();

}

}

**OUTPUT**



**4. Create an class Employee with an abstract method calculatePay(). Create subclasses SalariedEmployee and HourlyEmployee that implement the calculatePay() method. Write a program to create objects of both subclasses and call the calculatePay() method.**

**CODE**

**package** ja;

**abstract** **class** Employee {

**public** **abstract** **void** calculatePay();

}

**class** SalariedEmployee **extends** Employee {

**private** String name;

**private** **double** annualSalary;

**public** SalariedEmployee(String name, **double** annualSalary) {

**this**.name = name;

**this**.annualSalary = annualSalary;

}

@Override

**public** **void** calculatePay() {

**double** monthlyPay = annualSalary / 12;

System.***out***.println(name + "'s monthly pay is:" + String.*format*("%.2f", monthlyPay));

}

}

**class** HourlyEmployee **extends** Employee {

**private** String name;

**private** **double** hourlyRate;

**private** **int** hoursWorked;

**public** HourlyEmployee(String name, **double** hourlyRate, **int** hoursWorked) {

**this**.name = name;

**this**.hourlyRate = hourlyRate;

**this**.hoursWorked = hoursWorked;

}

@Override

**public** **void** calculatePay() {

**double** totalPay = hourlyRate \* hoursWorked;

System.***out***.println(name + "'s total pay is: " + String.*format*("%.2f", totalPay));

}

}

**public** **class** cam {

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

SalariedEmployee salariedEmployee = **new** SalariedEmployee("Veeraj", 72000);

HourlyEmployee hourlyEmployee = **new** HourlyEmployee("Raj", 25, 160);

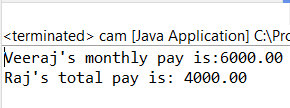
salariedEmployee.calculatePay();

hourlyEmployee.calculatePay();

}

}

**OUTPUT**



**5. Create an class Document with an method void open(). Implement subclasses WordDocument, PDFDocument, and SpreadsheetDocument that extend Document and provide implementations for open(). Write a main class to demonstrate opening different types of documents.(implement compile time- polymorphism).**

**CODE**

**package** doc;

**class** Document {

**public** **void** open() {

System.***out***.println("Opening a generic document");

}

}

**class** WordDocument **extends** Document {

**public** **void** open() {

System.***out***.println("Opening a Word document");

}

}

**class** PDFDocument **extends** Document {

**public** **void** open() {

System.***out***.println("Opening a PDF document");

}

}

**class** SpreadsheetDocument **extends** Document {

**public** **void** open() {

System.***out***.println("Opening a Spreadsheet document");

}

}

**public** **class** veer {

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

Document doc1 = **new** WordDocument();

Document doc2 = **new** PDFDocument();

Document doc3 = **new** SpreadsheetDocument();

doc1.open();

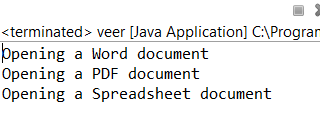
doc2.open();

doc3.open();

}

}

**OUTPUT**



**6.Create a class Calculator with overloaded methods add() that take different numbers and types of parameters: int add(int a, int b) double add(double a, double b)**

**CODE**

**package** clams;

**public** **class** Calculator {

**public** **int** add(**int** a, **int** b) {

**return** a + b;

}

**public** **double** add(**double** a, **double** b) {

**return** a + b;

}

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

Calculator calculator = **new** Calculator();

**int** intResult = calculator.add(2, 10);

System.***out***.println("Sum of integers: " + intResult);

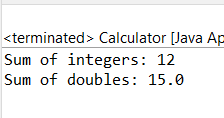
**double** doubleResult = calculator.add(4.5, 10.5);

System.***out***.println("Sum of doubles: " + doubleResult);

}

}

**OUTPUT**



**7. Create a JavaBean class Person with properties firstName, lastName, age, and email. Implement the required no-argument constructor, getter and setter methods for each property. Write a main class to create an instance of Person, set its properties, and print them out.**

**CODE**

**package** det;

**public** **class** Person {

**private** String firstName;

**private** String lastName;

**private** **int** age;

**private** String email;

**public** Person() {

}

**public** String getFirstName() {

**return** firstName;

}

**public** **void** setFirstName(String firstName) {

**this**.firstName = firstName;

}

**public** String getLastName() {

**return** lastName;

}

**public** **void** setLastName(String lastName) {

**this**.lastName = lastName;

}

**public** **int** getAge() {

**return** age;

}

**public** **void** setAge(**int** age) {

**this**.age = age;

}

**public** String getEmail() {

**return** email;

}

**public** **void** setEmail(String email) {

**this**.email = email;

}

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

Person person = **new** Person();

person.setFirstName("Veeraj");

person.setLastName("Salain");

person.setAge(20);

[person.setEmail("veerajSalain@gmail.com](mailto:person.setEmail("veerajSalain@gmail.com)");

System.***out***.println("First Name: " + person.getFirstName());

System.***out***.println("Last Name: " + person.getLastName());

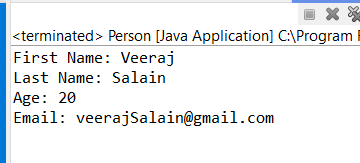
System.***out***.println("Age: " + person.getAge());

System.***out***.println("Email: " + person.getEmail());

}

}

**OUTPUT**



**8.Create a JavaBean class Car with properties make, model, year, and color. Implement the required no-argument constructor, getter and setter methods for each property. Write a main class to create an instance of Car, set its properties, and print the car details.**

**CODE**

**package** news;

**public** **class** Car {

**private** String make;

**private** String model;

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

**private** String color;

**public** Car() {

}

**public** String getMake() {

**return** make;

}

**public** **void** setMake(String make) {

**this**.make = make;

}

**public** String getModel() {

**return** model;

}

**public** **void** setModel(String model) {

**this**.model = model;

}

**public** **int** getYear() {

**return** year;

}

**public** **void** setYear(**int** year) {

**this**.year = year;

}

**public** String getColor() {

**return** color;

}

**public** **void** setColor(String color) {

**this**.color = color;

}

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

Car car = **new** Car();

car.setMake("RR");

car.setModel("R1");

car.setYear(2022);

car.setColor("Black");

System.***out***.println("Make: " + car.getMake());

System.***out***.println("Model: " + car.getModel());

System.***out***.println("Year: " + car.getYear());

System.***out***.println("Color: " + car.getColor());

}

}

**OUTPUT**

