Question 1

Write a program to check whether a list of numbers are palindrome or not.

/\* This is a program to check if a string is a palindrome\*/

/\* Date:- 28th January 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

import java.util.\*;

import java.io.\*;

public class palindrome {

public static void main(String[] args)throws IOException {

System.out.println("enter no of strings");

Scanner keyboard = new Scanner(System.in);//To take keyboard input

String myint;

int n =keyboard.nextInt();

ArrayList<String> stuff = new ArrayList<String>();//To take strings in an ArrayList

System.out.println("enter strings");

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

{

myint = keyboard.nextLine();

//System.out.println(myint);

stuff.add(myint);

}

Iterator iti=stuff.iterator();// Iterate through the list

//System.out.println(stuff);

String b=(String)iti.next();

while(iti.hasNext())

{

if(isPalindrome((String)iti.next()))//Check whether it is a palindrome

System.out.println("Yes");

else

System.out.println("no");

}

}

public static boolean isPalindrome(String pal) {//Function checks whether string is a palindrome or not

String pal1=""; int i=pal.length()-1;

while(i>=0)// Checks palindrome

pal1+=pal.charAt(i--);

if(pal1.compareTo(pal)==0)

return true;

return false;

}

}

ipse

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 1.01 | 2, 438, 949 (no. of inputs, number 1, number 2) | No Yes | No Yes | Pass |

Question 2

Write a program to demonstrate the way of method overriding by subclasses with different numbers and types of parameters.

/\* This is a program to create a base class \*/

/\* Date:- 28th January 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

public class base {// Creating a base class

int neo(int a,int b)

{

return (a\*b+a-b);

}

}

/\* This is a program to demonstrate the use of inheritance \*/

/\* Date:- 28th January 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

class derivedclass extends base {// creating a derivedclass

int neo(int a,int b)//method overriding

{

return a\*b;

}

};

public class derived{

public static void main(String args[])

{

derivedclass abc= new derivedclass();//Inheritance

base abcd = new base();//Inheritance

base ab = new derivedclass();

System.out.println(abc.neo(4,5));

System.out.println(abcd.neo(4,5));

System.out.println(ab.neo(4,5));

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 2.01 | 4, 5 (number 1, number 2) | Base output: 19 Sub output: 20 sub output: 20 | Base output: 19 Sub output: 20 sub output: 20 | Pass |

Question 3

Write a program to demonstrate how to use finally block to catch runtime exception (illegal argument exception) by the use of e.getMessage()

/\* This is a program to demonstrate how to use finally block to catch runtime exception using e.getMessage()\*/

/\* Date:- 28th January 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

import java.io.\*;

public class Finally

{

public static void main(String[]args)throws IOException

{

BufferedReader br=new BufferedReader(new InputStreamReader(System.in));//taking input via keyboard

System.out.print("Enter divisor: ");//enter divisor

int div=Integer.parseInt(br.readLine());//taking input

int a,c;

try //try block

{

a=5;

c=a/div;//division

System.out.println("Result obtained :"+c);

System.out.println("End of try block reached.");

}

catch(ArithmeticException e)//catching ArithmeticException

{

System.out.println("Error: "+e.getMessage());//print error message

}

finally //finally block

{

System.out.println("Completed");//Printing

}

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 3.01 | 0 (divisor) | Error: / by zero Completed. | Error: / by zero Completed. | Pass |
| 3.02 | 100 (divisor) | Result obtained: 1 End of try block reached. Completed | Result obtained: 1 End of try block reached. Completed | Pass |

Question 4

Write a program to demonstrate how to handle multiple exception by using System.err.println() method of the System class.

/\* This is a program to demonstrate how to handle multiple exception by using System.err.println()\*/

/\* Date:- 28th January 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

import java.io.\*;

public class MultipleExcep

{

public static void main(String[] args)throws IOException//throwing exception

{

int c;

int a[]=new int[5];

a[0]=1; a[1]=2; a[2]=3; a[3]=4; a[4]=5;//initializing array

System.out.print("Array: ");

for(int i:a)//running a for loop

System.out.print(i+" ");//printing the array

System.out.println();

BufferedReader br=new BufferedReader(new InputStreamReader(System.in));//taking keyboard input

try //try block

{

System.out.print("Enter element number to be accessed: ");

int index=Integer.parseInt(br.readLine())-1;//entering index

System.out.print("Element accessed: "+a[index]+"\nEnter divisor: ");//entering divisor

int div=Integer.parseInt(br.readLine());

System.out.println("Result obtained after division: "+a[index]/div);//division taking place

}

catch(ArrayIndexOutOfBoundsException e)//catching ArrayIndexOutOfBoundsException

{

System.err.println("Error: "+e);//PrintingError

}

catch(ArithmeticException e)//Catching ArithmeticException

{

System.err.println("Error: "+e);//PrintingError

}

finally //finally block

{

System.out.println("Finally");

}

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 4.01 | 4, 2 (element number in array, divisor) | Element accessed: 4 Result obtained after division: 2 Finally | Element accessed: 4 Result obtained after division: 2 Finally | Pass |
| 4.02 | 6 (element number in array) | Finally Error: java.lang.ArrayIndexOutOfBoundsException: 5 | Finally Error: java.lang.ArrayIndexOutOfBoundsException: 5 | Pass |
| 4.03 | 5, 0 (element number in array, divisor) | Element accessed: 5 Finally Error: java.lang.ArithmeticException: / by zero | Element accessed: 5 Finally Error: java.lang.ArithmeticException: / by zero | Pass |

Question 5

Write a program to compare the performance of two strings created in two different ways.

/\* This is a program to compare creation time of strings in two different ways \*/

/\* Date:- 28th January 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

public class stringcompare {

public static void main(String[] args) {

string1();//first way of creating

string2();//second way of creating

}

public static void string1()

{

long start = System.nanoTime();//to take the start time

String n="Sankarshan";

long end1 = System.nanoTime();//to take the end time

System.out.println(end1-start);//Time taken by first method of creation

}

public static void string2()

{

long start = System.nanoTime();//to take the start time

String n=new String("Sankarshan");

long end1 = System.nanoTime();//to take the end time

System.out.println(end1-start);//Time taken by second method of creation

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 5.01 | Program run with input string: "Sankarshan" | Performance of 1 = 4 Performance of 2 = 50 | Performance of 1 = 4 Performance of 2 = 50 | Pass |

Question 6

Write a program to implement multiple inheritance.

/\* This is a program to implement multiple inheritance \*/

/\* Date:- 28th January 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

import java.io.\*;

public class base {// Creating a base class

int fun(int a,int b)

{

return (a\*b+a-b);

}

}

interface a//creating an interface

{

int fun(int a,int b);//defining a function fun

};

class test extends base implements a//creating a class which implements the interface

{

@Override//overriding fun

public int fun(int a, int b)

{

return a-b;

}

};

public class MultipleInheritance//creating a class

{

public static void main(String[] args)throws IOException//throws Exception

{

BufferedReader br=new BufferedReader(new InputStreamReader(System.in));//Taking Keyboard Input

System.out.println("Enter two numbers: ");

int num1=Integer.parseInt(br.readLine());

int num2=Integer.parseInt(br.readLine());

test temp=new test();//creating object of class test

System.out.println("Addition from base class: "+temp.add(num1,num2));//Performing addition

System.out.println("Calculation(fun) from test class: "+temp.fun(num1,num2));//performing calculation

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 6.01 | 7, 8 (number 1, number 2) | Addition from base class: 15 Calculation(fun) from test class: 55 | Addition from base class: 15 Calculation(fun) from test class: 55 | Pass |

Question 7

Write a program to get a file’s size in bytes by using file.exists and file.length method of File class.

/\* This is a program to get the size of a file \*/

/\* Date:- 4th February 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

import java.io.\*;

public class FileSize{

public static void main(String args[]){

File file = new File("/home/nitdgp/workspace/12CS88/sre/palindrome/src/palindrome/test.txt");//enter full path of //file

boolean fileExists = file.exists();//checking if file exists

if (fileExists)

System.out.println(file.length()+" bytes");//print the length

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 7.01 | Program run with input file path: "/home/nitdgp/workspace/12CS88/sre/palindrome/src/palindrome/test.txt" | 2 bytes | 2 bytes | Pass |

Question 8

Write a program to demonstrate the conversion of an array to a collection using Arrays.asList(name) method of Java Util class.

/\* This is a program to demonstrate the conversion of an array to a collection \*/

/\* Date:- 4th February 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

import java.util.\*;

import java.io.\*;

public class ConvertArrayList{

public static void main(String args[]){

BufferedReader in;//to take keyboard input

int n=0;

try{ //try block

in = new BufferedReader(new InputStreamReader(System.in));//taking input

System.out.println("How many elements you want to add to the array: ");

n = Integer.parseInt(in.readLine());//no of elements to add into array

}

catch(IOException e){//catching IOException

System.out.println("Read error!!");

}

String[] name = new String[n];

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

try{//try block

in=new BufferedReader(new InputStreamReader(System.in));

name[i] = in.readLine();//taking the elements

}

catch(IOException e){

System.out.println("Read Error!!");//catching IOException

}

}

List list = Arrays.asList(name); //Converting array into list

System.out.println();

for(Object li: list)//traversing the list

{

String str = (String)li;

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

}

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 8.01 | 0 (Number of elements to be in array) | No output | No output | Pass |
| 8.02 | 5, "J", "S", "R", "L", "L" (number of elements to be in array, string 1, string 2, string 3, string 4, string 5) | J S R L L | J S R L L | Pass |

Question 9

Write a program to demonstrate how to remove a certain element from a collection with the help of collection.remove() method of collection class.

/\* This is a program to demonstrate the removal of a certain element \*/

/\* Date:- 4th February 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

import java.util.\*;

public class RemoveCollection{

public static void main(String [] args){

System.out.println( "Collection Example!\n" );

int size;

HashSet collection = new HashSet ();//making a HashSet

String str1 = "Yellow", str2 = "White", str3 =

"Green", str4 = "Blue"; //Making Strings

Iterator iterator;

collection.add(str1);// Adding strings HashSet

collection.add(str2);// Adding strings HashSet

collection.add(str3);// Adding strings HashSet

collection.add(str4);// Adding strings HashSet

System.out.print("Collection data: ");

iterator = collection.iterator();

while (iterator.hasNext()){//Traversing through the collection

System.out.print(iterator.next() + " ");

}

System.out.println();

collection.remove(str2);//remove the element

System.out.println("After removing [" + str2 + "]\n");

System.out.print("Now collection data: ");

iterator = collection.iterator();

while (iterator.hasNext()){

System.out.print(iterator.next() + " ");

}

System.out.println();

size = collection.size();

System.out.println("Collection size: " + size + "\n");

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 9.01 | Program run with input strings: "Yellow", "White", "Green" and "Blue" | Collection Example! Collection data: Yellow White Green Blue After removing [White] Now collection data: Yellow Green Blue Collection size: 3 | Collection Example! Collection data: Yellow White Green Blue After removing [White] Now collection data: Yellow Green Blue Collection size: 3 | Pass |

Question 10

Write a program to demonstrate the uses of different types of collection classes and add an element in those collections.

/\* This is a program to demonstrate the different types of collection classes \*/

/\* Date:- 4th February 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

import java.util.Map;

import java.util.Set;

import java.util.SortedMap;

import java.util.SortedSet;

import java.util.TreeMap;

import java.util.TreeSet;

import java.util.ArrayList;

import java.util.Collection;

import java.util.HashMap;

import java.util.HashSet;

import java.util.Iterator;

import java.util.LinkedHashMap;

import java.util.LinkedHashSet;

import java.util.LinkedList;

import java.util.List;

public class CollectionClasses

{

public static void main(String[] args)

{

List lnkLst = new LinkedList();//creating a linkedList

lnkLst.add("First");

lnkLst.add("Second");

lnkLst.add("Third");

lnkLst.add("Fourth");

displayAll(lnkLst);

List aryLst = new ArrayList();//creating an ArrayList

aryLst.add("x");

aryLst.add("y");

aryLst.add("z");

aryLst.add("w");

displayAll(aryLst);

Set hashSet = new HashSet();//creating a HashSet

hashSet.add("setElement1");

hashSet.add("setElement2");

hashSet.add("setElement3");

hashSet.add("setElement4");

displayAll(hashSet);

SortedSet treeSet = new TreeSet();//creating a TreeSet

treeSet.add("1");

treeSet.add("2");

treeSet.add("3");

treeSet.add("4");

displayAll(treeSet);

LinkedHashSet lnkHashset = new LinkedHashSet();//creating a linkedHashSet

lnkHashset.add("one");

lnkHashset.add("two");

lnkHashset.add("three");

lnkHashset.add("four");

displayAll(lnkHashset);

Map map1 = new HashMap();//creating a HashMap

map1.put("key1", "J");

map1.put("key2", "K");

map1.put("key3", "L");

map1.put("key4", "M");

displayAll(map1.keySet());

displayAll(map1.values());

SortedMap map2 = new TreeMap();//creating a Sorted TreeMap

map2.put("key1", "JJ");

map2.put("key2", "KK");

map2.put("key3", "LL");

map2.put("key4", "MM");

displayAll(map2.keySet());

displayAll(map2.values());

LinkedHashMap map3 = new LinkedHashMap();//creating a linkedHashMap

map3.put("key1", "JJJ");

map3.put("key2", "KKK");

map3.put("key3", "LLL");

map3.put("key4", "MMM");

displayAll(map3.keySet());

displayAll(map3.values());

}

static void displayAll(Collection col) {

Iterator itr = col.iterator();

while (itr.hasNext()) {

String str = (String) itr.next();

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

}

System.out.println();

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 10.01 | Program run with inputs: "F", "S", "T", "F" (Input for linked list) "A1", "A2", "A3", "A4" (Input for ArrayList) "1st", "2nd", "3rd", "4th" (Input for HashSet) "100", "101", "102", "103" (Input for TreeSet) "HSet1", "HSet2", "HSet3", "HSet4" (Input for LinkedHashSet) "Key1"-"Sam", "Key2"-"Mike", "Key3"-"Ron", "Key4"- "Sanky" (Input for HashMap) "Key1"-"abc", "Key2"-"def", "Key3"-"ghi", "Key4"- "ijk" (Input for TreeMap) "Key1"-"lmn", "Key2"-"opq", "Key3"-"rst", "Key4"-" uvw" (Input for LinkedHashMap) | F S T F A1 A2 A3 A4 4th 3rd 2nd 1st Member 100 101 102 103 HSet1 HSet2 HSet3 HSet4  Key2 Key1 Key4 Key3 Mike Sam Sanky Ron Key1 Key2 Key3 Key4  abc def ghi ijk  Key1 Key2 Key3 Key4  lmn opq rst uvw | F S T F A1 A2 A3 A4 4th 3rd 2nd 1st Member 100 101 102 103 HSet1 HSet2 HSet3 HSet4  Key2 Key1 Key4 Key3 Mike Sam Sanky Ron Key1 Key2 Key3 Key4  abc def ghi ijk  Key1 Key2 Key3 Key4  lmn opq rst uvw Second Key Third Key Fourth Key | Pass |

Question 11

Write a program to solve the producer consumer problem using threads.

/\* This is a program to solve the Producer Consumer Problem using Threads \*/

/\* Date:- 4th February 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

public class ProducerConsumerThreads { //Defining the primary class ProducerConsumerThreads

public static void main(String[] args) { //Defining the main function of ProducerConsumerThreads

Data c = new Data();

Producer produce = new Producer(c, 1);//Creating a Producer object

Consumer consume = new Consumer(c, 1);//Creating a Consumer object

produce.start(); //Starting the Producer thread

consume.start(); //Starting the Consumer thread

}

}class Data{

private double contents; //Checking contents

private boolean available = false; //Checking availability of contents

public synchronized double get() {

while (available == false) {

try {

wait(); //Waiting for producer to produce

}

catch (InterruptedException e) {

}

}

available = false;

notifyAll();

return contents;

}

public synchronized void put(double value) {

while (available == true) {

try {

wait();//Waiting for consumer to consume

}

catch (InterruptedException e) {

}

}

contents = value;// make contents equal to value

available = true;

notifyAll();

}

}

class Consumer extends Thread { //Making a Consumer class which extends Thread

private Data Data;// Making a private Data object

private double number;

public Consumer(Data c, double number) {//Constructor initialisation

Data = c;

this.number = number;

}

public void run() {//In this function, Consumer consumes the data

double value = 0;

for (int i = 0; i < 10; i++) {

value = Data.get();//Consumption going on

System.out.println("Consumer #"

+ this.number

+ " got: " + value);// printing the Consumed Value

}

}

}

class Producer extends Thread {//Making a Producer class which extends Thread

private Data Data;// Making a private Data object

private double number;

public Producer(Data c, double number) {//Constructor initialisation

Data = c;

this.number = number;

}

public void run() {//In this function, Producer produces the data

for (int i = 0; i < 10; i++) {

double copy=Math.random()\*100;//Production going on

Data.put(copy);

System.out.println("Producer #" + this.number

+ " put: " + copy);//Printing the produced value

try {

sleep((int)(Math.random() \* 100));

} catch (InterruptedException e) { }

}

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 11.01 | Program run with Producer and Consumer loop input: 10 | Consumer #1 got: 0 Producer #1 put: 0 Consumer #1 got: 1 Producer #1 put: 1 Consumer #1 got: 2 Producer #1 put: 2 Consumer #1 got: 3 Producer #1 put: 3 Consumer #1 got: 4 Producer #1 put: 4 Consumer #1 got: 5 Producer #1 put: 5 Consumer #1 got: 6 Producer #1 put: 6 Consumer #1 got: 7 Producer #1 put: 7 Consumer #1 got: 8 Producer #1 put: 8 Consumer #1 got: 9 Producer #1 put: 9 | Consumer #1 got: 0 Producer #1 put: 0 Consumer #1 got: 1 Producer #1 put: 1 Consumer #1 got: 2 Producer #1 put: 2 Consumer #1 got: 3 Producer #1 put: 3 Consumer #1 got: 4 Producer #1 put: 4 Consumer #1 got: 5 Producer #1 put: 5 Consumer #1 got: 6 Producer #1 put: 6 Consumer #1 got: 7 Producer #1 put: 7 Consumer #1 got: 8 Producer #1 put: 8 Consumer #1 got: 9 Producer #1 put: 9 | Pass |

Question 12

Write a program to implement stack by creating user defined push() method for entering elements and pop() method for retrieving elements from stack.

/\* This is a program to implement stack \*/

/\* Date:- 4th February 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

import java.io.\*;

public class stack {

private int size;

private long[] arr;

private int top;

public stack(int s) {

size = s;

arr = new long[size];

top = -1;//initialize top to -1

}

public void push(long j) {

arr[++top] = j;//push elements into stack

}

public long pop() {

return arr[top--];//pop elements out of the stack

}

public boolean isEmpty() {

return (top == -1);//check if stack is empty

}

public boolean isFull() {

return (top == size - 1);//check if stack is full

}

public static void main(String[] args)throws IOException {

stack s = new stack(10);//creating a stack object

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));//taking keyboard input

System.out.print("Enter number of elements in stack ");

int n=Integer.parseInt(br.readLine());

while(n-->0)

s.push(Integer.parseInt(br.readLine()));//push

while (!s.isEmpty()) {

long value = s.pop();//pop

System.out.print(value);

System.out.print(" ");

}

System.out.println("");

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 12 | Stack size entered: 3 |  |  |  |
| 12.01 | Entered: 0 (command to pop) when stack is empty | Error!! Empty Stack. | Error!! Empty Stack. | Pass |
| 12.02 | Entered: 1 (command to push) when stack is empty Entered: 45 (element to be pushed) | Pushed | Pushed | Pass |
| 12.03 | Entered: 0 (command to pop) when stack is in normal condition with only one element | Popped element: 45 Stack empty | Popped element: 45 Stack empty | Pass |
| 12.04 | Entered: 1 (command to push) when stack has an element already Entered: 10 (element to be pushed) | Pushed | Pushed | Pass |
| 12.05 | Entered: 1 (command to push) when stack has 2 elements Entered: 12 (element to be pushed) | Pushed Stack full | Pushed Stack full | Pass |
| 12.06 | Entered: 1 (command to push) when stack is full | Stack overflow. | Stack overflow. | Pass |
| 12.07 | Entered: 0 (command to push) when stack is full and last element pushed is 12 | Popped element: 12 | Popped element: 12 | Pass |
| 12.08 | Entered: 0 (command to push) when stack has 2 elements and the last element pushed is 10 | Popped element: 10 | Popped element: 10 | Pass |
| 12.09 | Entered: -1 (command to stop) | No output | No output | Pass |

Question 13

Write a program to convert an infix to postfix expression using stack.

/\* This is a program to convert an infix to postfix \*/

/\* Date:- 11th February 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

import java.io.\*;

public class InToPost {

private Stack theStack;

private String input;

private String output = "";

public InToPost(String in) {

input = in;

int stackSize = input.length();//input stack size

theStack = new Stack(stackSize);

}

public String doTrans() {

for (int j = 0; j < input.length(); j++) {

char ch = input.charAt(j);

switch (ch) {

case '+':

case '-':

gotOper(ch, 1);

break;

case '\*':

case '/':

gotOper(ch, 2);

break;

case '(':

theStack.push(ch);//push

break;

case ')':

gotParen(ch);

break;

default:

output = output + ch;

break;

}

}

while (!theStack.isEmpty()) {//check if the stack is empty

output = output + theStack.pop();

}

System.out.println(output);

return output;

}

public void gotOper(char opThis, int prec1) {

while (!theStack.isEmpty()) {//check if stack is empty

char opTop = theStack.pop();

if (opTop == '(') {

theStack.push(opTop);

break;

}

else {

int prec2;

if (opTop == '+' || opTop == '-')

prec2 = 1;

else

prec2 = 2;

if (prec2 < prec1) {

theStack.push(opTop);

break;

}

else

output = output + opTop;

}

}

theStack.push(opThis);

}

public void gotParen(char ch){

while (!theStack.isEmpty()) {

char chx = theStack.pop();

if (chx == '(')

break;

else

output = output + chx;

}

}

public static void main(String[] args)

throws IOException {

System.out.println("Enter the string");

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

String input = br.readLine();

String output;

InToPost theTrans = new InToPost(input);

output = theTrans.doTrans();

System.out.println("Postfix is " + output + '\n');

}

class Stack {

private int maxSize;

private char[] stackArray;

private int top;

public Stack(int max) {

maxSize = max;

stackArray = new char[maxSize];

top = -1;

}

public void push(char j) {

stackArray[++top] = j;

}

public char pop() {

return stackArray[top--];

}

public char peek() {

return stackArray[top];

}

public boolean isEmpty() {

return (top == -1);

}

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 13.01 | 2\*3+7/5-2 (infix expression) | Postfix is 23\*75/+2- | Postfix is 23\*75/+2- | Pass |

Question 14

Write a program to display a clock using valueof() methods of string class and using calendar class to get the second, minute and hours.

/\* This is a program to display a clock\*/

/\* Date:- 11th February 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

import java.awt.\*;

import java.applet.\*;

import java.applet.\*;

import java.awt.\*;

import java.util.\*;

public class clock extends Applet implements Runnable{//implements Runnable and extends Applet

Thread t,t1;

public void start(){

t = new Thread(this);//start a new Thread

t.start();

}

public void run(){

t1 = Thread.currentThread();// initialize current thread

while(t1 == t){

repaint();//repaint the graphics

try{ //try

t1.sleep(1000);

}

catch(InterruptedException e){}//catch InterruptedException

}

}

public void paint(Graphics g){

Calendar cal = new GregorianCalendar();//making a Calendar Object

String hour = String.valueOf(cal.get(Calendar.HOUR));

String minute = String.valueOf(cal.get(Calendar.MINUTE));

String second = String.valueOf(cal.get(Calendar.SECOND));

g.drawString(hour + ":" + minute + ":" + second, 20, 30);//Printing Calendar

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 14.01 | Program run without any inputs from user | Real time clock display | Real time clock display | Pass |

Question 15

Write a program to create an applet which will fill colour in a rectangle using setcolor(), fillrect() methods of graphics class to fill colour in a rectangle

/\* This is a program to create a rectangle and fill it with a colour\*/

/\* Date:- 11th February 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

import java.applet.\*;

import java.awt.\*;

public class rectangleapplet extends Applet{

public void paint(Graphics g){

g.drawRect(300,150,200,100);//drawing rectangle

g.setColor(Color.black); //filling colour

g.fillRect( 300,150, 200, 100 );//drawing rectangle

g.setColor(Color.red);//filling colour

g.drawString("Rectangle",500,150);

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 15.01 | Program run with inputs: 400, 250 (X and Y coordinates of rectangle position) 500, 400 (rectangle width and height) Red (rectangle fill colour) | A red coloured rectangle of dimension 500 x 400, at position (400,250) | A red coloured rectangle of dimension 500 x 400, at position (400,250) | Pass |

Question 16

Write a program to use swing applet in Java by implementing ActionListener and by creating JLabels.

/\* This is a program to use swing applet\*/

/\* Date:- 11th February 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

import javax.swing.\*;

import java.applet.\*;

import java.awt.\*;

import java.awt.event.\*;

public class swingapplet extends Applet implements ActionListener {

TextField input,output;//creating TextField

Label label1,label2;//Creating Labels

Button b1;//Creating butttons

JLabel lbl;//Creating JLabel

int num, sum = 0;

public void init(){

label1 = new Label("please enter number : ");

add(label1);

label1.setBackground(Color.yellow);

label1.setForeground(Color.magenta);

input = new TextField(5);

add(input);

label2 = new Label("Sum : ");

add(label2);

label2.setBackground(Color.yellow);

label2.setForeground(Color.magenta);

output = new TextField(20);

add(output);

b1 = new Button("Add");

add(b1);

b1.addActionListener(this);

lbl = new JLabel("Swing Applet Example. ");

add(lbl);

setBackground(Color.yellow);

}

public void actionPerformed(ActionEvent ae){

try{

num = Integer.parseInt(input.getText());

sum = sum+num;

input.setText("");

output.setText(Integer.toString(sum));

lbl.setForeground(Color.blue);

lbl.setText("Output of the second Text Box : "

+ output.getText());

}

catch(NumberFormatException e){

lbl.setForeground(Color.red);

lbl.setText("Invalid Entry!");

}

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 16.01 | Program run without any inputs from user initially. Entered number in the applet viewer: 4 Entered number in the applet viewer: 9 | Output of the second Text Box: 4 Output of the second Text Box: 13 | Output of the second Text Box: 4 Output of the second Text Box: 13 | Pass |

Question 17

Write a program for Binary Search.

/\* This is a program to perform Binary Search\*/

/\* Date:- 11th February 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

import java.util.Scanner;

class binarySearch

{

public static void main(String args[])

{

int c, first, last, middle, n, search, array[];

Scanner in = new Scanner(System.in);//to take keyboard input

System.out.println("Enter number of elements");

n = in.nextInt(); //enter number of elements

array = new int[n];

System.out.println("Enter the integers");

for (c = 0; c < n; c++)

array[c] = in.nextInt();//enter the integers

System.out.println("Enter value to find");

search = in.nextInt();//enter key value

first = 0;

last = n - 1;

middle = (first + last)/2;

while( first <= last )

{

if ( array[middle] < search )

first = middle + 1;

else if ( array[middle] == search )

{

System.out.println(search + " found at location " + (middle + 1) + ".");

break;

}

else

last = middle - 1;

middle = (first + last)/2;

}

if ( first > last )

System.out.println(search + " is not there\n");

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 17.01 | 1, 17,17 (number of elements, array element 1, key) | 17 found at location 1 | 17 found at location 1 | Pass |
| 17.02 | 1, 17, 0 (number of elements, array element 1, key) | 0 is not present in the list | 0 is not present in the list | Pass |
| 17.03 | 4, 17, 21, 23, 29, 17 (number of elements, 4 array elements, key) | 17 found at location 1 | 17 found at location 1 | Pass |
| 17.04 | 7, 9, 16, 18, 30, 31, 41, 45, 45 (number of elements, 7 array elements, key) | 45 found at location 7 | 45 found at location 7 | Pass |
| 17.05 | 7, 17, 18, 21, 23, 29, 38, 41, 23 (number of elements, 7 array elements, key) | 23 found at location 4 | 23 found at location 4 | Pass |
| 17.06 | 7, 17, 18, 21, 23, 29, 33, 38, 21 (number of elements, 7 array elements, key) | 21 found at location 3 | 21 found at location 3 | Pass |
| 17.07 | 5, 12, 18, 21, 23, 32, 23 (number of elements, 5 array elements, key) | 23 found at location 4 | 23 found at location 4 | Pass |
| 17.08 | 5, 21, 23, 29, 33, 38, 25 (number of elements, 5 array elements, key) | 25 is not present in the list | 25 is not present in the list | Pass |

Question 18

Write a program to a database using JDBC. Assume that database name is testDB and it has table named employee which has 2 records.

/\* This is a program to create a database using JDBC\*/

/\* Date:- 9th March 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

**import java.sql.\*;** // *The star ( \* ) indicates that all of the classes in the package java.sql are to be imported.*

**public class odbc\_retrieve {**

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

**{**

**try**

**{**

**Class.forName("sun.jdbc.odbc.JdbcOdbcDriver"**); *// It loads class in memory. and while loading, it also calls static block of the class, which can instantiate the object and register it with DriverManager. DriverManager manages the set of Java Database Connectivity (JDBC) drivers that are available for an application to use.*

**Connection cobj = DriverManager.getConnection("jdbc:odbc:testDB");** // *getConnection() method is used to establish a connection to a database.*

**Statement st = cobj.createStatement(); //** *A statement object is used to send and execute SQL statements to a database.*

**ResultSet ob = st.executeQuery("select \* from employee");** *// returns a ResultSet object from which the result data can be read.*

**while(ob.next()) //** *The next() method is used to successively step through the rows of the tabular results.*

**{**

**System.out.println(ob.getString(1) + ob.getString(2));**

**}**

**}**

**catch(Exception e)**

**{**

**e.printStackTrace(); //** *method to print exceptional error*

**}**

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 18.01 | Program runs with input from the user..Making a connection to testDB and using quesry select \* from employee | The entire table | The entire Table | Pass |

Question 19

Write a program to execute multiple sql commands simultaneously on a database.

/\* This is a program to execute multiple sql commands simultaneously on a database\*/

/\* Date:- 9th March 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

import java.sql.\*;

public class jdbcConn {

public static void main(String[] args) throws Exception{

Class.forName("org.apache.derby.jdbc.ClientDriver");

Connection con = DriverManager.getConnection

("jdbc:derby://localhost:1527/testDb","name","pass");

Statement stmt = con.createStatement

(ResultSet.TYPE\_SCROLL\_SENSITIVE,

ResultSet.CONCUR\_UPDATABLE);

String insertEmp1 = "insert into emp values

(10,'jay','trainee')";//entering values

String insertEmp2 = "insert into emp values

(11,'jayes','trainee')";//entering values

String insertEmp3 = "insert into emp values

(12,'shail','trainee')";//entering values

con.setAutoCommit(false);

stmt.addBatch(insertEmp1);

stmt.addBatch(insertEmp2);

stmt.addBatch(insertEmp3);

ResultSet rs = stmt.executeQuery("select \* from emp");//execute query

rs.last();

System.out.println("rows before batch execution= "

+ rs.getRow());

stmt.executeBatch();

con.commit();

System.out.println("Batch executed");

rs = stmt.executeQuery("select \* from emp");//execute query

rs.last();

System.out.println("rows after batch execution= "

+ rs.getRow());

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 19.01 | Program runs with input :- 10,jay,trainee ; 11,jayes,trainee ; 12,shail,trainee | rows before batch execution= 6 Batch executed rows after batch execution= = 9 | rows before batch execution= 6 Batch executed rows after batch execution= = 9 | Pass |

Question 20

Write a program to join contents of more than one table and display the inner join sql command to combine data from the two tables.

/\* This is a program to join contents of more than one table\*/

/\* Date:- 9th March 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

import java.sql.\*;

public class jdbcConn\_1 {

public static void main(String[] args) throws Exception{

Class.forName("org.apache.derby.jdbc.ClientDriver");

Connection con = DriverManager.getConnection

("jdbc:derby://localhost:1527/testDb","username",

"password");

Statement stmt = con.createStatement();//create statement

String query ="SELECT fname,lname,isbn from author

inner join books on author.AUTHORID = books.AUTHORID";//query

ResultSet rs = stmt.executeQuery(query);

System.out.println("First Name Last Name ISBN");

while (rs.next()) {

String fname = rs.getString("fname");

String lname = rs.getString("lname");

int isbn = rs.getInt("isbn");

System.out.println(fname + " " + lname+" "+isbn);

}

System.out.println();

System.out.println();

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 20.01 | Program runs with input :- (First Name Last Name ISBN) john grisham 123 ; jeffrey archer 113 ; J.K. Rowling 112 ; Chetan Bhagat 122 | john grisham 123 ; jeffrey archer 113 ; J.K. Rowling 112 ; Chetan Bhagat 122 | john grisham 123 ; jeffrey archer 113 ; J.K. Rowling 112 ; Chetan Bhagat 122 | Pass |

Question 21

Write a program to design a login app using database and swing application

/\* This is a program to design a login app using database and swing application\*/

/\* Date:- 9th March 2015 \*/

/\* Authored by Sankarshan Purkayastha and Arun Pandey \*/

/\* Roll no 12/CS/88 and Roll no 12/CS/91\*/

/\* For fetching records we need a database table; for that we create a "reg" table in our database.

Syntax

reg.sql

create table emp

(

name varchar2(30), email varchar2(40),

pass varchar2(30), count varchar2(30),

state varchar2(30), phone number(15)

);

Insert some rows in it

1. insert into emp values ('Sankarshan', 'sankarshan@me.com', 'welcome', 'India','West Bengal','8100878312');

2. insert into emp values ('Spandan', 'spandie990@gmail.com' , '1234', 'India','West Bengal','9831494819');\*/

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.sql.\*;

public class Login extends JFrame implements ActionListener

{

JLabel l1, l2, l3;

JTextField tf1;

JButton btn1;

JPasswordField p1;

Login()

{

setTitle("Login Form in Windows Form");

setVisible(true);

setSize(800, 800);

setLayout(null);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

l1 = new JLabel("Login Form in Windows Form:");

l1.setForeground(Color.blue);

l1.setFont(new Font("Serif", Font.BOLD, 20));

l2 = new JLabel("Enter Email:");

l3 = new JLabel("Enter Password:");

tf1 = new JTextField();

p1 = new JPasswordField();

btn1 = new JButton("Submit");

l1.setBounds(100, 30, 400, 30);

l2.setBounds(80, 70, 200, 30);

l3.setBounds(80, 110, 200, 30);

tf1.setBounds(300, 70, 200, 30);

p1.setBounds(300, 110, 200, 30);

btn1.setBounds(150, 160, 100, 30);

add(l1);

add(l2);

add(tf1);

add(l3);

add(p1);

add(btn1);

btn1.addActionListener(this);

}

public void actionPerformed(ActionEvent e)

{

showData();

}

public void showData()

{

JFrame f1 = new JFrame();

JLabel l, l0;

String str1 = tf1.getText();

char[] p = p1.getPassword();

String str2 = new String(p);

try

{

Class.forName("oracle.jdbc.driver.OracleDriver");

Connection con = DriverManager.getConnection("jdbc:oracle:thin:@mcndesktop07:1521:xe", "username", "password");

PreparedStatement ps = con.prepareStatement("select name from reg where email=? and pass=?");

ps.setString(1, str1);

ps.setString(2, str2);

ResultSet rs = ps.executeQuery();

if (rs.next())

{

f1.setVisible(true);

f1.setSize(600, 600);

f1.setLayout(null);

l = new JLabel();

l0 = new JLabel("you are succefully logged in..");

l0.setForeground(Color.blue);

l0.setFont(new Font("Serif", Font.BOLD, 30));

l.setBounds(60, 50, 400, 30);

l0.setBounds(60, 100, 400, 40);

f1.add(l);

f1.add(l0);

l.setText("Welcome " + rs.getString(1));

l.setForeground(Color.red);

l.setFont(new Font("Serif", Font.BOLD, 30));

} else

{

JOptionPane.showMessageDialog(null,

"Incorrect email-Id or password..Try Again with correct detail");

}

}

catch (Exception ex)

{

System.out.println(ex);

}

}

public static void main(String arr[])

{

new Login();

}

}

Test Cases

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test ID | Description | Expected Result | Output | Remarks |
| 21.01 | Program runs with input :- [sankarshan@me.com](mailto:sankarshan@me.com) (e-mail) abc(password) | Incorrect e-mail id or password | Incorrect e-mail id or password | Pass |
| 21.0.2 | Program runs with input :- [sankarshana@me.com](mailto:sankarshan@me.com) (e-mail) welcome(password) | Incorrect e-mail id or password | Incorrect e-mail id or password | Pass |
| 21.0.2 | Program runs with input :- [sankarshana@me.com](mailto:sankarshana@me.com) (e-mail) abc(password) | Incorrect e-mail id or password | Incorrect e-mail id or password | Pass |
| 21.0.3 | Program runs with input :- [sankarshan@me.com](mailto:sankarshan@me.com) (e-mail) welcome(password) | Welcome Sankarshan, you are successfully logged in | Welcome Sankarshan, you are successfully logged in | Pass |