TASK 4 SINCHANA E

1. DatabaseConnection.java

```java
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class DatabaseConnection {

    private static final String URL =
"jdbc:mysql://localhost:3306/your_database_name";
    private static final String USER = "root";
    private static final String PASSWORD = "";

    public static Connection getConnection() {
        try {
            Class.forName("com.mysql.cj.jdbc.Driver");
            Connection conn = DriverManager.getConnection(URL, USER,
PASSWORD);
            System.out.println("Connected to MySQL database!");
            return conn;
        } catch (Exception e) {
            e.printStackTrace();
            return null;
        }
    }
}
```

2. LoginForm.java

```java
import javax.swing.*;
import java.awt.*;

public class LoginForm extends JFrame {

    public LoginForm() {
        setTitle("Login");
        setSize(350, 280);
        setLocationRelativeTo(null);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
```

```java
        JPanel panel = new JPanel();
        panel.setBackground(new Color(30, 136, 229));
        panel.setLayout(new GridBagLayout());

        JLabel title = new JLabel("User Login");
        title.setForeground(Color.WHITE);
        title.setFont(new Font("Segoe UI", Font.BOLD, 18));

        JTextField userField = new JTextField(15);
        JPasswordField passField = new JPasswordField(15);
        JButton loginBtn = new JButton("Login");

        loginBtn.addActionListener(e -> {
            new Homepage("User").setVisible(true);
            dispose();
        });

        GridBagConstraints gbc = new GridBagConstraints();
        gbc.insets = new Insets(8, 8, 8, 8);

        gbc.gridy = 0; panel.add(title, gbc);
        gbc.gridy = 1; panel.add(userField, gbc);
        gbc.gridy = 2; panel.add(passField, gbc);
        gbc.gridy = 3; panel.add(loginBtn, gbc);

        add(panel);
    }

    public static void main(String[] args) {
        new LoginForm().setVisible(true);
    }
}
```

3. Homepage.java

```java
import javax.swing.*;
import java.awt.*;
import java.time.LocalTime;
import java.time.format.DateTimeFormatter;
import javax.swing.Timer;
```

```java
public class Homepage extends JFrame {

    private JLabel greetingLabel;
    private JLabel timeLabel;

    public Homepage(String username) {
        setTitle("Smart Dashboard");
        setSize(600, 420);
        setLocationRelativeTo(null);
        setDefaultCloseOperation(EXIT_ON_CLOSE);
        setLayout(new BorderLayout());

        // ===== Header Panel =====
        JPanel header = new JPanel(new BorderLayout());
        header.setBackground(new Color(25, 118, 210));
        header.setPreferredSize(new Dimension(600, 80));

        greetingLabel = new JLabel(getGreeting() + ", " + username + "!");
        greetingLabel.setForeground(Color.WHITE);
        greetingLabel.setFont(new Font("Segoe UI", Font.BOLD, 20));
        greetingLabel.setBorder(BorderFactory.createEmptyBorder(10, 20, 0, 0));

        timeLabel = new JLabel();
        timeLabel.setForeground(Color.WHITE);
        timeLabel.setBorder(BorderFactory.createEmptyBorder(0, 20, 10, 0));

        header.add(greetingLabel, BorderLayout.NORTH);
        header.add(timeLabel, BorderLayout.SOUTH);

        // ===== Center Cards =====
        JPanel center = new JPanel(new GridLayout(1, 3, 15, 15));
        center.setBorder(BorderFactory.createEmptyBorder(30, 20, 30, 20));
        center.setBackground(Color.WHITE);

        center.add(createCard(" 👤 ", "Profile"));
        center.add(createCard("⚙", "Settings"));
        center.add(createCard(" 🚪 ", "Logout", true));

        // ===== Footer =====
        JPanel footer = new JPanel();
```

```java
        footer.setBackground(new Color(245, 245, 245));
        footer.add(new JLabel("Status: Connected"));

        add(header, BorderLayout.NORTH);
        add(center, BorderLayout.CENTER);
        add(footer, BorderLayout.SOUTH);

        startClock();
    }

    private JPanel createCard(String icon, String text) {
        return createCard(icon, text, false);
    }

    private JPanel createCard(String icon, String text, boolean logout) {
        JPanel card = new JPanel(new BorderLayout());
        card.setBackground(new Color(227, 242, 253));
        card.setBorder(BorderFactory.createLineBorder(new Color(30, 136, 229)));

        JLabel iconLabel = new JLabel(icon, SwingConstants.CENTER);
        iconLabel.setFont(new Font("Segoe UI Emoji", Font.PLAIN, 40));

        JLabel textLabel = new JLabel(text, SwingConstants.CENTER);
        textLabel.setFont(new Font("Segoe UI", Font.BOLD, 16));

        card.add(iconLabel, BorderLayout.CENTER);
        card.add(textLabel, BorderLayout.SOUTH);

        card.setCursor(new Cursor(Cursor.HAND_CURSOR));

        card.addMouseListener(new java.awt.event.MouseAdapter() {
            public void mouseClicked(java.awt.event.MouseEvent evt) {
                if (logout) {
                    dispose();
                    new LoginForm().setVisible(true);
                }
            }
        });

        return card;
    }
```

```java
private String getGreeting() {
    int hour = LocalTime.now().getHour();
    if (hour < 12) return "Good Morning";
    if (hour < 18) return "Good Afternoon";
    return "Good Evening";
}

private void startClock() {
    Timer timer = new Timer(1000, e -> {
        timeLabel.setText(
            LocalTime.now().format(DateTimeFormatter.ofPattern("HH:mm:ss"))
        );
    });
    timer.start();
}
}
```