Practical:-Create basic form and display the data on te console

package com.typractical;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class UserInputForm extends JFrame {

public JTextField nameField;

public JPasswordField passwordField;

public JCheckBox rememberMe;

public ButtonGroup genderGroup;

public JRadioButton male, female, other;

public JButton submitButton;

public UserInputForm() {

setTitle("User Input Form");

setSize(400, 200);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setLayout(new GridLayout(5, 2)); // Use GridLayout for better component arrangement

// Create components

JLabel nameLabel = new JLabel("Name:");

nameField = new JTextField();

JLabel passwordLabel = new JLabel("Password:");

passwordField = new JPasswordField();

rememberMe = new JCheckBox("Remember Me");

JLabel genderLabel = new JLabel("Gender:");

male = new JRadioButton("Male");

female = new JRadioButton("Female");

other = new JRadioButton("Other");

genderGroup = new ButtonGroup();

genderGroup.add(male);

genderGroup.add(female);

genderGroup.add(other);

submitButton = new JButton("Submit");

// Add components to the frame

add(nameLabel);

add(nameField);

add(passwordLabel);

add(passwordField);

add(rememberMe);

add(new JPanel()); // Empty panel for spacing

add(genderLabel);

add(new JPanel() {{

add(male);

add(female);

add(other);

}}); // Add radio buttons to a panel

add(new JPanel()); // Empty panel for spacing

add(submitButton);

// Add action listener to the submit button

submitButton.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

String name = nameField.getText();

String password = new String(passwordField.getPassword());

boolean isRemembered = rememberMe.isSelected();

String gender = "";

if (male.isSelected()) {

gender = "Male";

} else if (female.isSelected()) {

gender = "Female";

} else if (other.isSelected()) {

gender = "Other";

}

// Display entered details on console

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

System.out.println("Password: " + password);

System.out.println("Remember Me: " + isRemembered);

System.out.println("Gender: " + gender);

}

});

}

public static void main(String[] args) {

SwingUtilities.invokeLater(() -> {

new UserInputForm().setVisible(true);

});

}

}

Practical:- Write a java program to present a set of choices for a user to select Stationary products and display the price of Product after Selection from the list.

**package** com.typractical;

**import** javax.swing.\*;

**import** java.awt.event.\*;

**public** **class** prac1 {

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

// Create a JFrame

JFrame frame = **new** JFrame("Stationary Selector");

frame.setSize(400, 200);

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

frame.setLayout(**null**);

// Create a label for instructions

JLabel instructionLabel = **new** JLabel("Select a stationary product:");

instructionLabel.setBounds(20, 20, 200, 25);

frame.add(instructionLabel);

// Create a JComboBox with stationary product choices

String[] products = {"Notebook", "Pen", "Pencil", "Eraser", "Sharpener"};

JComboBox<String> productComboBox = **new** JComboBox<>(products);

productComboBox.setBounds(20, 50, 150, 25);

frame.add(productComboBox);

// Create a label to display the price

JLabel priceLabel = **new** JLabel("Price: ");

priceLabel.setBounds(20, 100, 200, 25);

frame.add(priceLabel);

// Add an ActionListener to the JComboBox

productComboBox.addActionListener(**new** ActionListener() {

@Override

**public** **void** actionPerformed(ActionEvent e) {

// Get the selected item

String selectedProduct = (String) productComboBox.getSelectedItem();

// Determine the price based on the selection

String price;

**switch** (selectedProduct) {

**case** "Notebook":

price = "$5.00";

**break**;

**case** "Pen":

price = "$1.50";

**break**;

**case** "Pencil":

price = "$0.75";

**break**;

**case** "Eraser":

price = "$0.50";

**break**;

**case** "Sharpener":

price = "$1.00";

**break**;

**default**:

price = "Unknown";

**break**;

}

// Display the price

priceLabel.setText("Price: " + price);

}

});

// Make the frame visible

frame.setVisible(**true**);

}

}

Write a java program to demonstrate typical Editable Table, describing employee details such as name roll-number and age.

**package** com.typractical;

**import** javax.swing.\*;

**import** javax.swing.table.DefaultTableModel;

**import** java.awt.\*;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.ActionListener;

**import** java.awt.event.MouseAdapter;

**import** java.awt.event.MouseEvent;

**public** **class** table3 **implements** ActionListener {

**public** JFrame frame;

**public** JTable table;

**public** JTextField nameField, rollnoField, ageField;

**public** JButton addButton, deleteButton;

**public** DefaultTableModel model;

**public** table3() {

// Create the frame

frame = **new** JFrame("Student Table");

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

frame.setLayout(**new** BorderLayout());

// Create the table model

model = **new** DefaultTableModel();

model.addColumn("Name");

model.addColumn("Roll No");

model.addColumn("Age");

// Create the table

table = **new** JTable(model);

table.setSelectionMode(ListSelectionModel.***SINGLE\_SELECTION***); // Allow single row selection

JScrollPane scrollPane = **new** JScrollPane(table);

frame.add(scrollPane, BorderLayout.***CENTER***);

// Create the input fields and buttons

JPanel inputPanel = **new** JPanel(**new** GridLayout(4, 2));

inputPanel.add(**new** JLabel("Name:"));

nameField = **new** JTextField();

inputPanel.add(nameField);

inputPanel.add(**new** JLabel("Roll No:"));

rollnoField = **new** JTextField();

inputPanel.add(rollnoField);

inputPanel.add(**new** JLabel("Age:"));

ageField = **new** JTextField();

inputPanel.add(ageField);

addButton = **new** JButton("Add Row");

addButton.addActionListener(**this**);

inputPanel.add(addButton);

deleteButton = **new** JButton("Delete Row");

deleteButton.addActionListener(**this**);

deleteButton.setEnabled(**false**); // Initially disable delete button

inputPanel.add(deleteButton);

frame.add(inputPanel, BorderLayout.***SOUTH***);

// Add mouse listener to table for row selection

table.addMouseListener(**new** MouseAdapter() {

@Override

**public** **void** mouseClicked(MouseEvent e) {

// Enable delete button only if a row is selected

**if** (table.getSelectedRow() >= 0) {

deleteButton.setEnabled(**true**);

} **else** {

deleteButton.setEnabled(**false**);

}

}

});

// Set frame size and visibility

frame.setSize(400, 300);

frame.setVisible(**true**);

}

@Override

**public** **void** actionPerformed(ActionEvent e) {

**if** (e.getSource() == addButton) {

String name = nameField.getText();

String rollno = rollnoField.getText();

String age = ageField.getText();

// Add the row to the table model

model.addRow(**new** Object[]{name, rollno, age});

// Clear the input fields

nameField.setText("");

rollnoField.setText("");

ageField.setText("");

}

**else** **if** (e.getSource() == deleteButton) {

// Get the selected row index

**int** selectedRow = table.getSelectedRow();

**if** (selectedRow >= 0) {

// Remove the selected row from the table model

model.removeRow(selectedRow);

// Clear selection and disable delete button

table.getSelectionModel().clearSelection();

deleteButton.setEnabled(**false**);

}

}

}

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

**new** table3();

}

}

Show some Implementation of advance component of Swing

//code already posted to github take from there.

Write a java program using Split pane to demonstrate a screen divided in two parts, one part contains the names of Planets and another Displays the image of planet. When user selects the planet name form Left screen, appropriate image of planet displayed in right screen.

**package** com.typractical;

**import** javax.swing.\*;

**import** javax.swing.event.ListSelectionEvent;

**import** javax.swing.event.ListSelectionListener;

**import** java.awt.\*;

**public** **class** prac3 {

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

// Create a JFrame

JFrame frame = **new** JFrame("Planet Viewer");

frame.setSize(600, 400);

frame.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

// Create a list of planet names

String[] planets = {"Mercury", "Venus", "Earth", "Mars", "Jupiter", "Saturn", "Uranus", "Neptune"};

JList<String> planetList = **new** JList<>(planets);

// Create a label to display the planet image

JLabel imageLabel = **new** JLabel("Select a planet to view its image.", JLabel.***CENTER***);

imageLabel.setFont(**new** Font("Arial", Font.***PLAIN***, 16));

// Create a SplitPane

JSplitPane splitPane = **new** JSplitPane(JSplitPane.***HORIZONTAL\_SPLIT***);

// Set the left pane with the planet list

splitPane.setLeftComponent(**new** JScrollPane(planetList));

// Set the right pane with the image label

splitPane.setRightComponent(imageLabel);

// Set the divider location

splitPane.setDividerLocation(200);

// Add an anonymous class to handle ListSelectionListener

planetList.addListSelectionListener(**new** ListSelectionListener() {

@Override

**public** **void** valueChanged(ListSelectionEvent e) {

**if** (!e.getValueIsAdjusting()) {

// Get the selected planet

String selectedPlanet = planetList.getSelectedValue();

// Determine the image path dynamically using traditional switch

String imagePath = **null**;

**switch** (selectedPlanet) {

// Note:-create a paint image by your own and save the image in the same location where ur code is present and give the location and

**case** "Mercury":

imagePath = "C:\\Users\\Lenovo\\eclipse-workspace\\Typrac\\src\\com\\typractical\\mercury.png";

**break**;

**case** "Venus":

imagePath = "images/venus.jpg";

**break**;

**case** "Earth":

imagePath = "images/earth.jpg";

**break**;

**case** "Mars":

imagePath = "images/mars.jpg";

**break**;

**case** "Jupiter":

imagePath = "images/jupiter.jpg";

**break**;

**case** "Saturn":

imagePath = "images/saturn.jpg";

**break**;

**case** "Uranus":

imagePath = "images/uranus.jpg";

**break**;

**case** "Neptune":

imagePath = "images/neptune.jpg";

**break**;

**default**:

**break**;

}

**if** (imagePath != **null**) {

imageLabel.setText(""); // Clear the text

imageLabel.setIcon(**new** ImageIcon(imagePath));

} **else** {

imageLabel.setText("Image not available.");

imageLabel.setIcon(**null**);

}

}

}

});

// Add the split pane to the frame

frame.add(splitPane);

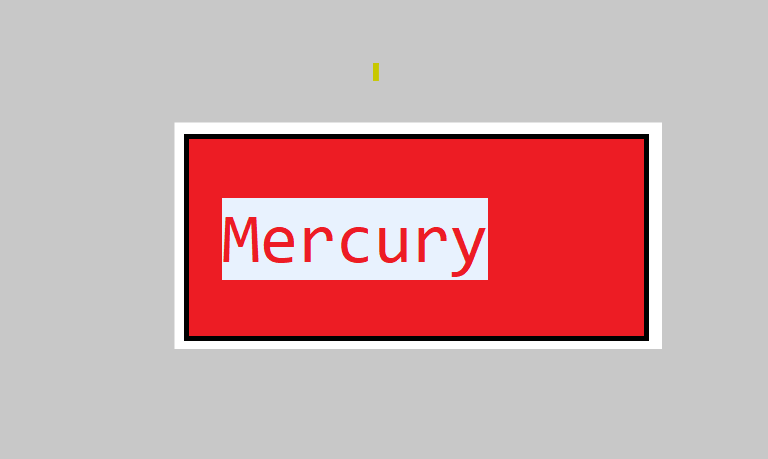
// Make the frame visible

frame.setVisible(**true**);

}

}

Ref:--

Image examle

After clicking on the mercury :-

Output:

