JOURNAL -2

1.Design a class to represent a bank account. Include the following members (Using multiple constructor)

Data members: name of depositor, account number,type of account, balance amount in the account.)

Method: To assign initial values, To deposit an amount, To withdraw an amount after checking balance,To display the name and balance.

**CODE**

import java.util.\*;

public class BankAccount

{

private String depositorName;

private int accountNumber;

private String accountType;

private double balance;

// Constructor with no arguments

public BankAccount() {

this("", 0, "Savings", 0.0);

}

// Constructor with name and account number arguments

public BankAccount(String name, int accountNumber) {

this(name, accountNumber, "Savings", 0.0);

}

// Constructor with all arguments

public BankAccount(String name, int accountNumber, String accountType, double balance) {

this.depositorName = name;

this.accountNumber = accountNumber;

this.accountType = accountType;

this.balance = balance;

}

// Method to assign initial values

public void setInitialValues(String name, int accountNumber, String accountType, double balance) {

this.depositorName = name;

this.accountNumber = accountNumber;

this.accountType = accountType;

this.balance = balance;

}

// Method to deposit an amount

public void deposit(double amount) {

if(amount > 0) {

balance += amount;

System.out.println("Deposit successful. New balance is " + balance);

} else {

System.out.println("Invalid amount. Please enter a positive amount to deposit.");

}

}

// Method to withdraw an amount after checking balance

public void withdraw(double amount) {

if(amount <= 0) {

System.out.println("Invalid amount. Please enter a positive amount to withdraw.");

} else if(amount > balance) {

System.out.println("Insufficient balance. You can withdraw up to " + balance);

} else {

balance -= amount;

System.out.println("Withdrawal successful. New balance is " + balance);

}

}

// Method to display name and balance

public void display() {

System.out.println("Depositor name: \n" + depositorName);

System.out.println("Account number: \n" + accountNumber);

System.out.println("Account type: \n" + accountType);

System.out.println("Current balance: \n" + balance);

}

public static void main(String args[]){

Scanner s=new Scanner(System.in);

// Create an instance of BankAccount with no arguments

BankAccount account1 = new BankAccount();

// Set initial values using setInitialValues method

account1.setInitialValues("John Smith", 12345, "Savings",0.0);

// Deposit 500

System.out.println("Enter the amount That You want to Deposit:");

double depositAmount=s.nextDouble();

account1.deposit(depositAmount);

// Withdraw 200

System.out.println("Enter the amount That You want to Withdrawr:");

double withdrawAmount=s.nextDouble();

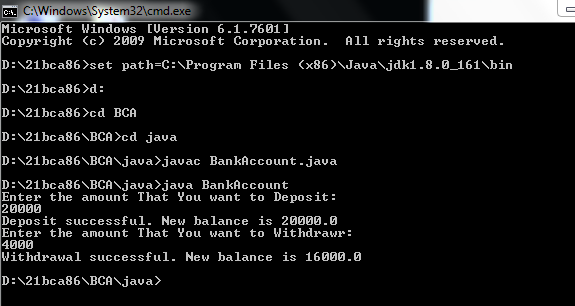
account1.withdraw(withdrawAmount);

// Display account information

}

}

**OUTPUT**

****

2. Write a program to print floyed’s triangle where n is command line input.

1

2 3

4 5 6

7 8 9 10

**CODE**

import java.util.\*;

class triangle

{

public static void main(String[] args)

{

int n = 5;

int i, j, k = 1;

for (i = 1; i <= n; i++)

{

for (j = 1; j <= i; j++)

{

System.out.print(k + " ");

k++;

}

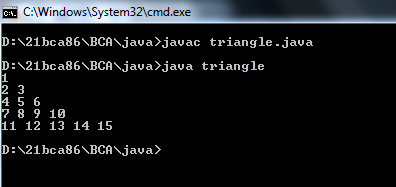
System.out.println();

}

}

}

**OUTPUT**



3.Design a class cricket having data member name a of of matches and appropriate member function to set the values. Drived two classes batsman and bowler from cricketer class with the data member total number of runs and wickets respectively.Batsman class is having method to calculate average wicket. Write a program to create two object and display information of one batsman and bowler along with average run and wicket.

**CODE**

class Crickter

{

public String name;

public double matchPlayed;

public void setValues(String name,double matchPlayed)

{

this.name=name;

this.matchPlayed=matchPlayed;

}

}

class Batsman extends Crickter

{

public double totalruns;

public double avarageRuns(double totalruns)

{

this.totalruns=totalruns;

return this.totalruns/matchPlayed;

}

}

class Bowler extends Crickter

{

public double wickets;

public double avarageWickets(double wickets)

{

this.wickets=wickets;

return this.wickets/matchPlayed;

}

}

class average

{

public static void main(String args[])

{

Batsman bm=new Batsman();

bm.setValues("Sachin Tendulkar",782);

double avgRuns=bm.avarageRuns(34357);

System.out.println("Batsman Information\n");

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

System.out.println("Batsman Run:"+bm.totalruns);

System.out.println("Batsman Match Played:"+bm.matchPlayed);

System.out.println("Batsman Avg Runs:"+avgRuns);

Bowler br=new Bowler();

br.setValues("Muttiah Muralitharan",583);

double avgWickets=br.avarageWickets(1347);

System.out.println("\nBowler Information\n");

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

System.out.println("Bpwler Wicketcs:"+br.wickets);

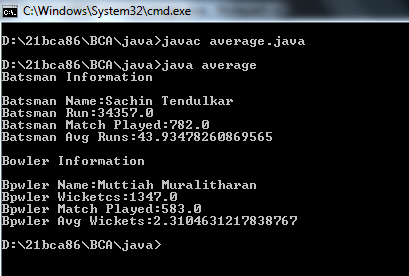
System.out.println("Bpwler Match Played:"+br.matchPlayed);

System.out.println("Bpwler Avg Wickets:"+avgWickets);

}

}

**OUTPUT**



4. Write a program that will accept two string or twonumber from command line and create overloaded method that add these two number or concate two string.

**CODE**

class OverloadMethod

{

public void display(String s1,String s2)

{

System.out.println("The Concated String is:"+(s1+s2));

}

Public void display(int a,int b)

{

System.out.println("The Addition of "+a+" and "+b+" is:"+(a+b));

}

public static void main(String args[])

{

OverloadMethod o=new OverloadMethod();

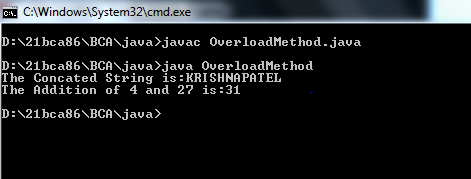
o.display("KRISHNA", "PATEL");

o.display(4,27);

}

}

**OUTPUT**



5.Write a program that accept a number from command line and check whether it is palindrome or not.

**CODE**

import java.util.\*;

class Palindrome1

{

public static void main(String args[])

{

String original, reverse = "";

Scanner in = new Scanner(System.in);

System.out.println("Enter a string to check if it is a palindrome");

original = in.nextLine();

int length = original.length();

for ( int i = length - 1; i >= 0; i-- )

reverse = reverse + original.charAt(i);

if (original.equals(reverse))

System.out.println("Entered is a palindrome.");

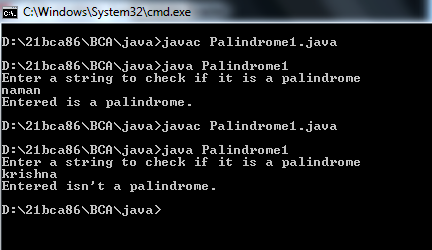
else

System.out.println("Entered isn't a palindrome.");

}

}

**OUTPUT**



6.Write a program that will accept a string from command liner and arrange all the character in alphabetical order.

E.g. input-computer

Output-cemoprtu

**CODE**

import java.util.Arrays;

import java.util.Scanner;

public class SortingString

{

public static void main(String args[])

{

Scanner sc = new Scanner(System.in);

System.out.println("Enter a string value: ");

String str = sc.nextLine();

char charArray[] = str.toCharArray();

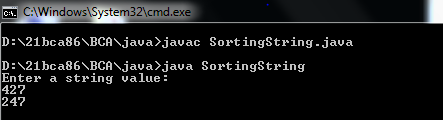
Arrays.sort(charArray);

System.out.println(new String(charArray));

}

}

**OUTPUT**



7.Write a program to create interface Area. Create three classes called rectangle,triangle and square calculater areas respectively.

**CODE**

interface Area{

public void Calculatearea();

}

class rectangle implements Area{

float x,y;

public rectangle(float x,float y){

this.x=x;

this.y=y;

}

public void Calculatearea(){

System.out.println("The Area of Rectangle is:"+(this.x\*this.y));

}

}

class triangle implements Area{

float x,y;

public triangle(float x,float y){

this.x=x;

this.y=y;

}

public void Calculatearea(){

System.out.println("The Area of Triangle is:"+((this.x\*this.y)/2));

}

}

class square implements Area{

float x;

public square(float x){

this.x=x;

}

public void Calculatearea(){

System.out.println("The Area of Square is:"+(this.x\*this.x));

}

}

class CalculateArea{

public static void main(String args[]){

//hare hight=20 and width=40

rectangle re=new rectangle(20,40);

re.Calculatearea();

//hare base=20 and hight=40

triangle tr=new triangle(20,40);

tr.Calculatearea();

//hare hight=40

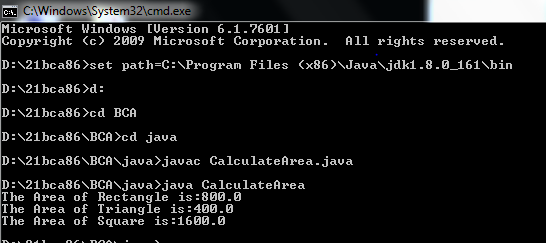
square sq=new square(40);

sq.Calculatearea();

}

}

**OUTPUT**



8. Write a program that will accept a number from command line and raise a user defined exception if the number consists of odd number of digits.

**CODE**

class OddException extends Exception

{

OddException(String str)

{

System.out.println(str);

}

}

class ExceptionHandaling

{

public static void main(String args[]){

int a=Integer.parseInt(args[0]);

try{

if(a%2==0){

System.out.println("The "+a+" is Even.");

}

else{

throw new OddException("Number Consist Odd Value.");

}

}

catch(OddException e)

{

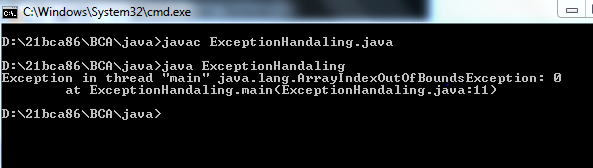
System.out.println(e);;

}

}

}

**OUTPUT**



9.Write a java application which accept 10 name of student and their age.Sort name and age in descending order.(using array)

**CODE**

import java.util.\*;

class StudentDetail

{

public static void main(String args[])

{

String StdName[]={"KRISHNA","ISHANVI","ZEBA","JESSICA","HEER","NIDHI","DIVYA","SIDDHI","KAJAL","SEJAL"};

int age[]={19,10,19,18,19,18,22,19,32,30};

int n=9;

String temp;

for (int i=0; i<=n;i++)

{

for (int j=i+1;j<=n;j++)

{

// to compare one string with other strings

if (StdName[i].compareTo(StdName[j]) > 0)

{

// swapping

temp = StdName[i];

StdName[i] = StdName[j];

StdName[j] = temp;

}

}

}

for(int i=0;i<=n;i++)

{

for(int j=i+1;j<=n;j++)

{

//to set the age in descnding order

int temp1;

if(age[i]>age[j])

{

temp1=age[i];

age[i]=age[j];

age[j]=temp1;

}

}

}

System.out.println("Names & Age in descnding order.");

System.out.println("Names\t\t\tAge");

System.out.println("==================== \t====================");

for (int i = n; i >=0; i--)

{

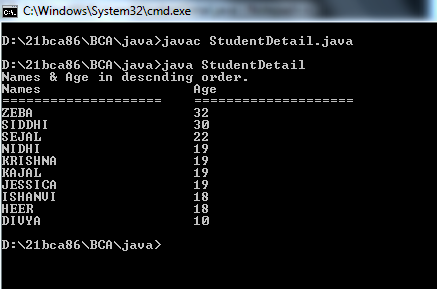
System.out.println(StdName[i]+"\t\t\t"+age[i]);

}

}

}

**OUTPUT**



10. Design a class Mystring having a data member of type string and add member function to achieve following task. (i) Reverse string (ii) String in Titlecase. (iii) Extract N-characters from right-end of the string Write a menu driven program to call these method of Mystring class. The program should not terminate abruptly.

**CODE**

import java.util.\*;

public class MyString

{

private String str;

public MyString(String str) {

this.str = str;

}

public String reverse()

{

return new StringBuilder(str).reverse().toString();

}

public String toTitleCase()

{

StringBuilder result = new StringBuilder(str.length());

String[] words = str.split("\\s");

for (String word : words) {

if (!word.isEmpty()) {

result.append(Character.toUpperCase(word.charAt(0)));

result.append(word.substring(1).toLowerCase());

}

result.append(" ");

}

return result.toString().trim();

}

public String extractNFromRight(int n) {

if (n >= str.length()) {

return str;

}

return str.substring(str.length() - n);

}

public static void main(String[] args)

{

Scanner scanner = new Scanner(System.in);

System.out.print("Enter a string: ");

String str = scanner.nextLine();

MyString myString = new MyString(str);

char ch;

do {

System.out.println();

System.out.println("Menu:");

System.out.println("1. Reverse string");

System.out.println("2. String in Titlecase");

System.out.println("3. Extract N-characters from right-end of the string");

System.out.println("4. Exit");

System.out.print("Enter your choice: ");

int choice = scanner.nextInt();

scanner.nextLine();

switch (choice) {

case 1:

System.out.println("Reversed string: " + myString.reverse());

break;

case 2:

System.out.println("Titlecased string: " + myString.toTitleCase());

break;

case 3:

System.out.print("Enter N: ");

int n = scanner.nextInt();

scanner.nextLine();

System.out.println("Extracted " + n + " characters from right: " + myString.extractNFromRight(n));

break;

case 4:

System.out.println("Exiting...");

System.exit(0);

break;

default:

System.out.println("Invalid choice. Try again.");

}

System.out.println("Do you want to continue?(press=y)");

ch=scanner.next().charAt(0);

}while(ch=='y');

}

}

**OUTPUT**

