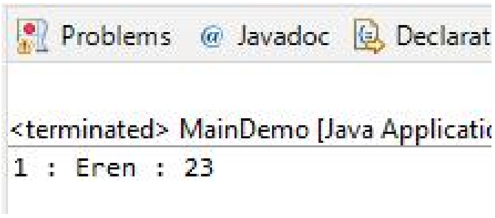
1) **Create a superclass Person with attributes name and age, and a method display(). Create a subclass Student that adds an attribute studentID. Write a program to create a Student object and display all its attributes.**

Program :-

```
package package_demo;      // package
import java.util.*;        //importing java.util package

class Person {             //parent class
	String name;
	int age;
	void display() {
		System.out.print(name + " : " + age);
	}
}
class Student extends Person {          //child class
	int studentID;
	Student(int studentID, String name, int age) {
		this.studentID = studentID;
		this.name = name;
		this.age = age;
	}
	@Override
	void display() {
		System.out.print(studentID + " : " + name + " : " + age);
	}
}

public class MainDemo {
	public static void main(String []args) {
		Student obj = new Student(1, "Eren", 23);
		obj.display();      //calling method to print studentID : name : age
	}
}
```

Output :-



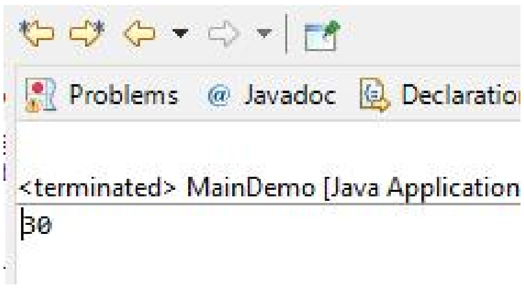
```
<terminated> MainDemo [Java Applicati
1 : Eren : 23
```

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.**

Program :-

```
package package_demo;      // package
import java.util.*;        //importing java.util package

class Calculator {                //parent class
	int add(int a, int b) {
		return a+b;
	}
}
class AdvancedCalculator extends Calculator { //child class
	int add(int a, int b, int c) {          //overloading
		return a+b+c;
	}
}
public class MainDemo {
	public static void main(String []args) {
		AdvancedCalculator obj = new AdvancedCalculator();
		System.out.println(obj.add(12,8,10));//calling the overloaded method
	}
}
```

Output :-



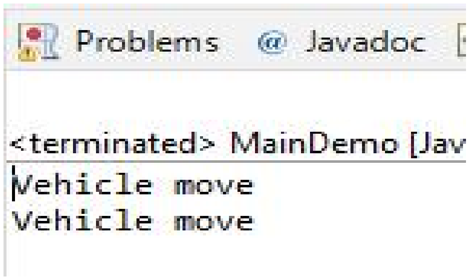
```
<terminated> MainDemo [Java Application
30
```

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.**

Program :-

```
package package_demo;		// package
import java.util.*;		//importing java.util package

class Vehicle {			//parent class
	void move() {
		System.out.println("Vehicle move");
	}
}
class Car extends Vehicle {  }		//child class
class Bike extends Vehicle {  }		//child class

public class MainDemo {
	public static void main(String []args) {
		Car car_obj = new Car();
		Bike bike_obj = new Bike();
		car_obj.move();		//calling method
		bike_obj.move();	//calling method
	}
}
```

Output :-

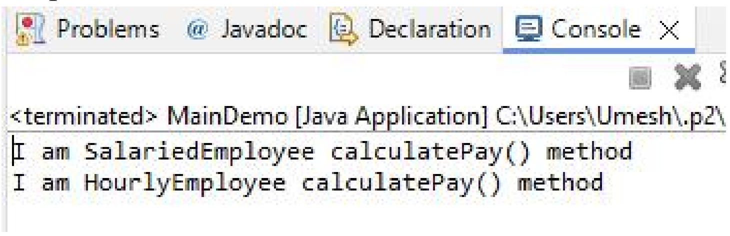


```
<terminated> MainDemo [Jav
Vehicle move
Vehicle move
```

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.**

Program :-

```
package package_demo;     // package

import java.util.*;       //importing java.util package

abstract class Employee {              //parent class
	abstract void calculatePay();
}

class SalariedEmployee extends Employee {     //child class
	@Override
	void calculatePay() {
		System.out.println("I am SalariedEmployee calculatePay() method");
	}
}

class HourlyEmployee extends Employee {//child class
	@Override
	void calculatePay() {
		System.out.println("I am HourlyEmployee calculatePay() method");
	}
}

public class MainDemo {
	public static void main(String []args) {
		SalariedEmployee obj1 = new SalariedEmployee();
		HourlyEmployee obj2 = new HourlyEmployee();
		obj1.calculatePay();
		obj2.calculatePay();
	}
}
```
Output :-



```
<terminated> MainDemo [Java Application] C:\Users\Umesh\.p2\
I am SalariedEmployee calculatePay() method
I am HourlyEmployee calculatePay() method
```

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 <u>complile</u> time-polymorphism).**

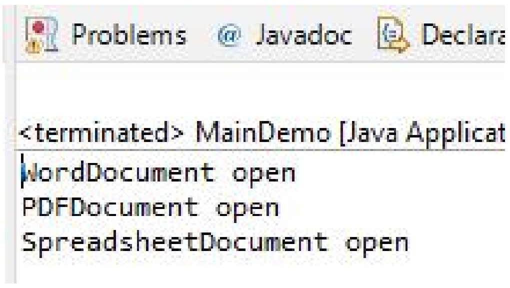
Program :-

package package_demo;        // package
import java.util.*;               //importing java.util package

```
class Document {             //parent class
	void open() {
		System.out.println("Document open");
	}
}
```

```java
class WordDocument extends Document {   //child class
	@Override
	void open() {
		System.out.println("WordDocument open");
	}
}

class PDFDocument extends Document {    //child class
	@Override
	void open() {
		System.out.println("PDFDocument open");
	}
}

class SpreadsheetDocument extends Document {  //child class
	@Override
	void open() {
		System.out.println("SpreadsheetDocument open");
	}
}

public class MainDemo {
	public static void main(String []args) {
		new WordDocument().open();
		new PDFDocument().open();
		new SpreadsheetDocument().open();
	}
}
```

Output :-



```
<terminated> MainDemo [Java Applicat
WordDocument open
PDFDocument open
SpreadsheetDocument open
```

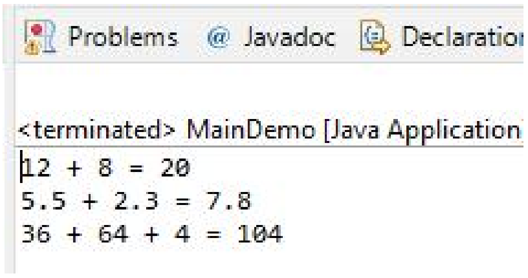
6) **Create a class Calculator with overloaded methods add() that take different numbers and types of parameters: int add(int a, int b), double add(double a, double b), int add(int a, int b, int c) Write a main class to demonstrate the usage of these methods.**

Program :-

```java
package package_demo; // package
import java.util.*; //importing java.util package

class Calculator {
	int add(int a, int b) {
		return a+b;
	}
	double add(double a, double b) { //overloaded method
		return a+b;
	}
	int add(int a, int b, int c) {   //overloaded method
		return a+b+c;
	}
}
```

```
public class MainDemo {
	public static void main(String []args) {
	Calculator obj = new Calculator();
	System.out.println("12 + 8 = "+ obj.add(12, 8));      //calling the method
	System.out.println("5.5 + 2.3 = " + obj.add(5.5,2.3));//calling the method
	System.out.println("36 + 64 + 4 = "+obj.add(36,64,4));//calling the method
	}
}
```

Output :-



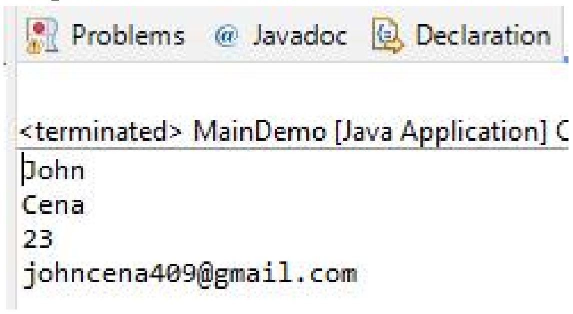
```
<terminated> MainDemo [Java Application
12 + 8 = 20
5.5 + 2.3 = 7.8
36 + 64 + 4 = 104
```

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.**

Program :-

```
package package_demo; // package
import java.util.*; //importing java.util package

class Person implements java.io.Serializable{ //JavaBean class
	private String firstName;             //properties
	private String lastName;
	private int age;
	private String email;
	Person() { }                                  //No-arg constructor
	public void setFirstName(String firstName) {  //setter method
		this.firstName = firstName;
	}
	public void setLastName(String lastName) {    //setter method
		this.lastName = lastName;
	}
	public void setAge(int age) {                 //setter method
		this.age = age;
	}
	public void setEmail(String email) {          //setter method
		this.email = email;
	}
	public String getFirstName() {          //getter method
		return firstName;
	}
	public String getLastName() {           //getter method
		return lastName;
	}
	public int getAge() {                   //getter method
		return age;
	}
	public String getEmail() {              //getter method
		return email;
	}
}

public class MainDemo {
	public static void main(String []args) {
		Person obj = new Person();
		obj.setFirstName("John");		//calling setter methods
		obj.setLastName("Cena");
		obj.setAge(23);
		obj.setEmail("johncena409@gmail.com");
		System.out.println(obj.getFirstName()); //calling getter methods
		System.out.println(obj.getLastName());
		System.out.println(obj.getAge());
		System.out.println(obj.getEmail());
	}
}
```

Output :-



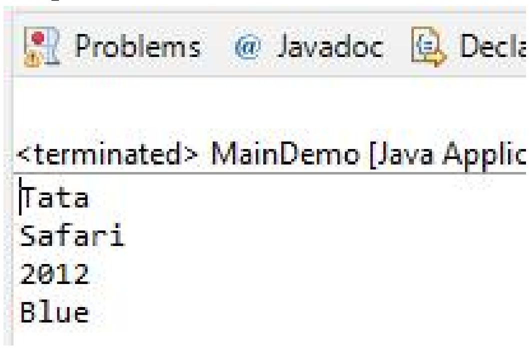
```
<terminated> MainDemo [Java Application] C
John
Cena
23
johncena409@gmail.com
```

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.**

Program :-

```
package package_demo;		// package
import java.util.*;		//importing java.util package

class Car implements java.io.Serializable{ //JavaBean class
	private String make;		//properties
	private String model;
	private int year;
	private String color;
	Car() { }			//No-arg constructor
	public void setMake(String make) {	//setter method
		this.make = make;
	}
	public void setModel(String model) {	//setter method
		this.model = model;
	}
	public void setYear(int year) {		//setter method
		this.year = year;
	}
	public void setColor(String color) {	//setter method
		this.color = color;
	}
	public String getMake() {		//getter method
		return make;
	}
	public String getModel() {		//getter method
		return model;
```

```
        }
        public int getYear() {           //getter method
                return year;
        }
        public String getColor() {       //getter method
                return color;
        }
}

public class MainDemo {
        public static void main(String []args) {
                Car obj = new Car();
                obj.setMake("Tata");                    //calling setter methods
                obj.setModel("Safari");
                obj.setYear(2012);
                obj.setColor("Blue");
                System.out.println(obj.getMake());      //calling getter methods
                System.out.println(obj.getModel());
                System.out.println(obj.getYear());
                System.out.println(obj.getColor());
        }
}
```
Output :-



```
<terminated> MainDemo [Java Applic
Tata
Safari
2012
Blue
```