import javax.swing.\***;**import java.awt.\***;**import java.awt.event.ActionEvent**;**import java.awt.event.ActionListener**;**import java.time.LocalTime**;**import java.util.Timer**;**import java.util.TimerTask**;**public class LightSwitchApp {  
  
 private JFrame frame**;** private JPanel panel**;** private JButton switchButton**;** private JLabel lightLabel**;** private JLabel bulbIconLabel**;** private boolean isLightOn**;** private String imageFolderPath**;** // Path to image folder  
  
 public LightSwitchApp() {  
 // Define the path to the image folder  
 imageFolderPath = "C:\\Users\\hillarius\\Desktop\\images\\"**;** // Initialize components  
 frame = new JFrame("Light Switch App")**;** panel = new JPanel(new GridBagLayout())**;** // Use GridBagLayout for better component alignment  
 switchButton = new JButton("Turn ON")**;** lightLabel = new JLabel()**;** bulbIconLabel = new JLabel()**;** // Set initial state  
 isLightOn = false**;** updateLightStatus()**;** // Configure components  
 lightLabel.setHorizontalAlignment(SwingConstants.CENTER)**;** lightLabel.setFont(new Font("Arial"**,** Font.BOLD**, 24**))**;** switchButton.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 toggleLight()**;** }  
 })**;** // Arrange components using GridBagConstraints  
 GridBagConstraints gbc = new GridBagConstraints()**;** gbc.insets = new Insets(**10, 10, 10, 10**)**;** gbc.gridx = **0;** gbc.gridy = **0;** gbc.anchor = GridBagConstraints.CENTER**;** panel.add(lightLabel**,** gbc)**;** gbc.gridy = **1;** panel.add(bulbIconLabel**,** gbc)**;** gbc.gridy = **2;** panel.add(switchButton**,** gbc)**;** frame.add(panel)**;** frame.setSize(**500, 500**)**;** frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE)**;** frame.setVisible(true)**;** // Schedule automatic light toggle based on time  
 scheduleAutomaticToggle()**;** }  
  
 private void toggleLight() {  
 isLightOn = !isLightOn**;** updateLightStatus()**;** }  
  
 private void updateLightStatus() {  
 if (isLightOn) {  
 lightLabel.setText("The light is ON")**;** panel.setBackground(Color.WHITE)**;** switchButton.setText("Turn OFF")**;** setBulbIcon(imageFolderPath + "light\_on.jpeg")**;** // Use JPEG for ON state  
 } else {  
 lightLabel.setText("The light is OFF")**;** panel.setBackground(Color.BLACK)**;** switchButton.setText("Turn ON")**;** setBulbIcon(imageFolderPath + "light\_off.jpeg")**;** // Use JPEG for OFF state  
 }  
 }  
  
 private void setBulbIcon(String imagePath) {  
 // Load the image and resize it to fit within the JLabel  
 ImageIcon originalIcon = new ImageIcon(imagePath)**;** Image scaledImage = originalIcon.getImage().getScaledInstance(**200, 200,** Image.SCALE\_SMOOTH)**;** // Resize to 200x200  
 bulbIconLabel.setIcon(new ImageIcon(scaledImage))**;** }  
  
 private void scheduleAutomaticToggle() {  
 Timer timer = new Timer(true)**;** timer.scheduleAtFixedRate(new TimerTask() {  
 @Override  
 public void run() {  
 LocalTime now = LocalTime.now()**;** if (isNightTime(now)) {  
 if (!isLightOn) {  
 isLightOn = true**;** SwingUtilities.invokeLater(() -> updateLightStatus())**;** }  
 } else {  
 if (isLightOn) {  
 isLightOn = false**;** SwingUtilities.invokeLater(() -> updateLightStatus())**;** }  
 }  
 }  
 }**, 0, 60** \* **1000**)**;** // Check every minute  
 }  
  
 private boolean isNightTime(LocalTime time) {  
 // Define night time as between 6:00 PM and 6:00 AM  
 return time.isAfter(LocalTime.of(**18, 0**)) || time.isBefore(LocalTime.of(**6, 0**))**;** }  
  
 public static void main(String[] args) {  
 SwingUtilities.invokeLater(() -> new LightSwitchApp())**;** }  
}