

**Object-Oriented Programming in Java**

**Mini Project Report**

Speed Converter

By

Bhuvan M (1RVU23CSE115)

Gayan G (1RVU23CSE166)

Under the guidance of:

Dr./Prof. Lokesh J K

School of Computer Science and Engineering

RV University, Bangalore



**School of Computer Science and Engineering**

CERTIFICATE

Certified that the CS2024 Mini Project work titled ‘Speed Converter’ is carried out by Bhuvan M (1RVU23CSE115) and Gayan G (1RVU23CSE166) who is a bonafide student of the School of Computer Science and Engineering, RV University, Bengaluru, during the year 2024–25. It is certified that all corrections/ suggestions from all the continuous internal evaluations have been incorporated in the project and in this report.

Dr./ Prof. Lokesh J K Dr. Sudhakar K. N

Faculty Guide Program Director

Source Code

**MainClass.java**

import javax.swing.\*;  
import java.awt.\*;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
  
abstract class Converter {  
 public abstract double convert(double value);  
}  
  
class SpeedConverter extends Converter {  
 private double value;  
  
 @Override  
 public double convert(double value) {  
 this.value = value;  
 return value \* 0.621371;   
 }  
  
 public double convertToKmh(double value) {  
 this.value = value;  
 return value / 0.621371;   
 }  
  
 public double convertToMsFromKmh(double value) {  
 this.value = value;  
 return value / 3.6;   
 }  
  
 public double convertToMsFromMph(double value) {  
 this.value = value;  
 return value \* 0.44704;   
 }  
  
 public double convertToKmhFromMs(double value) {  
 this.value = value;  
 return value \* 3.6;   
 }  
  
 public double convertToMphFromMs(double value) {  
 this.value = value;  
 return value / 0.44704;   
 }  
}  
  
public class Main extends JFrame {  
 private JComboBox<String> unitBox;  
 private JTextField inputField, outputField;  
 private JButton convertButton;  
  
 public Main() {  
 setTitle("Speed Converter");  
 setSize(400, 300);  
 setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);  
 setLayout(new BorderLayout());

try {  
 for (UIManager.LookAndFeelInfo info : UIManager.*getInstalledLookAndFeels*()) {  
 if ("Nimbus".equals(info.getName())) {  
 UIManager.*setLookAndFeel*(info.getClassName());  
 break;  
 }  
 }  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
  
 JPanel inputPanel = new JPanel();  
 inputPanel.setLayout(new GridLayout(3, 2, 10, 10));  
  
 JLabel inputLabel = new JLabel("Enter Speed:");  
 inputField = new JTextField();  
 JLabel outputLabel = new JLabel("Converted Speed:");  
 outputField = new JTextField();  
 outputField.setEditable(false);  
  
 String[] units = {  
 "Km/h to Mph", "Mph to Km/h",  
 "Km/h to m/s", "Mph to m/s",  
 "m/s to Km/h", "m/s to Mph"  
 };  
 unitBox = new JComboBox<>(units);  
  
 inputPanel.add(inputLabel);  
 inputPanel.add(inputField);  
 inputPanel.add(new JLabel("Conversion Type:"));  
 inputPanel.add(unitBox);  
 inputPanel.add(outputLabel);  
 inputPanel.add(outputField);  
  
 convertButton = new JButton("Convert");  
 convertButton.addActionListener(new ConvertActionListener());  
  
 add(inputPanel, BorderLayout.*CENTER*);  
 add(convertButton, BorderLayout.*SOUTH*);  
  
 setVisible(true);  
 }

private class ConvertActionListener implements ActionListener {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 try {  
 double inputValue = Double.*parseDouble*(inputField.getText());  
 SpeedConverter converter = new SpeedConverter();  
 double result = 0;  
  
 switch (unitBox.getSelectedItem().toString()) {  
 case "Km/h to Mph":  
 result = converter.convert(inputValue);   
 break;  
 case "Mph to Km/h":  
 result = converter.convertToKmh(inputValue);   
 break;  
 case "Km/h to m/s":  
 result = converter.convertToMsFromKmh(inputValue);   
 break;  
 case "Mph to m/s":  
 result = converter.convertToMsFromMph(inputValue);   
 break;  
 case "m/s to Km/h":  
 result = converter.convertToKmhFromMs(inputValue);   
 break;  
 case "m/s to Mph":  
 result = converter.convertToMphFromMs(inputValue);   
 break;  
 }  
  
 outputField.setText(String.*format*("%.2f", result));  
 } catch (NumberFormatException ex) {  
 JOptionPane.*showMessageDialog*(Main.this, "Invalid input. Please enter a numeric value.");  
 }  
 }  
 }  
  
 public static void main(String[] args) {  
 SwingUtilities.*invokeLater*(Main::new);  
 }  
}