Programming in Java

Assignment – 2

Name : Dharun Balaji P

Roll Number : 2016108

Department : Electronics and Instrumentation Engineering

Problem statement:

Create a Java applet program that allows users to view a list of tasks. The application should allow users to add tasks, mark tasks as complete, and delete tasks. The applet should display tasks in a user-friendly manner and provide options to filter tasks based on their completion status or search for specific tasks by their name.

Description:

The task manager application is designed to help users keep track of their tasks and designed to provide users with a convenient way to view their tasks. The applet will present a graphical user interface where users can interact with various features of the application. The main functionality of the applet includes displaying a list of tasks, filtering tasks based on completion status, and searching for tasks by name.

Concepts used:

1. **Java Applet:** The program is implemented as a Java applet, utilizing the java.applet.Applet class. Applets are designed to run within a web browser and provide interactive functionality.
2. **AWT Components:** The applet uses AWT (Abstract Window Toolkit) components to create the user interface elements. This includes components such as List, Button, and Text Field for displaying tasks, filtering tasks, and accepting user input, respectively.
3. **Event Handling:** Event listeners and event handling mechanisms are used to capture user actions and perform appropriate actions in response. For example, button clicks or text input events trigger actions like filtering tasks or searching for tasks.
4. **Task Management**: The applet manages tasks as a collection and provides methods to filter tasks based on completion status or search for tasks by name. The tasks may be stored in an Array List or another appropriate data structure.

Program Code:

```
import java.awt.*;
import java.awt.event.*;
import java.util.ArrayList;


/*
<Applet code=TaskManagerApplet width=1680 height=900>
</Applet>
*/


public class TaskManagerApplet extends java.applet.Applet {
   private TextField taskNameField;
   private Button addButton, completeButton, deleteButton;
   private List taskList;
   private ArrayList<String> tasks;


   public void init() {
      setLayout(new BorderLayout());


      tasks = new ArrayList<>();


      Panel topPanel = new Panel();
      topPanel.setLayout(new FlowLayout());


      Label taskNameLabel = new Label("Task Name:");
      taskNameField = new TextField(15);
      addButton = new Button("Add");
      completeButton = new Button("Complete");
      deleteButton = new Button("Delete");


      topPanel.add(taskNameLabel);
      topPanel.add(taskNameField);
      topPanel.add(addButton);
      topPanel.add(completeButton);
```

```
topPanel.add(deleteButton);


taskList = new List();
add(taskList, BorderLayout.CENTER);
add(topPanel, BorderLayout.NORTH);


addButton.addActionListener(new ActionListener() {
   public void actionPerformed(ActionEvent e) {
      String taskName = taskNameField.getText();
      if (!taskName.isEmpty()) {
         tasks.add(taskName);
         taskList.add(taskName);
         taskNameField.setText("");
      }
   }
});


completeButton.addActionListener(new ActionListener() {
   public void actionPerformed(ActionEvent e) {
      int selectedIndex = taskList.getSelectedIndex();
      if (selectedIndex != -1) {
         tasks.remove(selectedIndex);
         taskList.remove(selectedIndex);
      }
   }
});


deleteButton.addActionListener(new ActionListener() {
   public void actionPerformed(ActionEvent e) {
      int selectedIndex = taskList.getSelectedIndex();
      if (selectedIndex != -1) {
         tasks.remove(selectedIndex);
         taskList.remove(selectedIndex);
      }
```

```
        }
    });
  }
}
```

Screenshots: