***dt : 21/10/2022***

***======================================================================***

***Assignment-1:(solution)***

***Convert IArithmetic application into "Anonymous InnerClass as implementation***

***class" model.***

***=>Addition,Subtraction,Multiplication,Division,NodDivision classes as***

***Anonymous***

***IArithmetic.java***

***package test;***

***public interface IArithmetic {***

***public abstract double calculate(int x,int y);***

***}***

***DemoAnonymous4.java(MainClass)***

***package maccess;***

***import test.\*;***

***import java.util.\*;***

***public class DemoAnonymous4 {***

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

***Scanner s = new Scanner(System.in);***

***System.out.println("Enter the value x:");***

***int x = s.nextInt();***

***System.out.println("Enter the value y:");***

***int y = s.nextInt();***

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

***System.out.println("1.add\n2.sub\n3.mul\n4.div\n5.modDiv");***

***System.out.println("Enter the Choice:");***

***switch(s.nextInt())***

***{***

***case 1:***

***//Addition class as Anonymous***

***IArithmetic ad = new IArithmetic()***

***{***

***public double calculate(int x,int y)***

***{***

***return x+y;***

***}***

***};***

***System.out.println("Sum="+ad.calculate(x, y));***

***break;***

***case 2:***

***//Subtraction class as Anonymous***

***IArithmetic sb = new IArithmetic()***

***{***

***public double calculate(int x,int y)***

***{***

***return x-y;***

***}***

***};***

***System.out.println("Sub="+sb.calculate(x, y));***

***break;***

***case 3:***

***//Multiplication class as Anonymous***

***IArithmetic ml = new IArithmetic()***

***{***

***public double calculate(int x,int y)***

***{***

***return x\*y;***

***}***

***};***

***System.out.println("Mul="+ml.calculate(x, y));***

***break;***

***case 4:***

***//Division class as Anonymous***

***IArithmetic dv = new IArithmetic()***

***{***

***public double calculate(int x,int y)***

***{***

***return (float)x/y;***

***}***

***};***

***System.out.println("Div="+dv.calculate(x, y));***

***break;***

***case 5:***

***//ModDivision class as Anonymous***

***IArithmetic md = new IArithmetic()***

***{***

***public double calculate(int x,int y)***

***{***

***return x%y;***

***}***

***};***

***System.out.println("modDiv="+md.calculate(x, y));***

***break;***

***default:***

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

***}//end of switch***

***s.close();***

***}***

***}***

***-----------------------------------------------------------------------***

***ClassFiles:***

***IArithmetic.class***

***DemoAnonymous4.class(MainClass)***

***DemoAnonymous4$1.class***

***DemoAnonymous4$2.class***

***DemoAnonymous4$3.class***

***DemoAnonymous4$4.class***

***DemoAnonymous4$5.class***

***====================================================================***

***Assignment-2:***

***Convert BankTransaction application into "Anonymous InnerClass as***

***implementation class" model.***

***=>WithDraw and Deposit classes as Anonymous***

***Balance.java***

***package test;***

***public class Balance {***

***public double bal=2000;***

***public double getBalance() {***

***return bal;***

***}***

***}***

***Transaction.java***

***package test;***

***public interface Transaction {***

***public static final Balance b=new Balance();***

***public void process(int amt);***

***}***

***CheckPinNo.java***

***package test;***

***public class CheckPinNo {***

***public boolean verify(int pinNo) {***

***return switch(pinNo) {***

***case 1111 : yield true;***

***case 2222 : yield true;***

***case 3333 : yield true;***

***default : yield false;***

***};***

***}***

***}***

***BankMainClass.java(MainClass)***

***package maccess;***

***import test.\*;***

***import java.util.\*;***

***public class BankMainClass {***

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

***Scanner s = new Scanner(System.in);***

***int count=0;***

***abc:***

***while(true) {***

***System.out.println("Enter the PinNo:");***

***int pinNo = s.nextInt();***

***CheckPinNo cpn = new CheckPinNo();***

***boolean k = cpn.verify(pinNo);***

***if(k)***

***{***

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

***System.out.println("1.WithDraw\n2.Deposit");***

***System.out.println("Enter the Choice:");***

***switch(s.nextInt())***

***{***

***case 1:***

***System.out.println("Enter the amt:");***

***int a1 = s.nextInt();***

***if(a1>0 && a1%100==0)***

***{***

***//WithDraw class as Anonymous***

***Transaction wd = new Transaction()***

***{***

***public void process(int amt)***

***{***

***if(amt<=b.bal)***

***{***

***System.out.println("Amt WithDrawn:"+amt);***

***b.bal=b.bal-amt;***

***System.out.println("Balance amt:"+b.getBalance());***

***System.out.println("Transaction Completed...");***

***}//end of if***

***else***

***{***

***System.out.println("Insufficient fund...");***

***}***

***}***

***};***

***wd.process(a1);***

***}//end of if***

***else***

***{***

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

***}***

***break abc;//stop the loop***

***case 2:***

***System.out.println("Enter the amt:");***

***int a2 = s.nextInt();***

***if(a2>0 && a2%100==0)***

***{***

***//Deposit class as Anonymous***

***Transaction dp = new Transaction()***

***{***

***public void process(int amt)***

***{***

***System.out.println("Amt deposited:"+amt);***

***b.bal=b.bal+amt;***

***System.out.println("Balance amt:"+b.getBalance());***

***System.out.println("Transaction Completed...");***

***}***

***};***

***dp.process(a2);***

***}//end of if***

***else***

***{***

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

***}***

***break abc;//stop the loop***

***default:***

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

***break abc;***

***}//end of switch***

***}//end of if***

***else***

***{***

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

***count++;***

***}***

***if(count==3)***

***{***

***System.out.println("Transaction blocked...");***

***break;//stop the loop***

***}***

***}//end of loop***

***}***

***}***

***--------------------------------------------------------***

***ClassFiles:***

***Balance.class***

***Transaction.class***

***CheckPinNo.class***

***BankMainClass.class(MainClass)***

***BankMainClass$1.class***

***BankMainClass$2.class***

***==========================================================***

***Note:***

***=>"Anonymous InnerClass as Implementation class" model is modified as***

***"LambdaExpression" in Java8 version.***

***==========================================================***

***\*imp***

***LambdaExpressions in Java:(Java8 - new feature)***

***=>The process of declaring method without method\_name is known as***

***"Lambda Expression",which is also known as Anonymous method.***

***structure of LambdaExpression:***

***(para\_list)->***

***{***

***//method\_body***

***}***

***Note:***

***=>LambdaExpression is attached with the abstract method of Interface***

***and,LambdaExpression is executed by calling Interface abstarct method.***

***syntax:***

***interface ITest***

***{***

***public abstract void m1(int x);***

***}***

***ITest ob = (int x)->***

***{***

***//method\_body***

***};***

***Ex:***

***ITest.java***

***package test;***

***public interface ITest {***

***public abstract void m1(int x);***

***public default void m2(int y) {***

***System.out.println("====default m2(y)====");***

***System.out.println("The value y:"+y);***

***}***

***}***

***LambdaExpression1.java(MainClass)***

***package maccess;***

***import test.\*;***

***public class LambdaExpression1 {***

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

***//LambdaExpression***

***ITest ob = (int x)->***

***{***

***System.out.println("====method m1(x)====");***

***System.out.println("The value x:"+x);***

***};***

***ob.m1(11);***

***ob.m2(12);***

***}***

***}***

***o/p:***

***====method m1(x)====***

***The value x:11***

***====default m2(y)====***

***The value y:12***

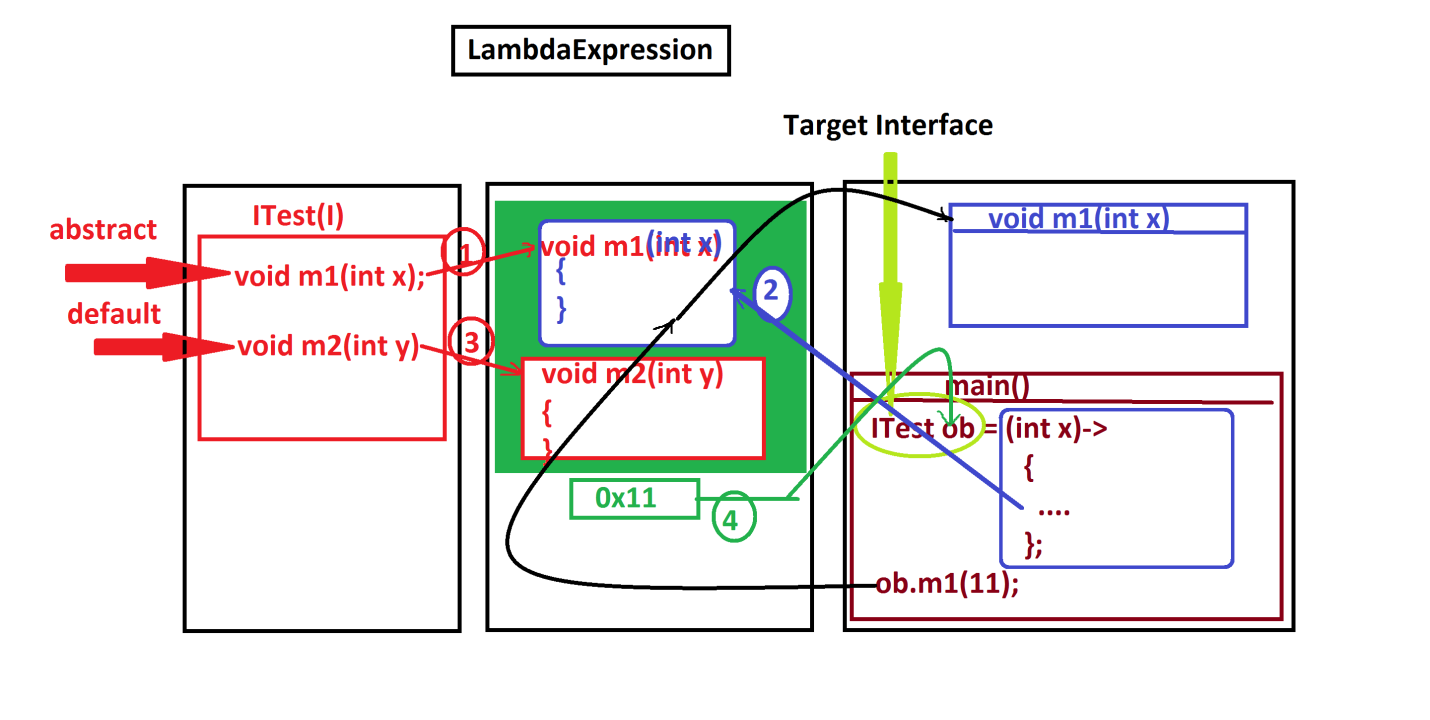
***--------------------------------------------------------***

***Execution flow of above program:***

***ClassFiles:***

***ITest.class***

***LambdaExpression1.class(MainClass)***

******

***===================================================================***

***Ex-program:***

***Convert IComparable application into LambdaExpression:***

***IComparable.java***

***package test;***

***public interface IComparable {***

***public abstract int compare(int x,int y);***

***}***

***LambdaExpression2.java(MainClass)***

***package maccess;***

***import test.\*;***

***import java.util.\*;***

***public class LambdaExpression2 {***

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

***Scanner s = new Scanner(System.in);***

***System.out.println("Enter the value of x:");***

***int x = s.nextInt();***

***System.out.println("Enter the value of y:");***

***int y = s.nextInt();***

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

***System.out.println("1.GreaterValue\n2.SmallerValue");***

***System.out.println("Enter the Choice:");***

***switch(s.nextInt())***

***{***

***case 1:***

***//LambdaExpression***

***IComparable gv = (int v1,int v2)->***

***{***

***if(v1>v2) return v1;***

***else return v2;***

***};***

***int r1 = gv.compare(x, y);***

***System.out.println("GreaterValue:"+r1);***

***break;***

***case 2:***

***//LambdaExpression***

***IComparable sv = (int v1,int v2)->***

***{***

***if(v1<v2) return v1;***

***else return v2;***

***}***

***;***

***int r2 = sv.compare(x, y);***

***System.out.println("SmallerValue:"+r2);***

***break;***

***default:***

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

***}//end of switch***

***s.close();***

***}***

***}***

***---------------------------------------------------------------***

***ClassFiles:***

***IComparable.class***

***LambdaExpression2.class(MainClass)***

***==================================================================***

***Assignment-1:***

***Convert IArithmetic application into LambdaExpressions***

***Assignment-2:***

***Convert BankTransaction application into LambdaExpression***

***===================================================================***