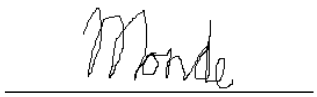
<table>
<tr><td colspan="9">Faculty of Information Technology</td></tr>
<tr><td rowspan="5">I declare that I am familiar with, and will abide to the Examination rules of CTU<br><br>  <br>Signature</td><td colspan="8">SUBJECT NAME: Advanced Java<br>SUBJECT CODE: JD522</td></tr>
<tr><td colspan="4">Formative Assessment 2<br>Duration:<br>Date:<br>Total Marks: 100<br>Total pages: 32</td><td colspan="4">Examiner: Junior Manganyi<br><br>Moderator:</td></tr>
<tr><td colspan="8">Student number</td></tr>
<tr><td>2</td><td>0</td><td>2</td><td>3</td><td>1</td><td>7</td><td>3</td><td>3</td><td></td></tr>
<tr><td colspan="4">Surname:<br>Mkhize</td><td colspan="3">Initials:<br>ME</td><td>/</td><td>%</td></tr>
</table>

## Contents

## Formative Question(s)

### Question 1:

You are tasked with creating a Java GUI-based Task Manager application that allows users to manage their tasks, categorize them, and store the information in a SQLite database. The application should provide features such as adding tasks, marking tasks as completed, and viewing tasks based on categories.

Unit 5: I/O and NIO (25 marks)

- Implement a GUI to list tasks from the SQLite database. (5 marks)
- Allow users to save tasks to a text file using OutputStream. (5 marks)
- Implement a mechanism to read tasks from the text file using InputStream. (5 marks)
- Display file properties like size and creation date using NIO. (5 marks)
- Provide an option to export task data to a CSV file using NIO. (5 marks)

Unit 6: Generics and Collections (20 marks)

- Design a task class that uses Generics to store task information. (5 marks)
- Use ArrayList to manage the list of tasks. (5 marks)
- Implement a filter mechanism to search for tasks based on user-defined criteria. (5 marks)
- Categorize tasks using HashMap to organize them based on user-defined categories. (5 marks)

Unit 7: Inner Classes (15 marks)

- Create an inner class to handle GUI components for task entry. (5 marks)
- Design an inner class to manage task categories and their corresponding actions. (5 marks)
- Implement a nested panel structure using inner panels for better organization. (5 marks)

Unit 8: JDBC (20 marks)

- Integrate a SQLite database with the application using JDBC. (5 marks)
- Design a database schema to store task information, including task name, description, completion status, and category. (5 marks)
- Implement functionalities to insert, update, and retrieve task data from the database. (5 marks)
- Display tasks in the GUI retrieved from the database. (5 marks)

Overall Design and Usability (20 marks)

- Design an intuitive and user-friendly GUI for the Task Manager application. (5 marks)
- Provide appropriate labels, buttons, and input fields for adding, viewing, and managing tasks. (5 marks)
- Implement error handling for input validation and database operations. (5 marks)
- Ensure a smooth and responsive user experience. (5 marks)

Answers:

*JD522FA2.java Code:*

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-
default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to
edit this template
 */
package org.mondemkhize.jd522fa2;

import javax.swing.JDialog;
import javax.swing.JOptionPane;

/**
 *
 * @author monde
 */
public class JD522FA2 extends javax.swing.JFrame {

    /**
     * Creates new form JD522FA2
     */
    public JD522FA2() {
        initComponents();
        this.setLocationRelativeTo(null);
    }

    /**
     * This method is called from within the constructor to initialize the
form.
     * WARNING: Do NOT modify this code. The content of this method is
always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-
BEGIN:initComponents
    private void initComponents() {

        NewTaskBtn = new javax.swing.JButton();
        ViewTaskBtn = new javax.swing.JButton();
        CloseBtn = new javax.swing.JButton();
        TitleLbl = new javax.swing.JLabel();
        textTasks = new javax.swing.JButton();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        setTitle("TaskManager");

        NewTaskBtn.setText("New Task");
        NewTaskBtn.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                NewTaskBtnActionPerformed(evt);
            }
        });

        ViewTaskBtn.setText("View Tasks");
        ViewTaskBtn.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                ViewTaskBtnActionPerformed(evt);
            }
        });
```

```
        CloseBtn.setText("Close me!");
        CloseBtn.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                CloseBtnActionPerformed(evt);
            }
        });

        TitleLbl.setText("Task Manager");

        textTasks.setText("View Text Tasks");
        textTasks.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                textTasksActionPerformed(evt);
            }
        });

        javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
                    .addGroup(layout.createSequentialGroup()
                        .addGap(161, 161, 161)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAI
LING)
                            .addComponent(TitleLbl)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAI
LING, false)
                                .addComponent(ViewTaskBtn,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
                                .addComponent(NewTaskBtn,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
                                .addComponent(CloseBtn,
javax.swing.GroupLayout.Alignment.LEADING,
javax.swing.GroupLayout.PREFERRED_SIZE, 102,
javax.swing.GroupLayout.PREFERRED_SIZE))))
                    .addGroup(layout.createSequentialGroup()
                        .addGap(153, 153, 153)
                        .addComponent(textTasks,
javax.swing.GroupLayout.PREFERRED_SIZE, 119,
javax.swing.GroupLayout.PREFERRED_SIZE)))
                .addContainerGap(188, Short.MAX_VALUE))
        );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
                .addGap(20, 20, 20)
                .addComponent(TitleLbl)
```

```
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 72,
Short.MAX_VALUE)
                .addComponent(NewTaskBtn)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(ViewTaskBtn)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(textTasks)
                .addGap(31, 31, 31)
                .addComponent(CloseBtn)
                .addGap(95, 95, 95))
        );

        pack();
    }// </editor-fold>//GEN-END:initComponents

    private void CloseBtnActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_CloseBtnActionPerformed
        // TODO add your handling code here:
        this.dispose();
    }//GEN-LAST:event_CloseBtnActionPerformed

    private void ViewTaskBtnActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_ViewTaskBtnActionPerformed
        // TODO add your handling code here:
        /*Task task = new Task();
        task.readFromFile(this);*/
        ViewForm view = new ViewForm();
        view.setVisible(true);
        this.dispose();
    }//GEN-LAST:event_ViewTaskBtnActionPerformed

    private void NewTaskBtnActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_NewTaskBtnActionPerformed
        // TODO add your handling code here:
        NewTaskForm newTaskForm = new NewTaskForm(this, "", "", "", false);
        newTaskForm.setVisible(true);
        this.dispose();
    }//GEN-LAST:event_NewTaskBtnActionPerformed

    private void textTasksActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_textTasksActionPerformed
        // TODO add your handling code here:
        TextViewForm tvf = new TextViewForm();
        tvf.setVisible(true);
        this.dispose();
    }//GEN-LAST:event_textTasksActionPerformed

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel
setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with
the default look and feel.
         * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
```

```
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(JD522FA2.class.getName()).log(java.util.
logging.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(JD522FA2.class.getName()).log(java.util.
logging.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(JD522FA2.class.getName()).log(java.util.
logging.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(JD522FA2.class.getName()).log(java.util.
logging.Level.SEVERE, null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new JD522FA2().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton CloseBtn;
    private javax.swing.JButton NewTaskBtn;
    private javax.swing.JLabel TitleLbl;
    private javax.swing.JButton ViewTaskBtn;
    private javax.swing.JButton textTasks;
    // End of variables declaration//GEN-END:variables
}
```

## *NewTaskForm.java Code:*

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-
default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to
edit this template
 */
package org.mondemkhize.jd522fa2;

import java.awt.Component;
import java.awt.event.ActionEvent;
import javax.swing.JOptionPane;

/**
 *
```

```
 * @author monde
 */
public class NewTaskForm extends javax.swing.JFrame {

    /**
     * Creates new form NewTaskForm
     * @param comp
     * @param taskName
     * @param Category
     * @param description
     * @param compleationState
     */
    public NewTaskForm(Component comp, String taskName, String Category,
String description, boolean compleationState) {
        initComponents(comp);
        this.setLocationRelativeTo(null);
        if(comp instanceof ViewForm){
            //System.out.println("edit yay");
            this.setTitle("Edit Task");
            this.AddBtn.setText("Update");
            this.TaskNameField.setText(taskName);
            this.TaskNameField.setEditable(false);
            this.CategoryField.setText(Category);
            this.DecriptionField.setText(description);
            this.CompletionBox.setSelected(compleationState);

        }

        /*if(comp.getName()){
            System.out.println("cool");
        }*/

    }

    /**
     * This method is called from within the constructor to initialize the
form.
     * WARNING: Do NOT modify this code. The content of this method is
always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-
BEGIN:initComponents
    private void initComponents(Component comp) {

        TaskNameField = new javax.swing.JTextField();
        TaskNameLabel = new javax.swing.JLabel();
        DescriTaskLabel = new javax.swing.JLabel();
        DecriptionPane = new javax.swing.JScrollPane();
        DecriptionField = new javax.swing.JTextArea();
        CompletionBox = new javax.swing.JCheckBox();
        AddBtn = new javax.swing.JButton();
        CanclBtn = new javax.swing.JButton();
        CategoryLabel = new javax.swing.JLabel();
        CategoryField = new javax.swing.JTextField();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        setTitle("NewTask");
```

```
        TaskNameLabel.setText("Task Name:");

        DescriTaskLabel.setText("Description:");

        DecriptionField.setColumns(20);
        DecriptionField.setRows(5);
        DecriptionPane.setViewportView(DecriptionField);

        CompletionBox.setText("Complete");

        AddBtn.setText("Add");
        AddBtn.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                AddBtnActionPerformed(evt, comp);
            }
        });

        CanclBtn.setText("Cancel");
        CanclBtn.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                CanclBtnActionPerformed(evt, comp);
            }
        });

        CategoryLabel.setText("Category:");

        javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addGap(30, 30, 30)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAI
LING)
                    .addGroup(layout.createSequentialGroup()
                        .addComponent(DescriTaskLabel)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                        .addComponent(DecriptionPane,
javax.swing.GroupLayout.PREFERRED_SIZE, 294,
javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addGap(0, 105, Short.MAX_VALUE))
                    .addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
                            .addGroup(layout.createSequentialGroup()
                                .addGap(13, 13, 13)
                                .addComponent(CategoryLabel))
                            .addComponent(TaskNameLabel,
javax.swing.GroupLayout.Alignment.TRAILING))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
                            .addComponent(TaskNameField)
                            .addComponent(CategoryField))
```

```
                        .addGap(127, 127, 127))))
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
                .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
                    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
                            .addComponent(CanclBtn)
                            .addComponent(AddBtn))
                        .addGap(75, 75, 75))
                    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
                        .addComponent(CompletionBox)
                        .addGap(199, 199, 199))))
        );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addGap(17, 17, 17)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASE
LINE)
                    .addComponent(TaskNameField,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(TaskNameLabel))
                .addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASE
LINE)
                    .addComponent(CategoryLabel)
                    .addComponent(CategoryField,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGap(21, 21, 21)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
                    .addComponent(DecriptionPane,
javax.swing.GroupLayout.PREFERRED_SIZE, 143,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(DescriTaskLabel))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(CompletionBox)
                .addGap(18, 18, 18)
                .addComponent(AddBtn)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                .addComponent(CanclBtn)
                .addContainerGap(39, Short.MAX_VALUE))
        );
```

```
        pack();
    }// </editor-fold>//GEN-END:initComponents

    private void AddBtnActionPerformed(ActionEvent evt, Component comp) {
        Task currentTask = new Task();
        currentTask.setName(this.TaskNameField.getText());
        String taskName = currentTask.getName(); //New variable made to be
used in database and file

        currentTask.setDescription(this.DecriptionField.getText());
        String description = currentTask.getDescription();//New variable
made to be used in database and file

        currentTask.setCategory(this.CategoryField.getText());
        String category = currentTask.getCategory();//New variable made to
be used in database and file

        currentTask.setCompletionState(this.CompletionBox.isSelected());
        String completionState =
String.valueOf(currentTask.isCompletionState());
        if(comp instanceof JD522FA2){
            TaskDB insert = new TaskDB();
            insert.insert(taskName, category, description, completionState,
this);

        currentTask.writeToFile(taskName, category, description,
completionState, this);
        this.TaskNameField.setText("");
        this.CategoryField.setText("");
        this.DecriptionField.setText("");
        this.CompletionBox.setSelected(false);
        }
        else if(comp instanceof ViewForm){
            TaskDB update = new TaskDB();
            update.update(taskName, category, description, completionState,
this);
            ViewForm viewForm = new ViewForm();
            viewForm.setVisible(true);
            this.dispose();
        }
    }

    private void CanclBtnActionPerformed(java.awt.event.ActionEvent evt,
Component comp) {//GEN-FIRST:event_CanclBtnActionPerformed
        // TODO add your handling code here:
        if(comp instanceof JD522FA2){
            JD522FA2 mainwin = new JD522FA2();
            mainwin.setVisible(true);
            this.dispose();
        }else if(comp instanceof ViewForm){
            ViewForm mainwin = new ViewForm();
            mainwin.setVisible(true);
            this.dispose();
        }
    }//GEN-LAST:event_CanclBtnActionPerformed

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
```

```
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel
setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with
the default look and feel.
         * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(NewTaskForm.class.getName()).log(java.ut
il.logging.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(NewTaskForm.class.getName()).log(java.ut
il.logging.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(NewTaskForm.class.getName()).log(java.ut
il.logging.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(NewTaskForm.class.getName()).log(java.ut
il.logging.Level.SEVERE, null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new NewTaskForm(null, "", "", "", false).setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton AddBtn;
    private javax.swing.JButton CanclBtn;
    private javax.swing.JTextField CategoryField;
    private javax.swing.JLabel CategoryLabel;
    private javax.swing.JCheckBox CompletionBox;
    private javax.swing.JTextArea DecriptionField;
    private javax.swing.JScrollPane DecriptionPane;
    private javax.swing.JLabel DescriTaskLabel;
    private javax.swing.JTextField TaskNameField;
    private javax.swing.JLabel TaskNameLabel;
    // End of variables declaration//GEN-END:variables
}
```

*Task.java Code:*

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-
default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to
edit this template
 */
package org.mondemkhize.jd522fa2;

import com.opencsv.CSVWriter;
import java.awt.Component;
import java.io.BufferedWriter;
import java.io.File;
import java.nio.file.*;
import java.io.FileWriter;
import java.io.IOException;
import java.nio.file.attribute.FileTime;
import java.time.LocalDateTime;
import java.time.ZoneId;
import java.time.format.DateTimeFormatter;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import javax.swing.JOptionPane;
import javax.swing.JTable;

/**
 *
 * @author monde
 */
public class Task {
    private String Name;
    private String Description;
    private String Category;
    private boolean completionState;

    final HashMap<String, String> TaskDictionary = new HashMap();

    private ArrayList<String> taskNames = new ArrayList();

    public ArrayList<String> getTaskNames() {
        return taskNames;
    }

    public void setTaskNames(ArrayList<String> taskNames) {
        this.taskNames = taskNames;
    }

    public HashMap<String, String> getTaskDictionary() {
        TaskDictionary.put("TaskName", Name);
        TaskDictionary.put("Category", Category);
        TaskDictionary.put("Description", Description);
        TaskDictionary.put("CompletionState",
String.valueOf(completionState));
        return TaskDictionary;
    }

    public String getName() {
        return Name;
    }
```

```
    public void setName(String Name) {
        this.Name = Name;
    }

    public String getDescription() {
        return Description;
    }

    public void setDescription(String Description) {
        this.Description = Description;
    }

    public String getCategory() {
        return Category;
    }

    public void setCategory(String Category) {
        this.Category = Category;
    }

    public boolean isCompletionState() {
        return completionState;
    }

    public void setCompletionState(boolean completionState) {
        this.completionState = completionState;
    }

    public void writeToFile(String taskName, String category, String
description, String completionState, Component comp){
        BufferedWriter out;
        try {
            out = new BufferedWriter(new FileWriter("out.txt"));
            out.write(taskName + "\n" + category + "\n" + description +
"\n" + completionState);
            out.close();
            new JOptionPane().showMessageDialog(comp, "File saved
successfully");
        } catch (IOException ex) {
            new JOptionPane().showMessageDialog(comp, "IO Failed");
        }
    }

    public List<String> readFromFile(Component comp){
        Path filePath = Path.of("out.txt");
        try{
            List<String> contents = Files.readAllLines(filePath);
            //contents.forEach(line -> System.out.println(line));
            return contents;
        }catch (IOException ex) {
            new JOptionPane().showMessageDialog(comp, ex.getMessage());
            return null;
        }

    }

    public long fileSize(Component comp){
        Path path = Path.of("out.txt");
        try{
            long bytes = Files.size(path);
            return bytes;
```

```
        }catch(IOException exception){
            new JOptionPane().showMessageDialog(comp,
exception.getMessage());
            return 0;
        }
    }

    public String creationTime(Component comp){
        try{
            Path path = Path.of("out.txt");
            FileTime fileTime = (FileTime) Files.getAttribute(path,
"creationTime");
            LocalDateTime localDateTime =
fileTime.toInstant().atZone(ZoneId.systemDefault()).toLocalDateTime();

            return
localDateTime.format(DateTimeFormatter.ofPattern("dd/MM/yyyy HH:mm:ss"));
            //return fileTime;
        }catch (IOException ex) {
            new JOptionPane().showMessageDialog(comp, ex.getMessage());
            return "Never";
        }
    }

    public String lastModified(Component comp) {
        try {
            Path path = Path.of("out.txt");
            FileTime fileTime = Files.getLastModifiedTime(path);
            LocalDateTime localDateTime =
fileTime.toInstant().atZone(ZoneId.systemDefault()).toLocalDateTime();

            return
localDateTime.format(DateTimeFormatter.ofPattern("dd/MM/yyyy HH:mm:ss"));
            //return fileTime;
        } catch (IOException ex) {
            new JOptionPane().showMessageDialog(comp, ex.getMessage());
            return "Never";
        }
    }

    public void xportToCSV(String fileName, JTable table,Component comp){
        File file = new File(fileName);
        try {
            // create FileWriter object with file as parameter
            FileWriter outputfile = new FileWriter(file);

            // create CSVWriter with '|' as separator
            CSVWriter writer = new CSVWriter(outputfile, '|',
                    CSVWriter.NO_QUOTE_CHARACTER,
                    CSVWriter.DEFAULT_ESCAPE_CHARACTER,
                    CSVWriter.DEFAULT_LINE_END);

            // create a List which contains String array
            List<String[]> data = new ArrayList<String[]>();
            data.add(new String[]{"TaskName", "Category", "Description",
"Completon Status"});
            data.add(new String[]{table.getValueAt(table.getSelectedRow(),
0).toString(), table.getValueAt(table.getSelectedRow(), 1).toString(),
table.getValueAt(table.getSelectedRow(), 2).toString(),
table.getValueAt(table.getSelectedRow(), 3).toString()});
            writer.writeAll(data);
```

```
            // closing writer connection
            writer.close();
            new JOptionPane().showMessageDialog(comp, "Exported csv
successfully");
        } catch (IOException e) {
            // TODO Auto-generated catch block
            new JOptionPane().showMessageDialog(comp, e.getMessage());
        }
    }
}
```

*TaskDB.java Code:*

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-
default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to
edit this template
 */
package org.mondemkhize.jd522fa2;

import java.awt.Component;
import java.sql.*;
import java.util.ArrayList;
import java.util.Collections;
import java.util.Comparator;
import java.util.HashMap;
import java.util.List;
import javax.swing.JOptionPane;
import me.xdrop.fuzzywuzzy.FuzzySearch;
import me.xdrop.fuzzywuzzy.model.ExtractedResult;

/**
 *
 * @author monde
 */
public class TaskDB {
    final String OSname = System.getProperty("os.name");

    private ArrayList<Task> TasksList = new ArrayList();
    private ArrayList<HashMap<String, String>> TaskDictionList = new
ArrayList();
    private ArrayList<String> taskNames = new ArrayList();

    Comparator<HashMap<String, String>> categoryComparator = new
Comparator<HashMap<String, String>>() {

        @Override
        public int compare(HashMap<String, String> o1, HashMap<String,
String> o2) {
            // Get the distance and compare the distance.
            String firstValue = o1.get("Category");
            String secondValue = o2.get("Category");
            return firstValue.compareTo(secondValue);
        }
    };

    public ArrayList<HashMap<String, String>> getTaskDictionList() {
        return TaskDictionList;
    }
```

```
    public ArrayList<Task> getTasksList() {
        return TasksList;
    }

    public ArrayList<String> getTasksNames() {
        return taskNames;
    }

    public void clearLists(){
        TasksList.clear();
        taskNames.clear();
        TaskDictionList.clear();
    }

    private Connection connect (Component comp){
        Connection conn = null;

        try{
            if (OSname.equals("Windows 11")) {
                String connectionString =
"jdbc:sqlite:C:/sqlite/TaskManager.db";
                conn = DriverManager.getConnection(connectionString);
            } else {
                String connectionString =
"jdbc:sqlite:/home/monde/TaskManager";
                conn = DriverManager.getConnection(connectionString);
            }
        } catch (SQLException e) {
            new JOptionPane().showMessageDialog(comp,e.getMessage());
        }
        return conn;
    }
    public void insert(String Name, String Category, String Description,
String CompletionState, Component comp){
       String sql = "INSERT INTO TaskInfo(TaskName,Category, Description,
CompletionState) VALUES(?,?,?,?)" ;

       try(Connection conn = this.connect(comp);
           PreparedStatement pstmnt = conn.prepareStatement(sql)){
           pstmnt.setString(1,Name);
           pstmnt.setString(2, Category);
           pstmnt.setString(3, Description);
           pstmnt.setString(4, CompletionState);
           pstmnt.executeUpdate();
           new JOptionPane().showMessageDialog(comp, "Added task
successfullly");
       }catch (SQLException e){
           new JOptionPane().showMessageDialog(comp, e.getMessage());
       }
}

    public void selectAll(Component comp){
        String sql = "select * from TaskInfo";
        TasksList.clear();
        taskNames.clear();
        TaskDictionList.clear();
        try(Connection conn = this.connect(comp);
                Statement stmt = conn.createStatement();
                ResultSet rs = stmt.executeQuery(sql))
        {
```

```
            while(rs.next()){
                Task currentTask = new Task();

                currentTask.setName(rs.getString("TaskName"));
                currentTask.setCategory(rs.getString("Category"));
                currentTask.setDescription(rs.getString("Description"));
                boolean compState =
Boolean.parseBoolean(rs.getString("CompletionState"));

                currentTask.setCompletionState(compState);

                TasksList.add(currentTask);
                taskNames.add(currentTask.getName());
                currentTask.setTaskNames(taskNames);
                TaskDictionList.add(currentTask.getTaskDictionary());
            }
            Collections.sort(TaskDictionList, categoryComparator);
            /*
            System.out.println("\n\n");

            for (var a : TaskDictionList) {
                System.out.println(a);
            }*/
        }catch (SQLException e) {
            new JOptionPane().showMessageDialog(comp, e.getMessage());
        }
    }


    public void selctWhere(String TaskName, Component comp){
        String sql = "select * from TaskInfo where TaskName=?";

        try (Connection conn = this.connect(comp); PreparedStatement pstmt
= conn.prepareStatement(sql)){

            pstmt.setString(1, TaskName);
            ResultSet rs = pstmt.executeQuery();
            while (rs.next()) {
                Task currentTask = new Task();

                currentTask.setName(rs.getString("TaskName"));
                currentTask.setCategory(rs.getString("Category"));
                currentTask.setDescription(rs.getString("Description"));
                boolean compState =
Boolean.parseBoolean(rs.getString("CompletionState"));

                currentTask.setCompletionState(compState);

                TasksList.add(currentTask);
                taskNames.add(currentTask.getName());
                currentTask.setTaskNames(taskNames);
                TaskDictionList.add(currentTask.getTaskDictionary());
            }

        } catch (SQLException e) {
            new JOptionPane().showMessageDialog(comp, e.getMessage());
        }
    }

    public void update(String Name, String Category, String Description,
```

```
String CompletionState, Component comp){
        String sql = "UPDATE TaskInfo SET Category = ?, Description = ?,
CompletionState = ? "
                + "WHERE TaskName = ?";

        try (Connection conn = this.connect(comp); PreparedStatement pstmt
= conn.prepareStatement(sql)) {

            // set the corresponding param
            pstmt.setString(1, Category);
            pstmt.setString(2, Description);
            pstmt.setString(3, CompletionState);
            pstmt.setString(4, Name);
            // update
            pstmt.executeUpdate();
            new JOptionPane().showMessageDialog(comp, "Updated
succesfully");
        } catch (SQLException e) {
            new JOptionPane().showMessageDialog(comp, e.getMessage());
        }
    }

}
```

*TextViewForm.java Code:*

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-
default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to
edit this template
 */
package org.mondemkhize.jd522fa2;

import java.util.List;

/**
 *
 * @author Monde
 */
public class TextViewForm extends javax.swing.JFrame {

    /**
     * Creates new form TextViewForm
     */
    public TextViewForm() {
        initComponents();
        this.setLocationRelativeTo(null);
        Task task = new Task();
        List<String> fileTask = task.readFromFile(this);
        this.taskNameLbl.setText(fileTask.get(0));
        this.categoryLbl.setText(fileTask.get(1));
        this.descriLbl.setText(fileTask.get(2));
        this.compStateLbl.setText(fileTask.get(3));
        long fileSize = task.fileSize(this);
        fileSize = (fileSize<1024) ? fileSize:fileSize/1024;
        String form = (fileSize<1024) ? "%,d bytes": "%,d kilobytes";
        this.FileSizeLbl.setText(String.format(form, fileSize));
        this.savedTimeLbl.setText(task.creationTime(this));
        this.lastModifiedLbl.setText(task.lastModified(this));
```

```
    }

    /**
     * This method is called from within the constructor to initialize the
form.
     * WARNING: Do NOT modify this code. The content of this method is
always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-
BEGIN:initComponents
    private void initComponents() {

        jLabel4 = new javax.swing.JLabel();
        jLabel5 = new javax.swing.JLabel();
        taskNameLbl = new javax.swing.JLabel();
        categoryLbl = new javax.swing.JLabel();
        descriLbl = new javax.swing.JLabel();
        compStateLbl = new javax.swing.JLabel();
        CloseBtn = new javax.swing.JButton();
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();
        jLabel6 = new javax.swing.JLabel();
        FileSizeLbl = new javax.swing.JLabel();
        jLabel7 = new javax.swing.JLabel();
        savedTimeLbl = new javax.swing.JLabel();
        jLabel8 = new javax.swing.JLabel();
        lastModifiedLbl = new javax.swing.JLabel();
        jLabel9 = new javax.swing.JLabel();

        jLabel4.setText("jLabel4");

        jLabel5.setText("jLabel5");


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        taskNameLbl.setText("jLabel1");

        categoryLbl.setText("jLabel2");

        descriLbl.setText("jLabel3");

        compStateLbl.setText("jLabel4");

        CloseBtn.setText("Close");
        CloseBtn.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                CloseBtnActionPerformed(evt);
            }
        });

        jLabel1.setText("Task Name");

        jLabel2.setText("Category");

        jLabel3.setText("Completion Status");

        jLabel6.setText("Description");
```

```
        FileSizeLbl.setText("jLabel7");

        jLabel7.setText("File size");

        savedTimeLbl.setText("jLabel8");

        jLabel8.setText("Saved");

        lastModifiedLbl.setText("jLabel9");

        jLabel9.setText("Last Modified");

        javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
                .addGap(26, 26, 26)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAI
LING)
                    .addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAI
LING)
                            .addComponent(jLabel8)
                            .addComponent(jLabel7))
                        .addGap(26, 26, 26)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
                            .addComponent(FileSizeLbl,
javax.swing.GroupLayout.Alignment.TRAILING)
                            .addComponent(savedTimeLbl,
javax.swing.GroupLayout.Alignment.TRAILING)))
                    .addGroup(layout.createSequentialGroup()
                        .addComponent(jLabel9)
                        .addGap(18, 18, 18)
                        .addComponent(lastModifiedLbl)))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED,
javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE)
                .addComponent(CloseBtn)
                .addGap(44, 44, 44))
            .addGroup(layout.createSequentialGroup()
                .addGap(11, 11, 11)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
                    .addGroup(layout.createSequentialGroup()
                        .addComponent(descriLbl)

.addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE, Short.MAX_VALUE))
                    .addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
```

```
                            .addComponent(jLabel6)
                            .addComponent(jLabel1)
                            .addGroup(layout.createSequentialGroup()
                                .addComponent(taskNameLbl)
                                .addGap(72, 72, 72)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
                                    .addComponent(jLabel2)
                                    .addComponent(categoryLbl))))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 137,
Short.MAX_VALUE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
                            .addComponent(jLabel3)
                            .addComponent(compStateLbl))
                        .addGap(68, 68, 68))))
        );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addGap(25, 25, 25)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASE
LINE)
                    .addComponent(jLabel1)
                    .addComponent(jLabel2)
                    .addComponent(jLabel3))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASE
LINE)
                    .addComponent(taskNameLbl)
                    .addComponent(categoryLbl)
                    .addComponent(compStateLbl))
                .addGap(26, 26, 26)
                .addComponent(jLabel6)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
                .addComponent(descriLbl)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 133,
Short.MAX_VALUE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASE
LINE)
                    .addComponent(FileSizeLbl)
                    .addComponent(jLabel7))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEAD
ING)
                    .addGroup(layout.createSequentialGroup()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASE
LINE)
```

```
                            .addComponent(savedTimeLbl)
                            .addComponent(jLabel8))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASE
LINE)
                            .addComponent(lastModifiedLbl)
                            .addComponent(jLabel9))
                        .addContainerGap())
                    .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
                        .addComponent(CloseBtn)
                        .addGap(33, 33, 33))))
        );

        pack();
    }// </editor-fold>//GEN-END:initComponents

    private void CloseBtnActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_CloseBtnActionPerformed
        // TODO add your handling code here:
        JD522FA2 main = new JD522FA2();
        main.setVisible(true);
        this.dispose();
    }//GEN-LAST:event_CloseBtnActionPerformed

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel
setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with
the default look and feel.
         * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(TextViewForm.class.getName()).log(java.u
til.logging.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(TextViewForm.class.getName()).log(java.u
til.logging.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(TextViewForm.class.getName()).log(java.u
til.logging.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {
```

```
java.util.logging.Logger.getLogger(TextViewForm.class.getName()).log(java.u
til.logging.Level.SEVERE, null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new TextViewForm().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton CloseBtn;
    private javax.swing.JLabel FileSizeLbl;
    private javax.swing.JLabel categoryLbl;
    private javax.swing.JLabel compStateLbl;
    private javax.swing.JLabel descriLbl;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JLabel jLabel5;
    private javax.swing.JLabel jLabel6;
    private javax.swing.JLabel jLabel7;
    private javax.swing.JLabel jLabel8;
    private javax.swing.JLabel jLabel9;
    private javax.swing.JLabel lastModifiedLbl;
    private javax.swing.JLabel savedTimeLbl;
    private javax.swing.JLabel taskNameLbl;
    // End of variables declaration//GEN-END:variables
}
```

## *ViewForm.java Code:*

```
/*
 * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-
default.txt to change this license
 * Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to
edit this template
 */
package org.mondemkhize.jd522fa2;

import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import javax.swing.JDialog;
import javax.swing.JOptionPane;
import javax.swing.event.DocumentEvent;
import javax.swing.event.DocumentListener;
import me.xdrop.fuzzywuzzy.FuzzySearch;
import me.xdrop.fuzzywuzzy.ToStringFunction;
import me.xdrop.fuzzywuzzy.model.BoundExtractedResult;
import me.xdrop.fuzzywuzzy.model.ExtractedResult;

/**
 *
 * @author monde
 */
```

```
public class ViewForm extends javax.swing.JFrame {

    /**
     * Creates new form ViewForm
     */
    TaskDB tdb = new TaskDB();
    ArrayList<HashMap<String, String>> taskDictionList =
tdb.getTaskDictionList();
    ArrayList<String> taskNames = tdb.getTasksNames();
    public ViewForm() {
        initComponents();
        this.setLocationRelativeTo(null);
        this.searchField.getDocument().addDocumentListener(new
DocumentListener (){
            @Override//from  w  w  w .  j  av  a 2s.  com
            public void insertUpdate(DocumentEvent e) {
                List<ExtractedResult> match =
FuzzySearch.extractTop(searchField.getText(), taskNames, 5);
                tdb.clearLists();
                for(ExtractedResult a: match){
                    System.out.println(a.getString());
                    tdb.selctWhere(a.getString(), rootPane);
                }
                jTable1.setModel(new
javax.swing.table.DefaultTableModel(new String[]{"Task Name", "Category",
"Description", "CompletionState"}, taskDictionList.size()));
                for (int r = 0; r < taskDictionList.size(); r++) {

jTable1.setValueAt(taskDictionList.get(r).get("TaskName"), r, 0);

jTable1.setValueAt(taskDictionList.get(r).get("Category"), r, 1);

jTable1.setValueAt(taskDictionList.get(r).get("Description"), r, 2);

jTable1.setValueAt(taskDictionList.get(r).get("CompletionState"), r, 3);
                }
            }

            @Override
            public void removeUpdate(DocumentEvent e) {
                if(!searchField.getText().isEmpty()){
                List<ExtractedResult> match =
FuzzySearch.extractTop(searchField.getText(), taskNames, 5);
                tdb.clearLists();
                for (ExtractedResult a : match) {
                    System.out.println(a.getString());
                    tdb.selctWhere(a.getString(), rootPane);
                }
                jTable1.setModel(new
javax.swing.table.DefaultTableModel(new String[]{"Task Name", "Category",
"Description", "CompletionState"}, taskDictionList.size()));
                for (int r = 0; r < taskDictionList.size(); r++) {

jTable1.setValueAt(taskDictionList.get(r).get("TaskName"), r, 0);

jTable1.setValueAt(taskDictionList.get(r).get("Category"), r, 1);

jTable1.setValueAt(taskDictionList.get(r).get("Description"), r, 2);

jTable1.setValueAt(taskDictionList.get(r).get("CompletionState"), r, 3);
                }
```

```
                }else{
                    tdb.selectAll(rootPane);
                    ArrayList<HashMap<String, String>> taskDictionList =
tdb.getTaskDictionList();
                    jTable1.setModel(new
javax.swing.table.DefaultTableModel(new String[]{"Task Name", "Category",
"Description", "CompletionState"}, taskDictionList.size()));
                    for (int r = 0; r < taskDictionList.size(); r++) {

jTable1.setValueAt(taskDictionList.get(r).get("TaskName"), r, 0);

jTable1.setValueAt(taskDictionList.get(r).get("Category"), r, 1);

jTable1.setValueAt(taskDictionList.get(r).get("Description"), r, 2);

jTable1.setValueAt(taskDictionList.get(r).get("CompletionState"), r, 3);
                    }
                }
            }

            @Override
            public void changedUpdate(DocumentEvent e) {
                System.out.println(searchField.getText());
            }
        });
        /*
        jTable1.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
        {null, null, null, null},
        {null, null, null, null},
        {null, null, null, null}
    },
    new String [] {
        "Task Name", "Category", "Description", "CompletionState"
    }
) {
    Class[] types = new Class [] {
        java.lang.String.class, java.lang.String.class,
java.lang.String.class, java.lang.String.class
    };

    public Class getColumnClass(int columnIndex) {
        return types [columnIndex];
    }
});
        */
    }

    /**
     * This method is called from within the constructor to initialize the
form.
     * WARNING: Do NOT modify this code. The content of this method is
always
     * regenerated by the Form Editor.
     */
    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-
BEGIN:initComponents
    private void initComponents() {

        jScrollPane2 = new javax.swing.JScrollPane();
```

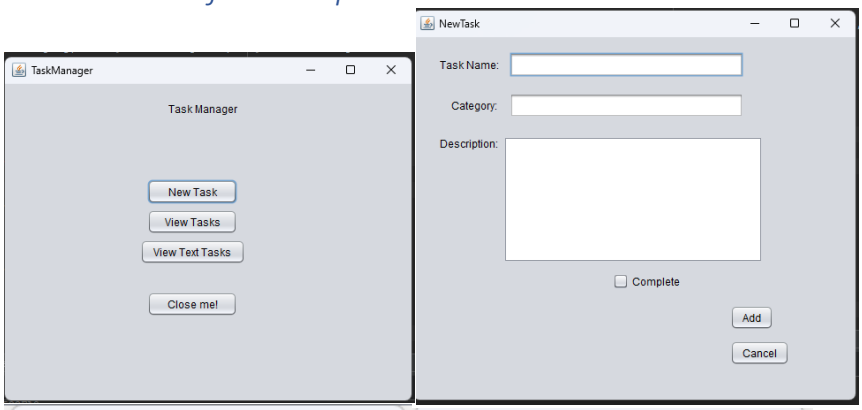
```
        tdb.selectAll(this);
        ArrayList<HashMap<String,String>> taskDictionList =
tdb.getTaskDictionList();
        jTable1 = new javax.swing.JTable();
        BackBtn = new javax.swing.JButton();
        editTaskBtn = new javax.swing.JButton();
        exportCSVBtn = new javax.swing.JButton();
        searchLbl = new javax.swing.JLabel();
        searchField = new javax.swing.JTextField();
        jLabel1 = new javax.swing.JLabel();


setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        jTable1.setModel(new javax.swing.table.DefaultTableModel(new String
[] {"Task Name", "Category", "Description",
"CompletionState"},taskDictionList.size()));
        for(int r=0;r<taskDictionList.size();r++){
            jTable1.setValueAt(taskDictionList.get(r).get("TaskName"), r,
0);
            jTable1.setValueAt(taskDictionList.get(r).get("Category"), r,
1);
            jTable1.setValueAt(taskDictionList.get(r).get("Description"),
r, 2);

jTable1.setValueAt(taskDictionList.get(r).get("CompletionState"), r, 3);
        }
        jTable1.setRowSelectionAllowed(false);
        jScrollPane2.setViewportView(jTable1);

        BackBtn.setText("Return Home");
        BackBtn.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                BackBtnActionPerformed(evt);
            }
        });

        editTaskBtn.setText("Edit Task");
        editTaskBtn.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                editTaskBtnActionPerformed(evt);
            }
        });

        exportCSVBtn.setText("Export to CSV");
        exportCSVBtn.addActionListener(new java.awt.event.ActionListener()
{
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                exportCSVBtnActionPerformed(evt);
            }
        });

        searchLbl.setText("Search:");

        jLabel1.setText("Tasks Saved");

        javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
        getContentPane().setLayout(layout);
        layout.setHorizontalGroup(
```

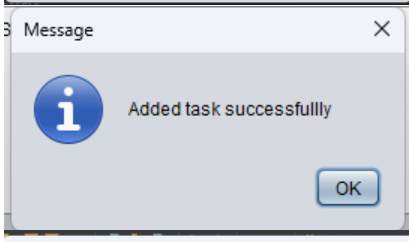
```
layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(layout.createSequentialGroup()
                .addContainerGap(82, Short.MAX_VALUE)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAI
LING)
                    .addGroup(layout.createSequentialGroup()
                        .addComponent(editTaskBtn)
                        .addGap(18, 18, 18)
                        .addComponent(exportCSVBtn)
                        .addGap(18, 18, 18)
                        .addComponent(BackBtn)
                        .addGap(52, 52, 52))
                    .addGroup(layout.createSequentialGroup()
                        .addComponent(jScrollPane2,
javax.swing.GroupLayout.PREFERRED_SIZE, 488,
javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addGap(134, 134, 134))
                    .addGroup(layout.createSequentialGroup()
                        .addComponent(jLabel1)
                        .addGap(179, 179, 179)
                        .addComponent(searchLbl)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
                        .addComponent(searchField,
javax.swing.GroupLayout.PREFERRED_SIZE, 117,
javax.swing.GroupLayout.PREFERRED_SIZE)
                        .addGap(22, 22, 22))))
        );
        layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
            .addGroup(javax.swing.GroupLayout.Alignment.TRAILING,
layout.createSequentialGroup()
                .addGap(19, 19, 19)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASE
LINE)
                    .addComponent(searchLbl)
                    .addComponent(searchField,
javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(jLabel1))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, 28,
Short.MAX_VALUE)
                .addComponent(jScrollPane2,
javax.swing.GroupLayout.PREFERRED_SIZE, 320,
javax.swing.GroupLayout.PREFERRED_SIZE)
                .addGap(18, 18, 18)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASE
LINE)
                    .addComponent(BackBtn)
                    .addComponent(editTaskBtn)
                    .addComponent(exportCSVBtn))
                .addGap(24, 24, 24))
        );

        pack();
```

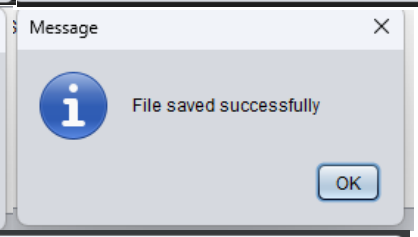
```
    }// </editor-fold>//GEN-END:initComponents

    private void BackBtnActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_BackBtnActionPerformed
        JD522FA2 mainwin = new JD522FA2();
        mainwin.setVisible(true);
        this.dispose();
    }//GEN-LAST:event_BackBtnActionPerformed

    private void editTaskBtnActionPerformed(java.awt.event.ActionEvent evt)
{//GEN-FIRST:event_editTaskBtnActionPerformed

        try{

//System.out.println(this.jTable1.getValueAt(this.jTable1.getSelectedRow(),
this.jTable1.getSelectedColumn()));

//System.out.println(this.jTable1.editCellAt(this.jTable1.getEditingRow(),
this.jTable1.getEditingColumn()));
            String taskName =
this.jTable1.getValueAt(this.jTable1.getSelectedRow(), 0).toString();
            String category =
this.jTable1.getValueAt(this.jTable1.getSelectedRow(), 1).toString();
            String description =
this.jTable1.getValueAt(this.jTable1.getSelectedRow(), 2).toString();
            String completionState =
this.jTable1.getValueAt(this.jTable1.getSelectedRow(), 3).toString();
            NewTaskForm editForm = new NewTaskForm(this, taskName,
category, description,Boolean.parseBoolean(completionState));
            editForm.setVisible(true);
            this.dispose();
        }catch(Exception a){
            new JOptionPane().showMessageDialog(this, a.getMessage());
        }

    }//GEN-LAST:event_editTaskBtnActionPerformed

    private void exportCSVBtnActionPerformed(java.awt.event.ActionEvent
evt) {//GEN-FIRST:event_exportCSVBtnActionPerformed
        // TODO add your handling code here:
        try{
        String pathName =
this.jTable1.getValueAt(this.jTable1.getSelectedRow(),
0).toString().replaceAll(" ", "-")+".csv";
        Task task = new Task();
        task.xportToCSV(pathName, this.jTable1, this);
        }catch(Exception a){
            new JOptionPane().showMessageDialog(this, a.getMessage());
        }
    }//GEN-LAST:event_exportCSVBtnActionPerformed

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel
setting code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with
the default look and feel.
         * For details see
```

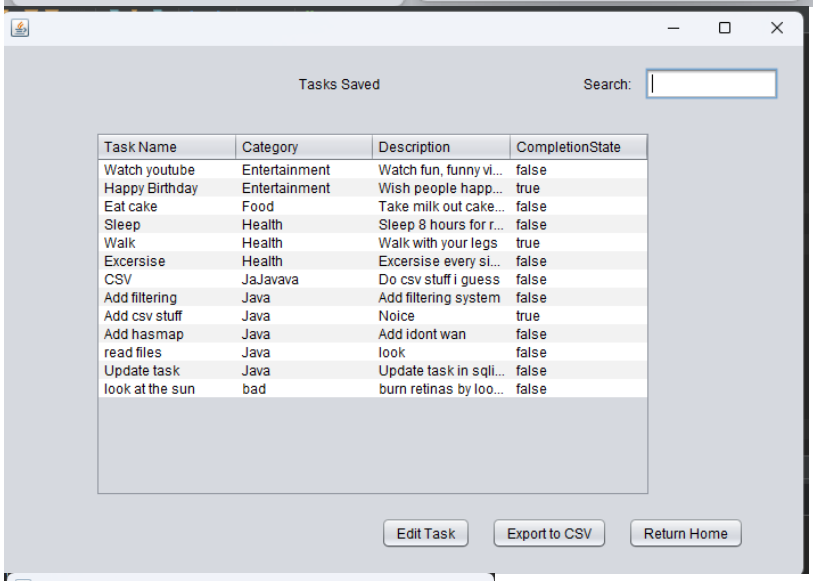
```
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(ViewForm.class.getName()).log(java.util.
logging.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(ViewForm.class.getName()).log(java.util.
logging.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(ViewForm.class.getName()).log(java.util.
logging.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(ViewForm.class.getName()).log(java.util.
logging.Level.SEVERE, null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new ViewForm().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton BackBtn;
    private javax.swing.JButton editTaskBtn;
    private javax.swing.JButton exportCSVBtn;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JScrollPane jScrollPane2;
    private javax.swing.JTable jTable1;
    private javax.swing.JTextField searchField;
    private javax.swing.JLabel searchLbl;
    // End of variables declaration//GEN-END:variables
}
```

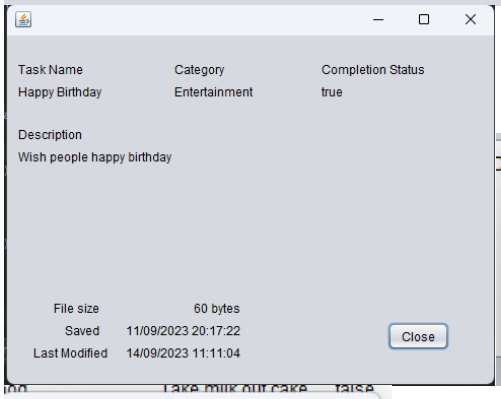
*Screenshots of the output:*

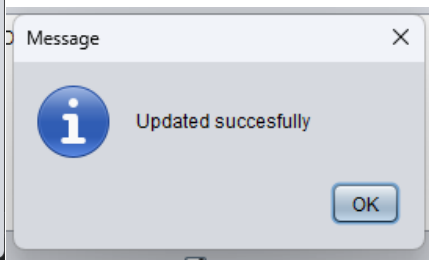


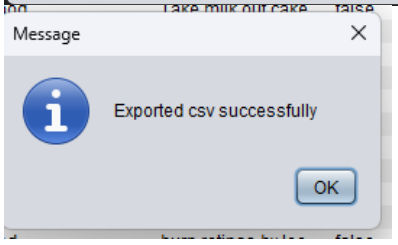



## Completed Declaration of Authenticity

I, Monde Mkhize, solemnly declare that the work presented in this formative is entirely my own. I have not plagiarized or copied the work of others without proper acknowledgment. I affirm that the content, ideas, and arguments presented herein are the result of my independent effort an intellectual contribution. I understand the significance of academic integrity and the detrimental consequences of engaging in plagiarism or other forms of dishonesty. Therefore, I assure you the following:

1. All sources used in this work, including but not limited to books, articles, websites, and personal communications, have been appropriately cited and referenced according to the specified guidelines or referencing style.
2. Any direct quotations or paraphrased information from external sources have been identified by using quotation marks or proper citation methods.
3. I have not received any unauthorized assistance or collaboration from others in completing this work, except for instances explicitly permitted by the instructor or clearly stated in the assignment guidelines.
4. The ideas, arguments, and interpretations expressed in this work are my own and have not been submitted for assessment in any other academic setting unless explicitly mentioned and properly acknowledged.
5. I acknowledge that failure to adhere to these principles of academic honesty and integrity may result in severe penalties, including but not limited to the rejection of this work, loss of marks, academic probation, or disciplinary action as deemed appropriate by the educational institution.

Signature:  Date: 14 September 2023