5. write the classes line, rectangle, cube etc. and make the shape as their base class. Add an abstract draw() method in the class shape and draw all shapes.

abstract class Shape{

abstract void draw();

}

class Line extends Shape{

void draw(){System.out.println("drawing Line");}

}

class Square extends Shape{

void draw(){System.out.println("drawing square");}

}

class Rectangle extends Shape{

void draw(){System.out.println("drawing rectangle");}

}

class Circle extends Shape{

void draw(){System.out.println("drawing circle");}

}

class Paint{

public static void main(String args[]){

Shape s1=new Circle();

Shape s2=new Rectangle();

Shape s3=new Line();

Shape s4=new Square();

s1.draw();

s2.draw();

s3.draw();

s4.draw();

}

}

Output

drawing circle

drawing rectangle

drawing Line

drawing square

6. write an abstract class persistence along with two sub classes.

abstract class Persistence{

abstract void persist();

}

class File Persistence extends Persistence{

@Override

void persist() {

// TODO Auto-generated method stub

System.out.println("Data Saved in File");

}

}

class Database Persistence extends Persistence{

@Override

void persist() {

// TODO Auto-generated method stub

System.out.println("Data saved in Database");

}

}

public class Client extends Persistence{

public static void main(String[] args) {

// TODO Auto-generated method stub

Persistence obj=new Client();

obj.persist();

}

@Override

void persist() {

// TODO Auto-generated method stub

System.out.println("Persist method from client");

}

}

Output

Persist method from client