

📄

🔍

🔗

🔧

📁

🔍

🔧

👤

⚙️

RUN AND DEBUG

⋮

Run and Debug

To customize Run and Debug create a launch.json file.

Show all automatic debug configurations.

BREAKPOINTS

Welcome

J javafx

J ExpenseTrackerApp.java X

J expence.java 9+

HTML

ExpenseTrackerApp.java > ExpenseTrackerApp > ExpenseTrackerApp()

```
1 import javax.swing.*;
2 import java.awt.*;
3 import java.awt.event.ActionEvent;
4 import java.awt.event.ActionListener;
5 import java.text.DecimalFormat;
6 import java.util.ArrayList;
7 import java.util.List;
8
9 public class ExpenseTrackerApp extends JFrame {
10     private List<Expense> expenses = new ArrayList<>();
11     private DefaultListModel<String> listModel;
12     private JList<String> expenseList;
13     private JLabel totalLabel;
14
15     public ExpenseTrackerApp() {
16         setTitle(title:"Expense Tracker");
17         setSize(width:550, height:550);
18         setDefaultCloseOperation(EXIT_ON_CLOSE);
19         setLayout(new BorderLayout());
20
21         listModel = new DefaultListModel<>();
22         expenseList = new JList<>(listModel);
23         JScrollPane scrollPane = new JScrollPane(expenseList);
24
25         JPanel topPanel = new JPanel();
26         topPanel.setLayout(new BorderLayout());
27
28         totalLabel = new JLabel(text:"Total Expenses: ₹0.00");
29         topPanel.add(totalLabel, BorderLayout.NORTH);
30
31         add(scrollPane, BorderLayout.CENTER);
32         add(topPanel, BorderLayout.NORTH);
33
34         JButton addExpenseButton = new JButton(text:"Add Expense");
35         addExpenseButton.addActionListener(new ActionListener() {
36             @Override
37             public void actionPerformed(ActionEvent e) {
38                 showExpenseDialog();
39             }
40         });
41
42         JButton viewSummariesButton = new JButton(text:"View Summaries");
43         viewSummariesButton.addActionListener(new ActionListener() {
44             @Override
45             public void actionPerformed(ActionEvent e) {
46                 showSummaries();
47             }
48         });
49
50         JPanel buttonPanel = new JPanel();
51         buttonPanel.setLayout(new FlowLayout());
52         buttonPanel.add(addExpenseButton);
```

Ln 32, Col 43

Spaces: 4

UTF-8

LF

{ } Java

📄

🔍

🔗

🔧

📦

🖥️

🧪

👤

⚙️

RUN AND DEBUG

Run and Debug

To customize Run and Debug create a launch.json file.

Show all automatic debug configurations.

BREAKPOINTS

Welcome

java

ExpenseTrackerApp.java

expence.java 9+

ExpenseTrackerApp.java

ExpenseTrackerApp

ExpenseTrackerApp()

```
49
50 JPanel buttonPanel = new JPanel();
51 buttonPanel.setLayout(new FlowLayout());
52 buttonPanel.add(addExpenseButton);
53 buttonPanel.add(viewSummariesButton);
54 add(buttonPanel, BorderLayout.SOUTH);
55 }
56
57 private void showExpenseDialog() {
58     JPanel expensePanel = new JPanel(new GridLayout(rows:3, cols:2));
59     JTextField descriptionField = new JTextField(columns:20);
60     JTextField amountField = new JTextField(columns:10);
61     JTextField categoryField = new JTextField(columns:20);
62
63     expensePanel.add(new JLabel(text:"Description:"));
64     expensePanel.add(descriptionField);
65     expensePanel.add(new JLabel(text:"Amount: ₹"));
66     expensePanel.add(amountField);
67     expensePanel.add(new JLabel(text:"Category:"));
68     expensePanel.add(categoryField);
69
70     int result = JOptionPane.showConfirmDialog(parentComponent:null, expensePanel, title:"Enter Expense Details", JOptionPane.OK_CANCEL_OPTION);
71     if (result == JOptionPane.OK_OPTION) {
72         String description = descriptionField.getText();
73         double amount = Double.parseDouble(amountField.getText());
74         String category = categoryField.getText();
75
76         Expense expense = new Expense(description, amount, category);
77         expenses.add(expense);
78         listModel.addElement(expense.toString());
79
80         updateTotalLabel();
81     }
82 }
83
84 private void showSummaries() {
85     JComboBox<String> categoryComboBox = new JComboBox<>();
86     categoryComboBox.addItem(item:"All Categories");
87     for (Expense expense : expenses) {
88         if (!categoryComboBox.getItemAt(categoryComboBox.getItemCount() - 1).equals(expense.getCategory())) {
89             categoryComboBox.addItem(expense.getCategory());
90         }
91     }
92
93     JPanel summaryPanel = new JPanel(new GridLayout(rows:2, cols:1));
94     summaryPanel.add(categoryComboBox);
95
96     int result = JOptionPane.showConfirmDialog(parentComponent:null, summaryPanel, title:"View Expense Summaries", JOptionPane.OK_CANCEL_OPTION);
97     if (result == JOptionPane.OK_OPTION) {
98         String selectedCategory = categoryComboBox.getSelectedItem().toString();
99         double totalExpense = calculateTotalExpense(selectedCategory);
100         displayExpenseSummary(selectedCategory, totalExpense);
```

Ln 32, Col 43

Spaces: 4

UTF-8

LF

{}

Java



☰

🔍

🔗

🔧

📁

🖨

🔬

👤

⚙️

RUN AND DEBUG

...

Run and Debug

To customize Run and Debug create a launch.json file.

Show all automatic debug configurations.

BREAKPOINTS

Welcome

J javafx

J ExpenseTrackerApp.java X

J expence.java 9+

ExpenseTrackerApp.java > ExpenseTrackerApp > ExpenseTrackerApp()

```
128         public void run() {
129             ExpenseTrackerApp expenseTracker = new ExpenseTrackerApp();
130             expenseTracker.setVisible(b:true);
131         }
132     };
133 }
134 }
135
136 class Expense {
137     private String description;
138     private double amount;
139     private String category;
140
141     public Expense(String description, double amount, String category) {
142         this.description = description;
143         this.amount = amount;
144         this.category = category;
145     }
146
147     public String getDescription() {
148         return description;
149     }
150
151     public double getAmount() {
152         return amount;
153     }
154
155     public String getCategory() {
156         return category;
157     }
158
159     @Override
160     public String toString() {
161         return description + " - ₹" + String.format(format:"%.2f", amount) + " (" + category + ")";
162     }
163 }
164
```

Ln 32, Col 43

Spaces: 4

UTF-8

LF

{ }

Java

🔍