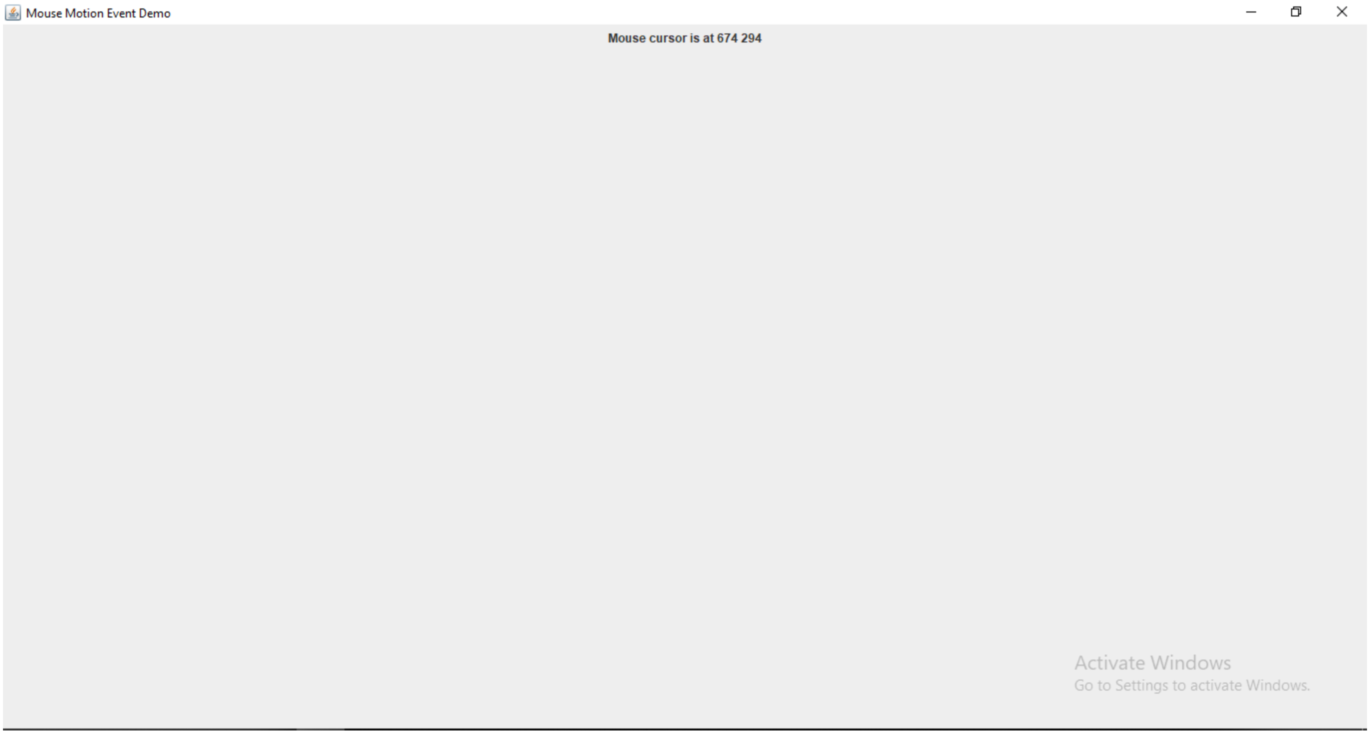
JAVA EXERCISE OUTPUT

Chapter - 1

Exercise 1:

```
import java.awt.*;

import java.awt.event.*;

import javax.swing.*;

public class MouseMoveDemo extends JFrame

{

   int x, y;

   JLabel position;


   public MouseMoveDemo()

   {

      super("Mouse Motion Event Demo");

      position = new JLabel();


      setLayout(new FlowLayout());


      add(position);


      setSize(320,300);

      setVisible(true);

      setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

     addMouseMotionListener(new MouseAdapter(){

       public void mouseMoved(MouseEvent me)

        {

            x = me.getX();

            y = me.getY();

            position.setText("Mouse cursor is at " + x + " " + y);
            }
        }
        );
    }
    public static void main(String[] args)
    {
        MouseMoveDemo obj = new MouseMoveDemo();
    }
}
```

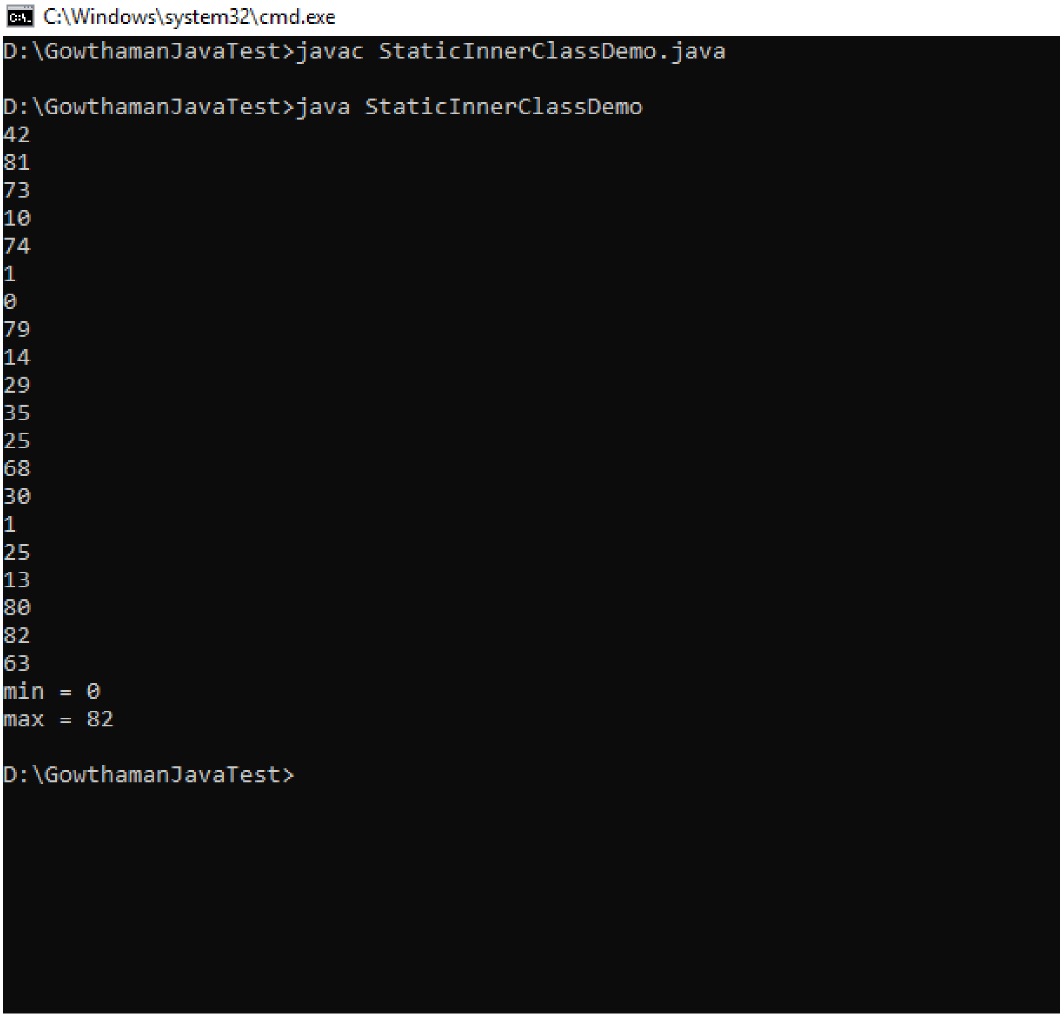


Mouse cursor is at 674 294

Exercise 2:

```
import java.util.Random;
public class StaticInnerClassDemo {
       public static class Pair{
       private int first;
      private int last;
      public Pair(int f, int s)  {
      first = f;
      last=s;
    }
    public int getFirst(){
      return first;
    }
    public int getLast(){
      return last;
    }
   }
   public static Pair minmax(int values[]){
     int min=2147483647;
     int max=-2147483648;
     for(int v:values){
       if(min>v)
         min=v;
       if(max<v)
         max=v;
     }
     return new Pair(min,max);
   }
 public static void main(String args[]){
   Random r=new Random();
```

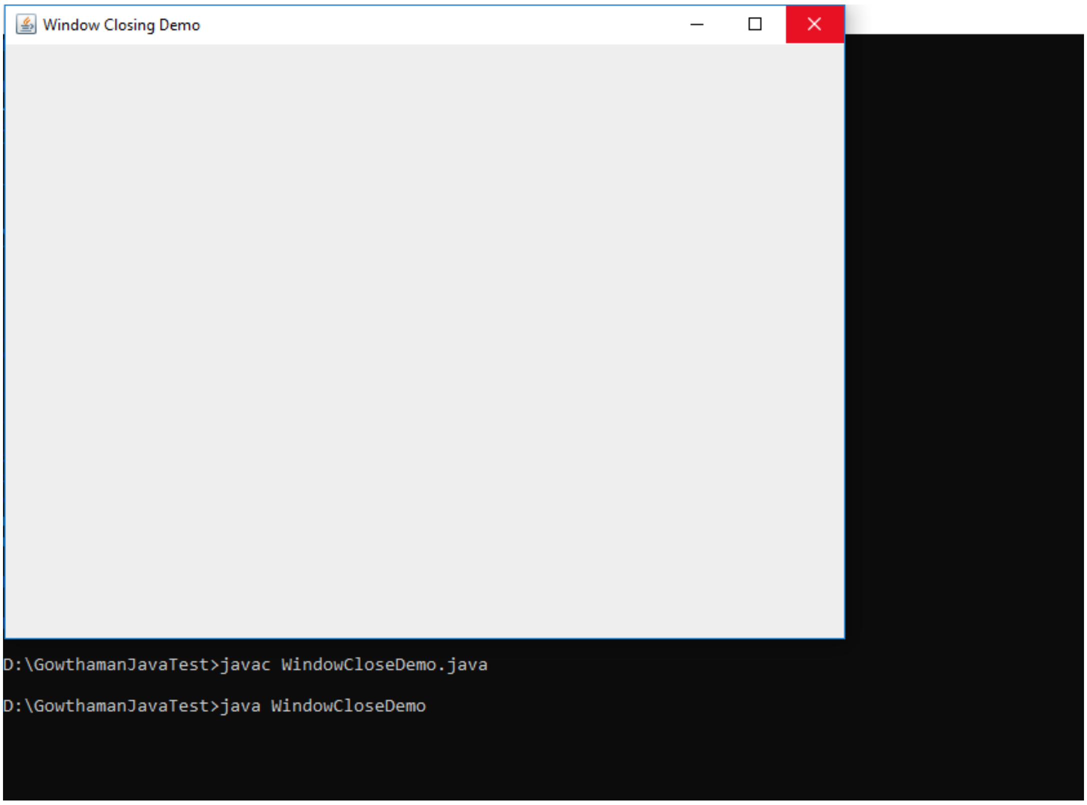
```
        int d[]=new int[20];

        for(int i=0;i<d.length;i++)

            {

          d[i]=r.nextInt(100);

          System.out.println(d[i]);

        }

        StaticInnerClassDemo.Pair p=StaticInnerClassDemo.minmax(d);

        System.out.println("min = "+p.getFirst());

        System.out.println("max = "+p.getLast());

    }

}
```



```
D:\GowthamanJavaTest>javac StaticInnerClassDemo.java

D:\GowthamanJavaTest>java StaticInnerClassDemo
42
81
73
10
74
1
0
79
14
29
35
25
68
30
1
25
13
80
82
63
min = 0
max = 82

D:\GowthamanJavaTest>
```

Exercise 3:

```
import javax.swing.*;

import java.awt.*;

import java.awt.event.*;

public class WindowCloseDemo extends JFrame

{

public WindowCloseDemo()

 {

super("Window Closing Demo");

this.addWindowListener(new WindowCloser());

setSize(300,300);

setVisible(true);

}

private class WindowCloser extends WindowAdapter

{

public void windowClosing(WindowEvent windowEvent)

{

System.exit(0);

}

}

public static void main(String args[]){

   WindowCloseDemo wcd=new WindowCloseDemo();

}

}
```

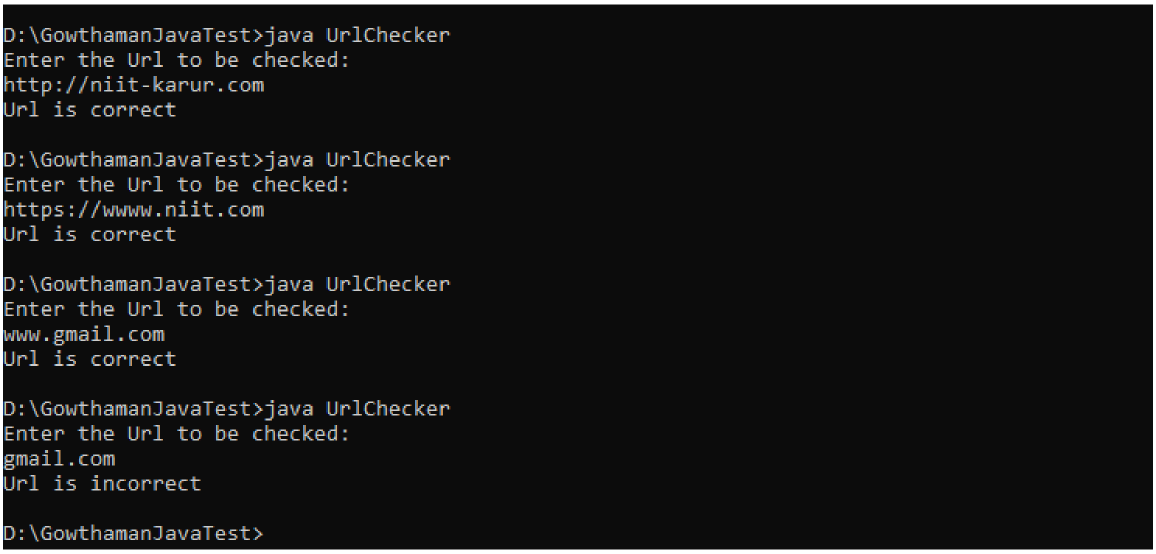


```
D:\GowthamanJavaTest>javac WindowCloseDemo.java

D:\GowthamanJavaTest>java WindowCloseDemo
```

Chapter - 2

Exercise: 1

```
import java.util.Scanner;
import java.util.regex.Matcher;
import java.util.regex.Pattern;

class UrlChecker {

   public static void main(String[] args) {
      Pattern Mypattern = Pattern.compile("^((https?|ftp)://|(www|ftp)\\.)[a-z0-9-]+(\\.[a-z0-9-]+[a-z]+)+([/?].*)?$");
      Scanner input = new Scanner(System.in);
      System.out.println("Enter the Url to be checked: ");
      String name = input.nextLine();
      Matcher Mymatcher = Mypattern.matcher(name);
      Boolean Myboolean = Mymatcher.matches();
      if (Myboolean == true) {
         System.out.println("Url is correct");
      } else {
         System.out.println("Url is incorrect");

      }
   }
}
```



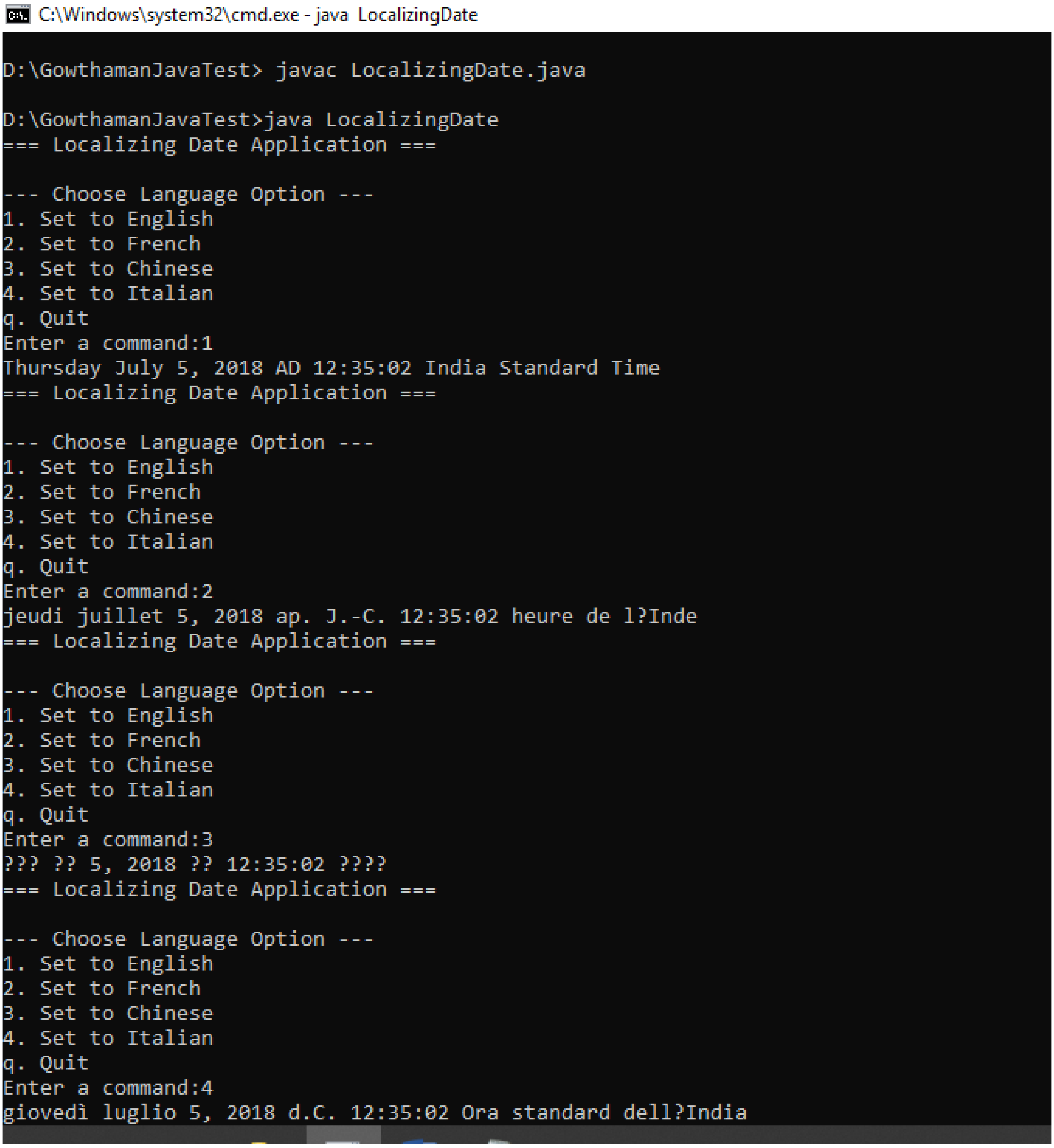
```
D:\GowthamanJavaTest>java UrlChecker
Enter the Url to be checked:
http://niit-karur.com
Url is correct

D:\GowthamanJavaTest>java UrlChecker
Enter the Url to be checked:
https://wwww.niit.com
Url is correct

D:\GowthamanJavaTest>java UrlChecker
Enter the Url to be checked:
www.gmail.com
Url is correct

D:\GowthamanJavaTest>java UrlChecker
Enter the Url to be checked:
gmail.com
Url is incorrect

D:\GowthamanJavaTest>
```

Exercise 2:

```
import java.text.DateFormat;

import java.text.SimpleDateFormat;

import java.util.Date;

import java.util.Locale;

import java.util.Scanner;


public class LocalizingDate

{

   Scanner br = new Scanner(System.in);

   Locale currentLocale = Locale.US;

   Date today = new Date();

   DateFormat df;

   SimpleDateFormat sdf;

   public static void main(String[] args)  {

      LocalizingDate dateApp = new LocalizingDate();

      dateApp.run();

   }

   public void run() {

      String line = "";
```

```
        this.printMenu();
        try {
            line = this.br.next();
        } catch (Exception e) {
            System.out.println(e);
        }
        switch (line) {
            case "1":
                setEnglish();
                run();
                break;
            case "2":
                setFrench();
                run();
                break;
case "3":
                setChinese();
                run();
                break;
    case "4":
                setItalian();
                run();
                break;
        }
    }

    public void printMenu() {
        System.out.println("=== Localizing Date Application ===");

        System.out.println("\n--- Choose Language Option ---");
        System.out.println("1. Set to English");
        System.out.println("2. Set to French");

        System.out.println("3. Set to Chinese");

        System.out.println("4. Set to Italian");

        System.out.println("q. Quit");

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

        df = DateFormat.getDateInstance(DateFormat.DEFAULT, currentLocale);

    }

    public void setEnglish() {


        currentLocale = Locale.US;




        sdf = new SimpleDateFormat("EEEE MMMM d, y G kk:mm:ss zzzz", currentLocale);

        System.out.println(sdf.format(today));

    }


    public void setFrench() {

        currentLocale = Locale.FRANCE;




        sdf = new SimpleDateFormat("EEEE MMMM d, y G kk:mm:ss zzzz", currentLocale);

        System.out.println(sdf.format(today));


    }

    public void setChinese() {

        currentLocale = Locale.CHINESE;

        sdf = new SimpleDateFormat("EEEE MMMM d, y G kk:mm:ss zzzz", currentLocale);

        System.out.println(sdf.format(today));

    }
```

```
public void setItalian() {
            currentLocale = Locale.ITALIAN;
            sdf = new SimpleDateFormat("EEEE MMMM d, y G kk:mm:ss zzzz", currentLocale);
      System.out.println(sdf.format(today));
   }
}
```



```
D:\GowthamanJavaTest> javac LocalizingDate.java

D:\GowthamanJavaTest>java LocalizingDate
=== Localizing Date Application ===

--- Choose Language Option ---
1. Set to English
2. Set to French
3. Set to Chinese
4. Set to Italian
q. Quit
Enter a command:1
Thursday July 5, 2018 AD 12:35:02 India Standard Time
=== Localizing Date Application ===

--- Choose Language Option ---
1. Set to English
2. Set to French
3. Set to Chinese
4. Set to Italian
q. Quit
Enter a command:2
jeudi juillet 5, 2018 ap. J.-C. 12:35:02 heure de l?Inde
=== Localizing Date Application ===

--- Choose Language Option ---
1. Set to English
2. Set to French
3. Set to Chinese
4. Set to Italian
q. Quit
Enter a command:3
??? ?? 5, 2018 ?? 12:35:02 ????
=== Localizing Date Application ===

--- Choose Language Option ---
1. Set to English
2. Set to French
3. Set to Chinese
4. Set to Italian
q. Quit
Enter a command:4
giovedì luglio 5, 2018 d.C. 12:35:02 Ora standard dell?India
```

Chapter 3

Exercise 1:

```
public class Event extends Task
{
   public Event(String value)
   {
      super(value);
   }
}
public class Meeting extends Task
{
   public Meeting(String value)
   {
      super(value);
   }
}
public class Task
{
   public String tsk;
   public Task(String value)
   {
      tsk = value;
   }
   public String toString()
   {
      return tsk;
   }
}
public class TaskProcessor<X>
{
    private X value;

    public TaskProcessor(X v)
    {
        value = v;
    }

    public X getTaskP()
    {
        return value;
    }
}
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import javax.swing.DefaultListModel;
import javax.swing.JButton;
import javax.swing.JComboBox;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JList;
import javax.swing.JOptionPane;
import javax.swing.JScrollPane;
import javax.swing.JTextField;

public class ToDoList extends JFrame implements ActionListener
{
    JLabel lbl_catg, lbl_task, lbl_time, lbl_list;
    JTextField txt_task;
    JComboBox jcb_task, jcb_hours, jcb_min;
```

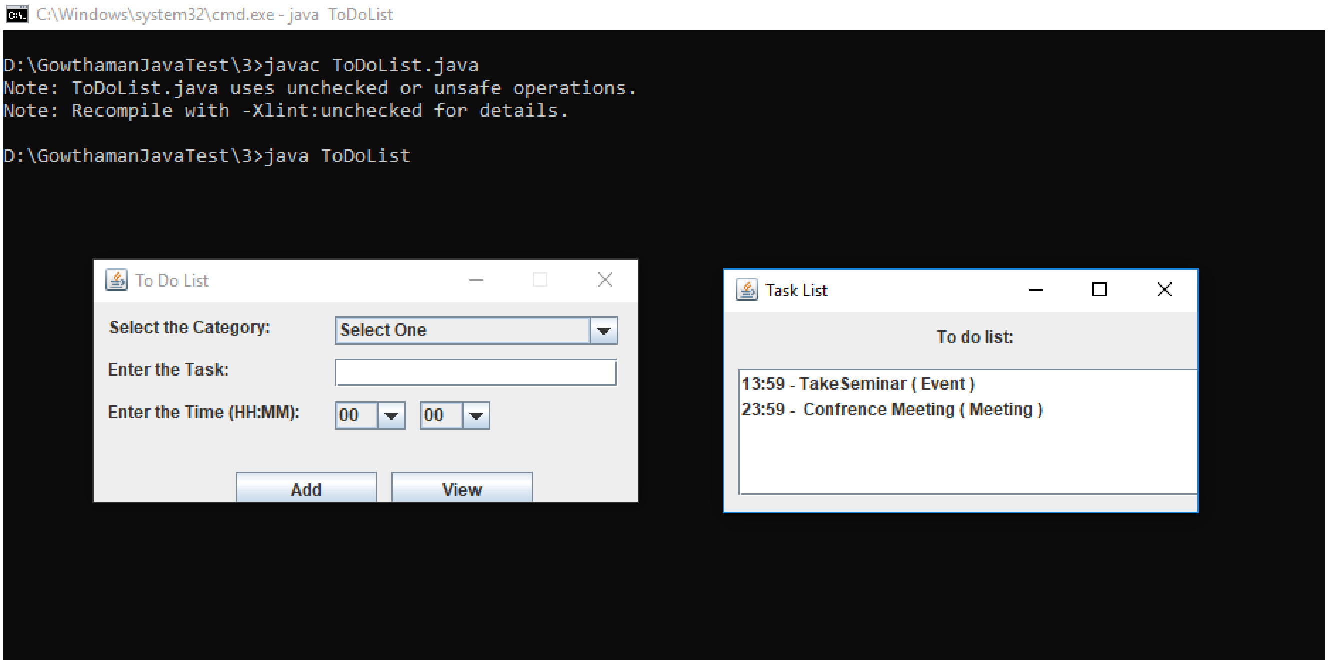
```
JList jl_task;
DefaultListModel dlm_task;
JScrollPane jsp_task;
JButton btn_add, btn_view;
JFrame list;
TaskProcessor<? extends Task> tp;
String hour, min;
int i_hour, i_min;
public ToDoList()
{
   setLocationRelativeTo(null);
   setLayout(null);
   setResizable(false);
   setTitle("To Do List");
   setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

   lbl_catg = new JLabel("Select the Category:");
   lbl_catg.setBounds(10, 10, 200, 15);
   add(lbl_catg);

   lbl_task = new JLabel("Enter the Task:");
   lbl_task.setBounds(10, 40, 100, 15);
   add(lbl_task);

   lbl_time = new JLabel("Enter the Time (HH:MM):");
   lbl_time.setBounds(10, 70, 150, 15);
   add(lbl_time);

   jcb_task = new JComboBox();
   jcb_task.addItem("Select One");
   jcb_task.addItem("Event");
   jcb_task.addItem("Meeting");
   jcb_task.setBounds(170, 10, 200, 20);
   add(jcb_task);

   txt_task = new JTextField();
   txt_task.setBounds(170, 40, 200, 20);
   add(txt_task);

   hour = new SimpleDateFormat("HH").format(Calendar.getInstance().getTime());
   min =  new SimpleDateFormat("MM").format(Calendar.getInstance().getTime());

   i_hour = Integer.parseInt(hour);
   i_min = Integer.parseInt(min);

   jcb_hours = new JComboBox();
   for (int i = 0; i < 24; i++)
   {
      if (i < 10)
      {
         jcb_hours.addItem("0" + i);
      }
      else
      {
         jcb_hours.addItem(i);
      }
   }

   jcb_hours.setBounds(170, 70, 50, 20);
   add(jcb_hours);

   jcb_min = new JComboBox();
```

```
for (int i = 0; i < 60; i++)
{
   if (i < 10)
   {
      jcb_min.addItem("0" + i);
   }
   else
   {
      jcb_min.addItem(i);
   }
}

jcb_min.setBounds(230, 70, 50, 20);
add(jcb_min);

btn_add = new JButton("Add");
btn_add.addActionListener(this);
btn_add.setBounds(100, 120, 100, 25);
add(btn_add);

btn_view = new JButton("View");
btn_view.addActionListener(this);
btn_view.setBounds(210, 120, 100, 25);
add(btn_view);

setSize(400, 180);
setVisible(true);

list = new JFrame("Task List");
list.setLayout(null);
list.setLocationRelativeTo(null);
```

```
    setResizable(false);


    lbl_list = new JLabel("To do list:");
    lbl_list.setBounds(150, 10, 100, 15);
    list.add(lbl_list);


    jl_task = new JList();
    dlm_task = new DefaultListModel();
    jl_task.setModel(dlm_task);
    jsp_task = new JScrollPane(jl_task);
    jsp_task.setBounds(10, 40, 335, 90);
    list.add(jsp_task);
    list.setSize(350, 180);
  }


  public void actionPerformed(ActionEvent e)
  {
    if (e.getSource() == btn_add)
    {
      int t_hour = Integer.parseInt(jcb_hours.getSelectedItem().toString());
      int t_min = Integer.parseInt(jcb_min.getSelectedItem().toString());
      if(jcb_task.getSelectedItem().equals("Select One"))
      {
        JOptionPane.showMessageDialog(this, "Please select the category of task.", "Warning",
JOptionPane.WARNING_MESSAGE);
      }
      else if(txt_task.getText().equals(""))
      {
        JOptionPane.showMessageDialog(this, "Please provide the proper task.", "Warning",
JOptionPane.WARNING_MESSAGE);
      }
      else if((t_hour < i_hour))
```

```
        {
            JOptionPane.showMessageDialog(this, "The time of the task must be greater than the current time.", "Warning", JOptionPane.WARNING_MESSAGE);
        }

        if (jcb_task.getSelectedItem().equals("Event")&&!(txt_task.getText().equals(""))&&!(t_hour < i_hour))
        {
            String item = jcb_hours.getSelectedItem().toString() + ":" + jcb_min.getSelectedItem().toString() + " - " + txt_task.getText() + " ( " + jcb_task.getSelectedItem() + " )";
            Event obj = new Event(item);
            tp = new TaskProcessor<>(obj);
            dlm_task.addElement(tp.getTaskP());
            jcb_task.setSelectedIndex(0);
            txt_task.setText(" ");
            jcb_hours.setSelectedIndex(0);
            jcb_min.setSelectedIndex(0);
        }
        else if (jcb_task.getSelectedItem().equals("Meeting")&&!(txt_task.getText().equals(""))&&!(t_hour <= i_hour))
        {
            String item = jcb_hours.getSelectedItem().toString() + ":" + jcb_min.getSelectedItem().toString() + " - " + txt_task.getText() + " ( " + jcb_task.getSelectedItem() + " )";
            Meeting obj2 = new Meeting(item);
            tp = new TaskProcessor<>(obj2);
            dlm_task.addElement(tp.getTaskP());
            jcb_task.setSelectedIndex(0);
            txt_task.setText(" ");
            jcb_hours.setSelectedIndex(0);
            jcb_min.setSelectedIndex(0);
        }
```

```
      }
      else if (e.getSource() == btn_view)
      {
         list.setVisible(true);
      }

   }

   public static void main(String[] args)
   {
      ToDoList tdl = new ToDoList();

   }
}
```



C:\Windows\system32\cmd.exe - java ToDoList

```
D:\GowthamanJavaTest\3>javac ToDoList.java
Note: ToDoList.java uses unchecked or unsafe operations.
Note: Recompile with -Xlint:unchecked for details.

D:\GowthamanJavaTest\3>java ToDoList
```

To Do List

Select the Category: Select One

Enter the Task:

Enter the Time (HH:MM): 00 00

Task List

To do list:

13:59 - TakeSeminar (Event)
23:59 - Confrence Meeting (Meeting)

Chapter 4

Exercise 1:

```
import java.util.LinkedList;

public class Animal
{
   LinkedList name;

   public Animal()
   {
      name = new LinkedList();
      name.add("ELEPHANT");
      name.add("TIGER");
      name.add("BUFFALO");
      name.add("GIRAFFE");
      name.add("LEOPARD");
   }
      public LinkedList getAnimal()
   {
      return name;
   }
   public String toString()
   {
      return "Animal";
   }
}
```

```
import java.util.LinkedList;


public class Country
{
  LinkedList name;

   public Country()
   {
      name = new LinkedList();
      name.add("AUSTRIA");
      name.add("CANADA");
      name.add("BAHRAIN");
      name.add("JAPAN");
      name.add("SYRIA");
   }

   public LinkedList getCountry()
   {
      return name;
   }

   public String toString()
   {
      return "Country";
   }

}
```

```
import java.util.LinkedList;
public class Fruit
{
   LinkedList name;

   public Fruit()
   {
      name = new LinkedList();
      name.add("MANGO");
      name.add("APPLE");
      name.add("GRAPES");
      name.add("PINEAPPLE");
      name.add("GUAVA");
   }

   public LinkedList getFruit()
   {
      return name;
   }

   public String toString()
   {
      return "Fruit";
   }
  }
```

```
import java.util.Arrays;

import java.util.Collection;

import java.util.LinkedList;

import java.util.Random;

import java.util.Scanner;

import java.util.TreeMap;


public class ListCreator

{

   TreeMap ctg, cust;

   Country country;

   Animal animal;

   Fruit fruit;


   public ListCreator()

   {

      ctg = new TreeMap();

      cust = new TreeMap();

   }


   public void showMenu()

   {

      int option;

      Scanner sc = new Scanner(System.in);


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

      System.out.println("1. Play");

      System.out.println("2. Instructions");

      System.out.println("3. Quit");
```

```
    System.out.print("\nChoose the option: ");

    option = sc.nextInt();

    switch (option)
    {
        case 1:
            playGame();
            break;
        case 2:
            instructGame();
            break;
        case 3:
            System.exit(0);
            break;
        default:
            showMenu();
            break;
    }
}

public void playGame()
{
    genCtgList();
    addItem();
    Scanner sc = new Scanner(System.in);
    int rnd, rnd2, size, count = 0, flag;
    String value=null;
    Random rd;
    StringBuffer guess = new StringBuffer();
    StringBuffer wrgString = new StringBuffer();
    StringBuffer rgtString = new StringBuffer();


    System.out.println("========================================================");
    System.out.println("Let's play the game.");
    rd = new Random();


    if(cust.size()==1)
    {
        rnd2 = rd.nextInt(cust.size());
    }
    else
    {
        rnd2 = rd.nextInt(cust.size()-1);
    }



    String cat = cust.get(cust.keySet().toArray()[rnd2]).toString();


    switch(cat)
    {
        case "Country": country = (Country)cust.get(cust.keySet().toArray()[rnd2]);
                    rnd = rd.nextInt(country.name.size());
                    value = country.name.get(rnd).toString();
                    break;
        case "Animal": animal = (Animal)cust.get(cust.keySet().toArray()[rnd2]);
                   rnd = rd.nextInt(animal.name.size());
                   value = animal.name.get(rnd).toString();
                   break;
        case "Fruit": fruit = (Fruit)cust.get(cust.keySet().toArray()[rnd2]);
                   rnd = rd.nextInt(fruit.name.size());
                   value = fruit.name.get(rnd).toString();
```

```
                break;
        default: System.out.println("Not Valid");
            break;
    }

    size = value.length();

    for (int j = 0; j < size; j++)
    {
        System.out.print("_ ");
        guess.append("_ ");
        rgtString.append(value.charAt(j) + " ");
    }



    while (true)
    {
        flag = 0;

        System.out.print("\n\nEnter your guess: ");
        String ch = sc.nextLine();
        ch = ch.toUpperCase();



        for (int i = 0; i < size; i++)
        {
            if (value.charAt(i) == ch.charAt(0))
            {
                String tmp = String.valueOf(value.charAt(i));
                guess.replace(i * 2, (i * 2) + 1, tmp);
                flag = 1;
                }

            }



            if (flag == 0)

            {

                wrgString.append(ch + ", ");

                System.out.println("\nMisses: " + wrgString);

                count++;

            }


            System.out.println(guess);


            if (guess.toString().equalsIgnoreCase(rgtString.toString()))

            {

                System.out.println("---------- Congrats :) You won ------------");

                break;

            }

        }

    }


    public void instructGame()

    {

        System.out.println("-------------Instructions---------------");

        System.out.println("1. You need to guess letters for the row of dashes");

        System.out.println("2: If you guess the correct letter, it will appear in the next statement");

        System.out.println("3: If you guess the incorrect letter, it will appear in the list of missed
letter.");

        showMenu();
```

```
}

public void genCtgList()
{
   ctg.put(1, new Country());
   ctg.put(2, new Animal());
   ctg.put(3, new Fruit());

   int size = ctg.size();
   System.out.println("=============================================================");
   System.out.println("The available categories are: ");
   for (int i = 0; i < size; i++)
   {
      System.out.println(ctg.keySet().toArray()[i] + " " + ctg.get(i + 1));
   }
}

public void genCustList(TreeMap obj)
{
   int size = obj.size();

   for (int i = 0; i < size; i++)
   {
      int input = (Integer) obj.keySet().toArray()[i];
      System.out.println(input + " " + cust.get(input));
   }
}

public void addItem()
{
   String choice;
```

```
        int input, count = 1;
        Scanner sc = new Scanner(System.in);

        do
        {
          System.out.println("================================================================");
          System.out.println("Select the category that you want to add (1/2/3): ");
          input = sc.nextInt();


          cust.put(input, ctg.get(input));
          System.out.println("================================================================");
          System.out.println("The customized list is: ");
          genCustList(cust);


          if (cust.size() < ctg.size())
          {
            do
            {
System.out.println("================================================================");
                System.out.print("Do you want to add more? (Y/N): ");
                choice = sc.next();
            }while(!(choice.toUpperCase().equals("N")||(choice.toUpperCase().equals("Y"))));
          }
          else
          {
            System.out.println("================================================================");
            System.out.println("You cannot add more categories");
            break;
            }
            count++;
        } while (choice.toUpperCase().equals("Y"));

        System.out.println("==========================================================");
        System.out.println("The final list is: ");
        genCustList(cust);
    }

    public static void main(String[] args)
    {
        ListCreator lc = new ListCreator();
        lc.showMenu();
    }
}
```

```
D:\GowthamanJavaTest\4>java ListCreator
---------Menu---------
1. Play
2. Instructions
3. Quit

Choose the option: 2
-------------Instructions---------------
1. You need to guess letters for the row of dashes
2: If you guess the correct letter, it will appear in the next statement
3: If you guess the incorrect letter, it will appear in the list of missed letter.
---------Menu---------
1. Play
2. Instructions
3. Quit
```

```
Choose the option: 1

=========================================================

The available categories are:

1 Country

2 Animal

3 Fruit

=========================================================

Select the category that you want to add (1/2/3):

1

=========================================================

The customized list is:

1 Country

=========================================================

Do you want to add more? (Y/N): N

=========================================================

The final list is:

1 Country

=========================================================

Let's play the game.

_ _ _ _ _ _ _


Enter your guess: INDIA

_ _ _ _ _ I _




Enter your guess: A

A _ _ _ _ I A




Enter your guess: S
```

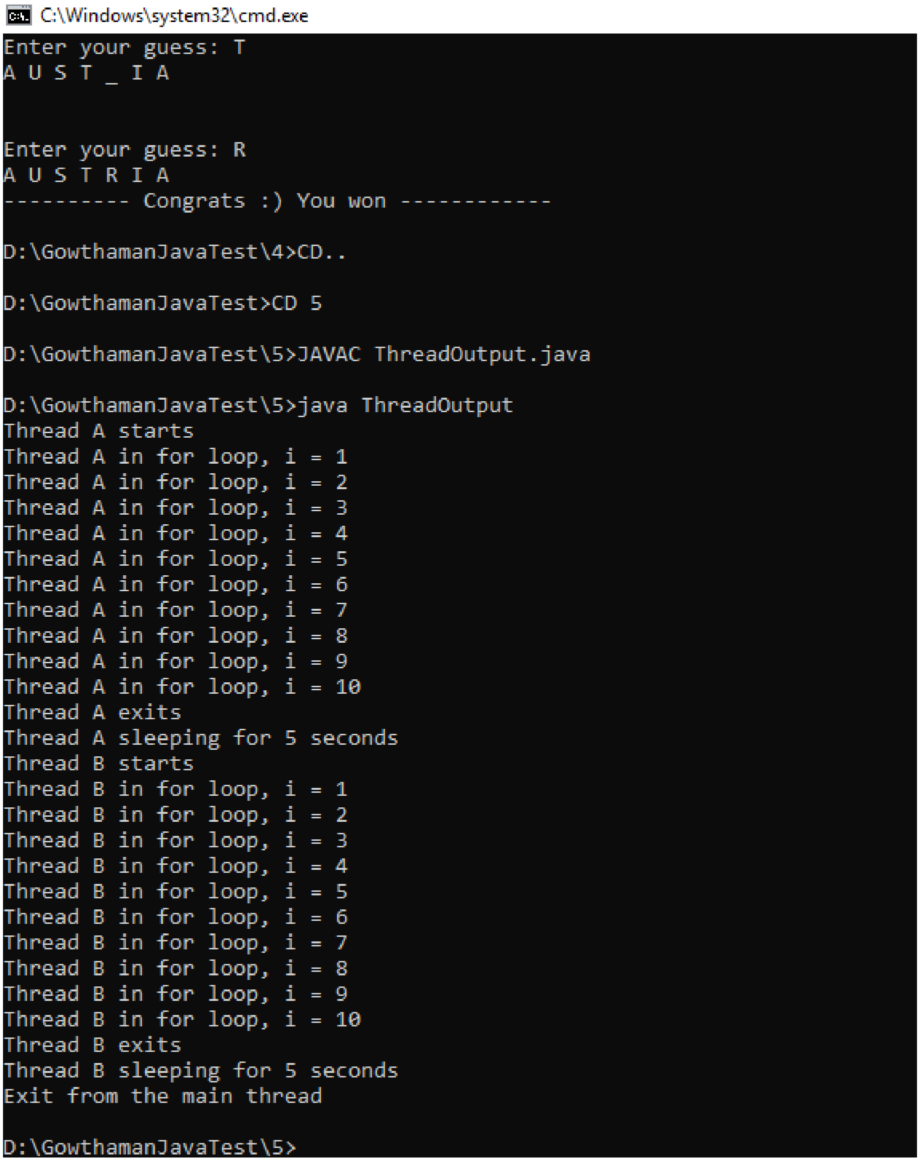
```
A _ S _ _ I A



Enter your guess: U

A U S _ _ I A



Enter your guess: T

A U S T _ I A


Enter your guess: R

A U S T R I A

---------- Congrats :) You won ------------
```

Chapter 5

Exercise 1:

```
class ThreadA extends Thread {

   public void run() {

      System.out.println("Thread A starts");

      for (int i = 1; i <= 10; i++) {

         System.out.println("Thread A in for loop, i = " + i);

      }

      System.out.println("Thread A exits");



      System.out.println("Thread A sleeping for 5 seconds");

      try {

         this.sleep(5000);

      } catch (InterruptedException e) {

         System.out.println("Exception caught:" + e);

      }
```

```
    }
}
class ThreadB extends Thread {
    public void run() {
        System.out.println("Thread B starts");
        for (int i = 1; i <= 10; i++) {
            System.out.println("Thread B in for loop, i = " + i);
        }
        System.out.println("Thread B exits");

        System.out.println("Thread B sleeping for 5 seconds");
        try {
            this.sleep(5000);
        } catch (InterruptedException e) {
            System.out.println("Exception caught:" + e);
        }
    }
}
public class ThreadOutput {
    public static void main(String[] args) {
        ThreadA objA = new ThreadA();
        objA.start();

        try {
            objA.join();
        } catch (InterruptedException e) {
            System.out.println(e);
        }

        ThreadB objB = new ThreadB();
        objB.start();
        try {
           objB.join();
        } catch (InterruptedException e) {
           System.out.println(e);
        }
        System.out.println("Exit from the main thread");
    }
}
```

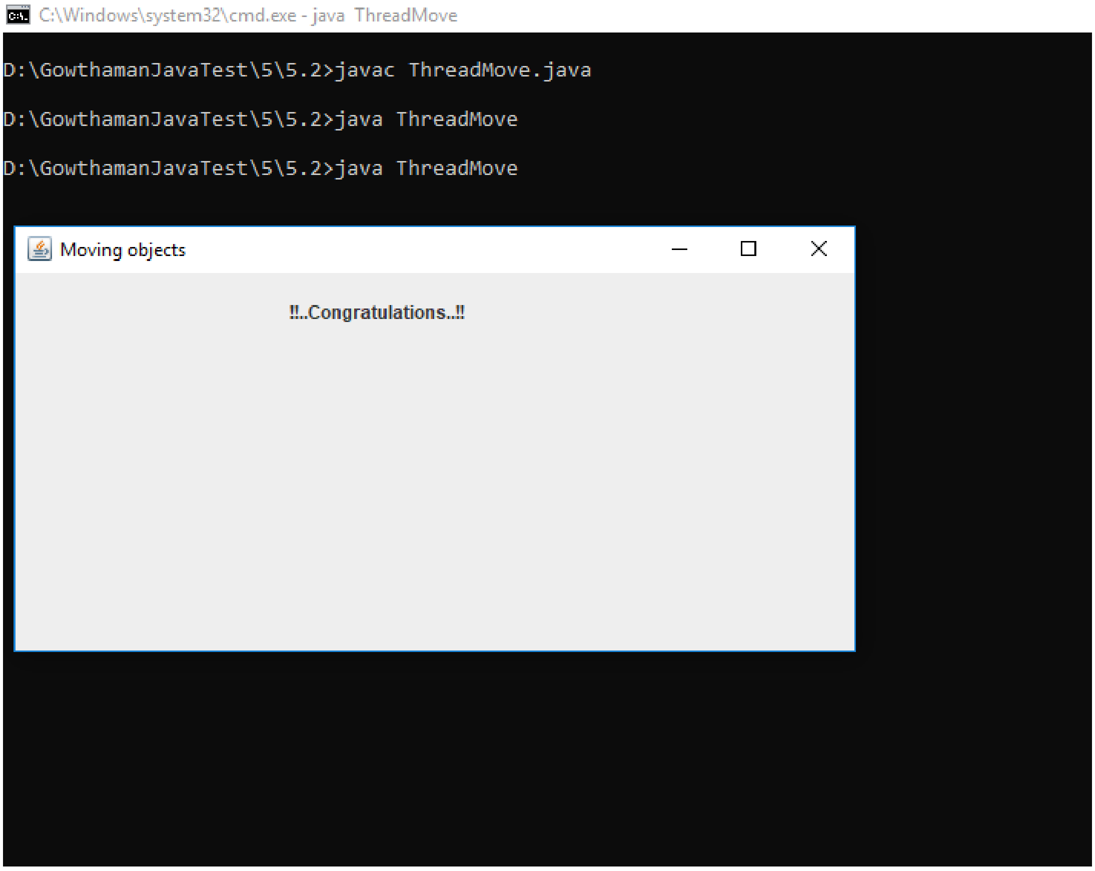


```
Enter your guess: T
A U S T _ I A


Enter your guess: R
A U S T R I A
---------- Congrats :) You won ------------

D:\GowthamanJavaTest\4>CD..

D:\GowthamanJavaTest>CD 5

D:\GowthamanJavaTest\5>JAVAC ThreadOutput.java

D:\GowthamanJavaTest\5>java ThreadOutput
Thread A starts
Thread A in for loop, i = 1
Thread A in for loop, i = 2
Thread A in for loop, i = 3
Thread A in for loop, i = 4
Thread A in for loop, i = 5
Thread A in for loop, i = 6
Thread A in for loop, i = 7
Thread A in for loop, i = 8
Thread A in for loop, i = 9
Thread A in for loop, i = 10
Thread A exits
Thread A sleeping for 5 seconds
Thread B starts
Thread B in for loop, i = 1
Thread B in for loop, i = 2
Thread B in for loop, i = 3
Thread B in for loop, i = 4
Thread B in for loop, i = 5
Thread B in for loop, i = 6
Thread B in for loop, i = 7
Thread B in for loop, i = 8
Thread B in for loop, i = 9
Thread B in for loop, i = 10
Thread B exits
Thread B sleeping for 5 seconds
Exit from the main thread

D:\GowthamanJavaTest\5>
```

Exercise 2:

```
import java.awt.Color;

import java.util.Random;

import javax.swing.*;

public class ThreadMove extends Thread {

   String ThreadName;

   JLabel l1;

   JFrame fr;

   public ThreadMove() {

      buildGUI();

   }

   public ThreadMove(String s) {

      super(s);

   }

   public void run() {

     while(true)

     {

      if (Thread.currentThread().getName().equals("RunnerA"))

      {

         runRunnerA();

         runRunnerB();

         }

     }

   }

   public void runRunnerA() {

      for (int i = 10; i < 260; i++) {

         l1.setBounds(i, 10, 120, 30);

         try {

            Thread.sleep(10);

         } catch (Exception e) {

            System.out.println(e);
         }
      }
   }
   public void runRunnerB() {
      for (int i = 260; i>10; i--) {
         l1.setBounds(i, 10, 120, 30);
         try {
            Thread.sleep(10);
         } catch (Exception e) {
            System.out.println(e);
         }
      }
   }
   public void buildGUI() {
      fr = new JFrame("Moving objects");
      fr.setVisible(true);
      fr.setSize(400, 200);
      fr.setLayout(null);
      l1 = new JLabel("!!..Congratulations..!!");
      l1.setSize(20, 20);
      l1.setBackground(Color.red);
      l1.setBounds(10, 10, 100, 100);
      fr.add(l1);

   }
   public static void main(String args[]) {
      ThreadMove obj = new ThreadMove();
      Thread Runner1 = new Thread(obj);
      Thread Runner2 = new Thread(obj);
      Runner1.setName("RunnerA");
      Runner2.setName("RunnerB");
```

```
       Runner1.start();
    try {
      Runner1.join();
    } catch (Exception e) {
    }
    Runner2.start();
    try {
      Runner2.join();
    } catch (Exception e) {
    }
  }
}
```



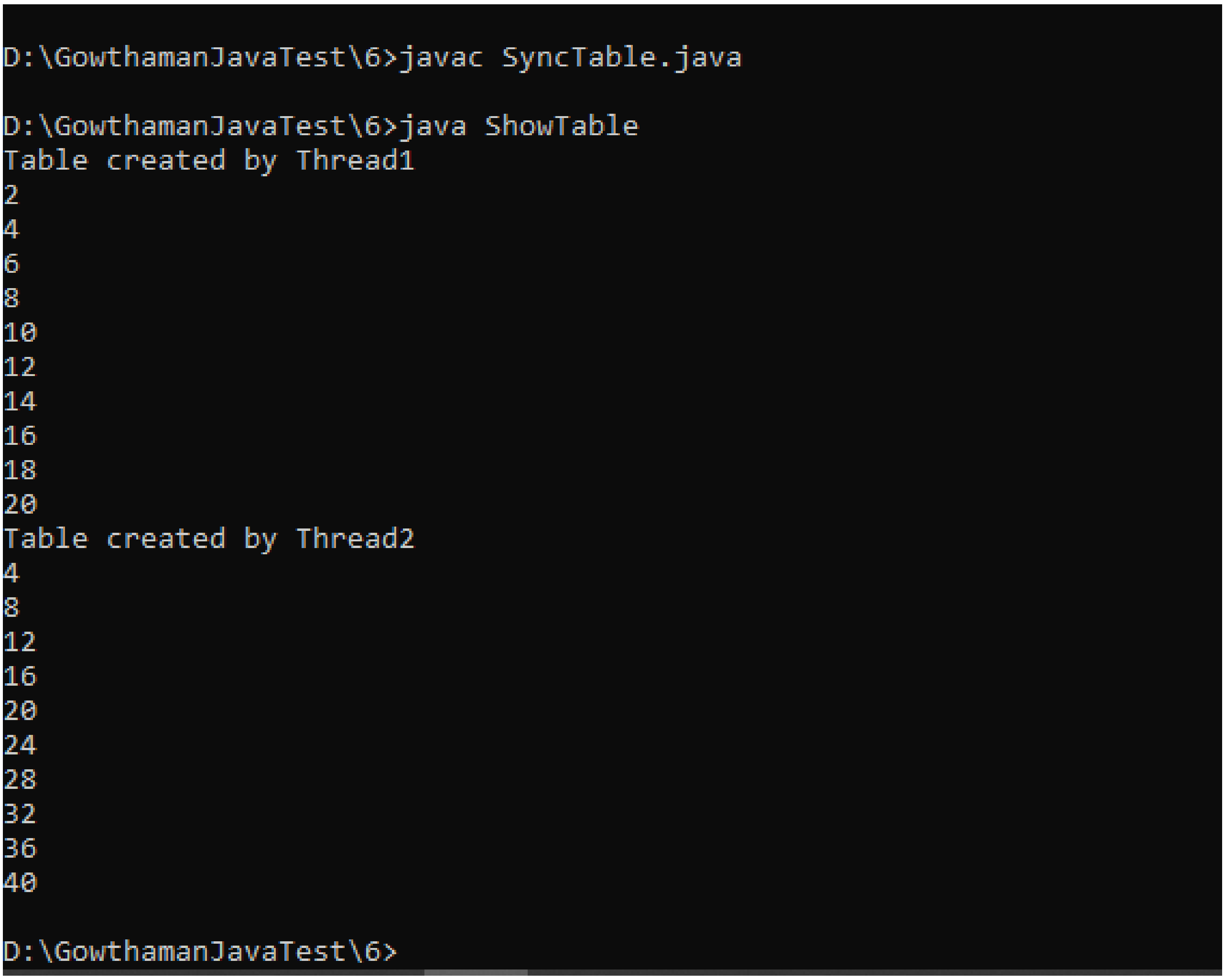
```
D:\GowthamanJavaTest\5\5.2>javac ThreadMove.java

D:\GowthamanJavaTest\5\5.2>java ThreadMove

D:\GowthamanJavaTest\5\5.2>java ThreadMove
```

‼..Congratulations..‼

Chapter 6

Exercise 1:

```
public class SyncTable {


    void print(int num) {
        synchronized (this) {
            Thread t = Thread.currentThread();
            String name = t.getName();
            System.out.println("Table created by " + name);
            for (int i = 1; i <= 10; i++) {
                System.out.println(num * i);
                try {
                t.sleep(1000);
                }
                catch(Exception e)
                {
                System.out.println("Interrupted");


                }
                }
        }
    }
}
class Thread1 extends Thread {
    SyncTable t;
    Thread1(SyncTable t) {
        this.t = t;
    }
    public void run() {
        t.print(2);
    }
}
class Thread2 extends Thread {
    SyncTable t;
    Thread2(SyncTable t) {
        this.t = t;
    }
    public void run() {
        t.print(4);
    }
}
class ShowTable {
    public static void main(String args[]) {
        SyncTable obj = new SyncTable();
        Thread1 newthread1 = new Thread1(obj);
        newthread1.setName("Thread1");
        Thread2 newthread2 = new Thread2(obj);
        newthread2.setName("Thread2");
        newthread1.start();
        newthread2.start();
    }
}
```

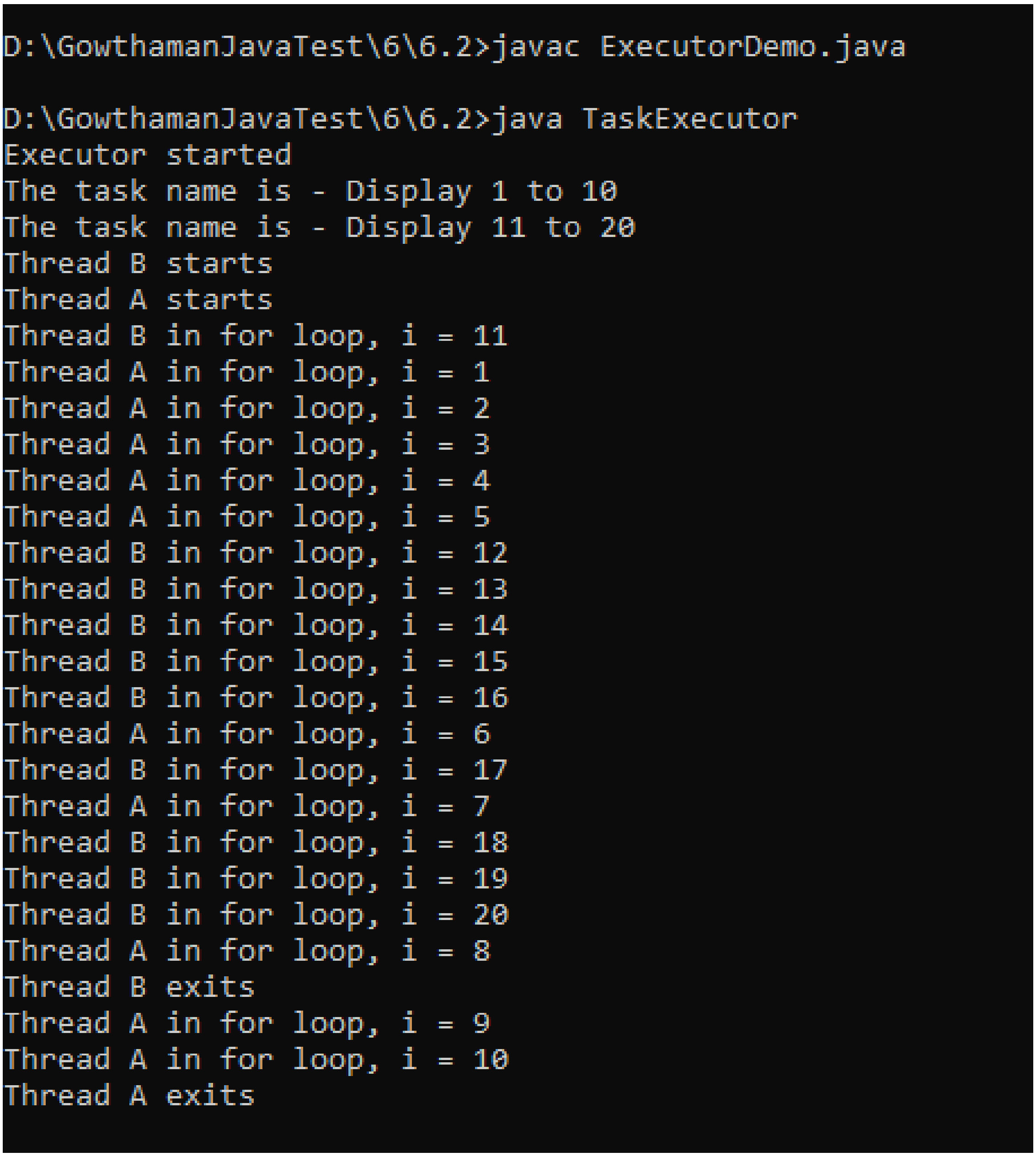


```
D:\GowthamanJavaTest\6>javac SyncTable.java

D:\GowthamanJavaTest\6>java ShowTable
Table created by Thread1
2
4
6
8
10
12
14
16
18
20
Table created by Thread2
4
8
12
16
20
24
28
32
36
40

D:\GowthamanJavaTest\6>
```

Exercise 2:

```
import java.util.concurrent.Executor;

import java.util.concurrent.ExecutorService;

import java.util.concurrent.Executors;


class Task implements Runnable {


   String taskname;


   public Task(String name) {

      taskname = name;

   }


   public void run() {

      System.out.println("The task name is - " + taskname);

      if (taskname.equals("Display 1 to 10")) {

         System.out.println("Thread A starts");

         for (int i = 1; i <= 10; i++) {

            System.out.println("Thread A in for loop, i = " + i);

         }

         System.out.println("Thread A exits");




      } else if (taskname.equals("Display 11 to 20")) {

         System.out.println("Thread B starts");

         for (int i = 11; i <= 20; i++)

         {

            System.out.println("Thread B in for loop, i = " + i);

         }
```

```
            System.out.println("Thread B exits");
        }
    }
}
class TaskExecutor {
    public static void main(String a[]) {
        Task task1 = new Task("Display 1 to 10");
        Task task2 = new Task("Display 11 to 20");
        ExecutorService threadexecutor = Executors.newCachedThreadPool();
        System.out.println("Executor started");
        try {
        threadexecutor.execute(task1);
        threadexecutor.execute(task2);
        }
        finally
        {
        threadexecutor.shutdown();
        }

        }
}
```



```
D:\GowthamanJavaTest\6\6.2>javac ExecutorDemo.java

D:\GowthamanJavaTest\6\6.2>java TaskExecutor
Executor started
The task name is - Display 1 to 10
The task name is - Display 11 to 20
Thread B starts
Thread A starts
Thread B in for loop, i = 11
Thread A in for loop, i = 1
Thread A in for loop, i = 2
Thread A in for loop, i = 3
Thread A in for loop, i = 4
Thread A in for loop, i = 5
Thread B in for loop, i = 12
Thread B in for loop, i = 13
Thread B in for loop, i = 14
Thread B in for loop, i = 15
Thread B in for loop, i = 16
Thread A in for loop, i = 6
Thread B in for loop, i = 17
Thread A in for loop, i = 7
Thread B in for loop, i = 18
Thread B in for loop, i = 19
Thread B in for loop, i = 20
Thread A in for loop, i = 8
Thread B exits
Thread A in for loop, i = 9
Thread A in for loop, i = 10
Thread A exits
```

Chapter 7

Exercise 1:

```
import java.io.*;
import java.util.*;
public class BookDetails {
   String bookName,authorName,price,choice,line;
   char choice_c;
    public void writeDetails(){
        do{
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the Book Name:");
        bookName=sc.next();
        System.out.println("Enter the Author Name:");
        authorName=sc.next();
        System.out.println("Enter the Price:");
        price=sc.next();
        try(FileWriter fw=new FileWriter("BookDetails.txt",true)){
            fw.write(bookName+"\t"+authorName+"\t"+price+"\n");

        }catch(Exception e){
            System.out.println(e);
        }
        System.out.println("Do you want to add more records (y/n)");
        choice=sc.next().toLowerCase();
        choice_c=choice.charAt(0);
        }while(choice_c=='y');
        System.out.println("---------------------------------------");
        displayMenu();
    }
```

```
public void displayDetails(){
 try (BufferedReader br = new BufferedReader(new FileReader("BookDetails.txt"))) {{
    while ((line = br.readLine())!= null) {
      System.out.println(line);
    }
    System.out.println("---------------------------------------");
    }
  }catch(Exception e){
    System.out.println(e);
  }
  displayMenu();
}
public void displayMenu(){
  Scanner sc=new Scanner(System.in);
  System.out.println("1. Enter Book Details");
  System.out.println("2. View All Book Details");
  System.out.println("3. Exit");
  System.out.println("Enter your choice");
  int i=sc.nextInt();
  System.out.println("---------------------------------------");
  switch(i){
    case 1: writeDetails();
        break;
    case 2: displayDetails();
        break;
    case 3: System.exit(0);
        break;
    default: System.out.println("Please enter a valid input");
         displayMenu();
  }
}
```

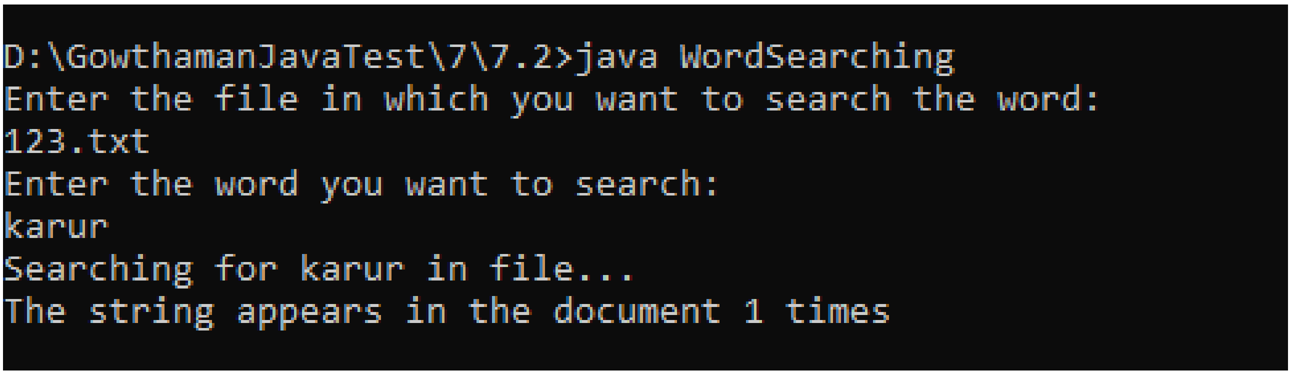
```
   public static void main(String args[]){


       BookDetails obj=new BookDetails();

       obj.displayMenu();

   }

}

D:\GowthamanJavaTest\7\7.1>java BookDetails

1. Enter Book Details

2. View All Book Details

3. Exit

Enter your choice

2

---------------------------------------

java.io.FileNotFoundException: BookDetails.txt (The system cannot find the file specified)

1. Enter Book Details

2. View All Book Details

3. Exit

Enter your choice

1

---------------------------------------

Enter the Book Name:

java

Enter the Author Name:

balagurusamy

Enter the Price:

300

Do you want to add more records (y/n)

y

Enter the Book Name:

c

Enter the Author Name:
```

```
Raman

Enter the Price:

230

Do you want to add more records (y/n)

n

---------------------------------------

1. Enter Book Details

2. View All Book Details

3. Exit

Enter your choice

2

--------------------------------------

java    balagurusamy    300

c       Raman   230

--------------------------------------

1. Enter Book Details

2. View All Book Details

3. Exit

Enter your choice

3

===========================
```

Exercise 2:

```
import java.io.BufferedReader;

import java.io.File;

import java.io.FileReader;

import java.io.IOException;

import java.util.Scanner;


public class WordSearching {


  public static void main(String args[]) {

    Scanner input1 = new Scanner(System.in);

      System.out.println("Enter the file in which you want to search the word: ");

      String stringSearch1 = input1.nextLine();

    File f=new File(stringSearch1);

    try (BufferedReader bf = new BufferedReader(new FileReader(f))) {

      Scanner input = new Scanner(System.in);

      System.out.println("Enter the word you want to search: ");

      String stringSearch = input.nextLine();

      int linecount = 0;

      int linecount1 = 0;

      String line;

      System.out.println("Searching for " + stringSearch + " in file...");

      while ((line = bf.readLine()) != null) {

        String txt[] = line.split(" ");

        for (int i = 0; i < txt.length; i++) {

          if (txt[i].equals(stringSearch)) {

            linecount++;

          }

        }

      }
```

```
            int i = linecount + linecount1;

            System.out.println("The string appears in the document " + i+" times");



        } catch (IOException e) {

            System.out.println("IO Error Occurred: " + e.toString());

        }

    }

}
```



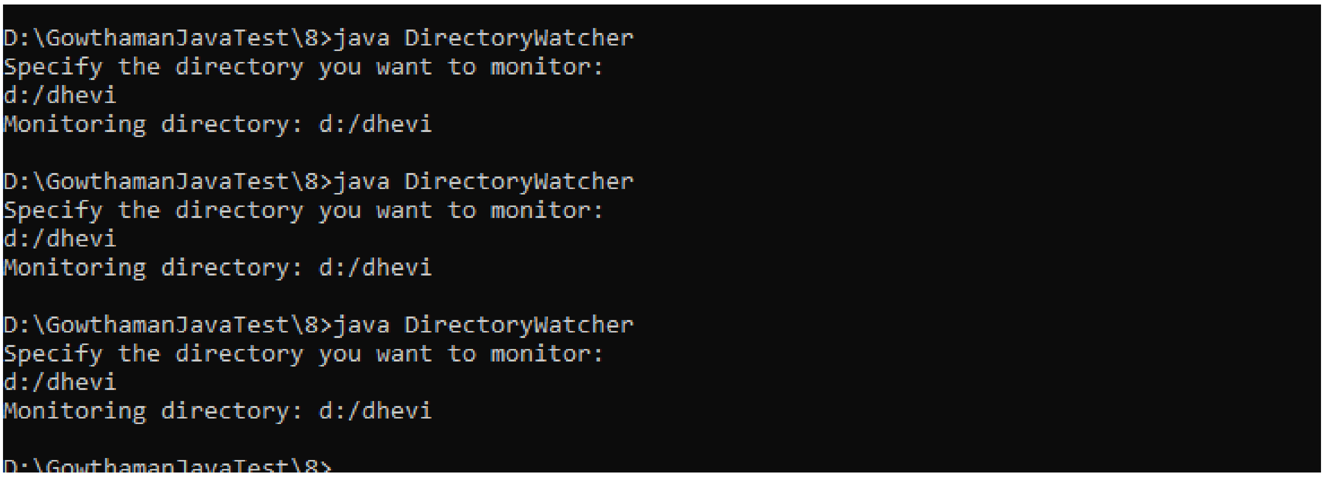
```
D:\GowthamanJavaTest\7\7.2>java WordSearching
Enter the file in which you want to search the word:
123.txt
Enter the word you want to search:
karur
Searching for karur in file...
The string appears in the document 1 times
```

Chapter – 8

Exercise 1:

```
import java.nio.file.WatchEvent.*;

import java.io.BufferedWriter;

import java.io.IOException;

import java.nio.charset.Charset;

import java.nio.file.FileSystems;

import java.nio.file.Files;

import java.nio.file.Path;

import java.nio.file.Paths;

import java.nio.file.StandardOpenOption;

import java.nio.file.WatchKey;

import java.nio.file.WatchService;

import static java.nio.file.StandardWatchEventKinds.*;

import java.nio.file.WatchEvent;

import java.text.SimpleDateFormat;

import java.util.Calendar;

import java.util.Scanner;

import javax.swing.JFrame;

import javax.swing.JLabel;


public class DirectoryWatcher {

   JFrame f;

   JLabel l;


   String directoryname;

   private Path path = null;

   Kind copy;

   WatchEvent eventcopy;
```

```
    WatchService watchService;

    private void initializeService()
    {
        path = Paths.get(directoryname);
        System.out.println("Monitoring directory: "+directoryname);
        try {
            watchService = FileSystems.getDefault().newWatchService();
            path.register(watchService, ENTRY_CREATE, ENTRY_DELETE);   // register the watch service
for the path.
        } catch (IOException e) {
            System.out.println("IOException" + e.getMessage());
        }
    }

    private void monitorDirectory() throws IOException {
        WatchKey key = null;
        String timeStamp = new
SimpleDateFormat("ddMMyyyy_HHmmss").format(Calendar.getInstance().getTime());
        Path file = Paths.get("D:/DirectoryLog_"+timeStamp+".txt");
        Charset charset = Charset.forName("US-ASCII");
        try (BufferedWriter writer = Files.newBufferedWriter(file, charset,
StandardOpenOption.CREATE, StandardOpenOption.APPEND)) {
            while (true) {  // infinite loop to monitor changes
                try {
                    key = watchService.take(); // get watch key
                    for (WatchEvent event : key.pollEvents()) {
                        Kind kind = event.kind(); // get event kind
                        copy = kind;
                        eventcopy = event;
                        writer.newLine();
                        writer.append("Log: The event that occurred on " + eventcopy.context().toString() + " is
" + copy + "\n");

                        writer.newLine();

                        writer.flush();

                        System.out.println("The event that occurred on " + event.context().toString() + " is " +
kind);

                    }

                } catch (InterruptedException e) {

                    System.out.println("InterruptedException: " + e.getMessage());

                }

                boolean reset = key.reset();

                if (!reset) {

                    break;

                }


            }

        }

    }

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        DirectoryWatcher watchservicedemo = new DirectoryWatcher();

        System.out.println("Specify the directory you want to monitor:");

        watchservicedemo.directoryname = sc.nextLine();

        watchservicedemo.initializeService();

        try {

            watchservicedemo.monitorDirectory();


        } catch (IOException ioe) {

            System.out.println(ioe);

        }

    }

}
```



```
D:\GowthamanJavaTest\8>java DirectoryWatcher
Specify the directory you want to monitor:
d:/dhevi
Monitoring directory: d:/dhevi

D:\GowthamanJavaTest\8>java DirectoryWatcher
Specify the directory you want to monitor:
d:/dhevi
Monitoring directory: d:/dhevi

D:\GowthamanJavaTest\8>java DirectoryWatcher
Specify the directory you want to monitor:
d:/dhevi
Monitoring directory: d:/dhevi

D:\GowthamanJavaTest\8>
```