# GUI components:

## JFrame: The main window of the application.

The ApplicationFrame class extends JFrame to create the main application window.

The constructor of the ApplicationFrame class sets up the window properties such as size, layout, and title.

## JPanel: A panel to contain all the other components.

In the ApplicationFrame constructor, a JPanel object is created and added to the JFrame.

The layout of the JPanel is set to GridLayout to organize the components in a grid-like structure.

## JButton: Buttons for performing different actions.

JButton objects are created for each action in the application (parking a car, displaying car park details, etc.).

The buttons are added to the JPanel.

Each button is assigned an ActionListener, which is the ApplicationFrame object itself (since it implements ActionListener). This allows the actionPerformed method to handle button clicks.

## JTextArea: A text area to display the output of the actions performed by the user.

A JTextArea object is created and added to the JPanel to display the output of the user's actions.

The JTextArea is set to be non-editable, so the user cannot modify the text displayed.

# Event handling functions:

## actionPerformed(ActionEvent e):

This method handles events triggered by the buttons.

* The method uses if-else statements to determine which button was clicked based on the event source (e.getSource()).
* For each button, the corresponding action is performed. For example, when Button 1 is clicked, a car is parked in a parking slot with the user input.
* For Buttons 1, 2, 4, 5, and 6, the user is prompted to enter information such as parking slot number, owner name, car registration number, and owner role through JOptionPane.showInputDialog().
* When Button 3 is clicked it displays all the car parked in a descriptive manner.
* The actionPerformed method updates the JTextArea with the results of the action performed, such as displaying the car park state, finding a parking slot, or removing a car.
* The background colour of the button is also updated based on the type of parking slot (visitor or staff) or the action performed.
* Button 7 handles the exit confirmation and closes the application if the user confirms the exit.

The combination of these GUI components and event handling functions creates an intuitive interface for the Car Park Management System, allowing users to interact with the system easily.