```java
package project3;

import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.GregorianCalendar;

public class boughtOnDialogCar extends JDialog implements ActionListener {

    private JTextField txtName;
    private JTextField txtDate;
    private JTextField txtTrimPackage;
    private JTextField turbo;
    private JTextField txtCost;
    private JButton okButton;
    private JButton cancelButton;
    private JComboBox<String> combobox;
    private int closeStatus;
    private Auto auto;
    static final int OK = 0;
    static final int CANCEL = 1;

    private static final int[] DAYS_IN_MONTH = {0, 31, 28, 31, 30, 31, 30, 31,
            31, 30, 31, 30, 31};

    /*****************************************************
     Instantiate a Custom Dialog as 'modal' and wait for the
     user to provide data and click on a button.

     @param parent reference to the JFrame application
     @param auto an instantiated object to be filled with data
     *****************************************************/

    public boughtOnDialogCar(JFrame parent, Auto auto) {
        // call parent and create a 'modal' dialog
        super(parent, true);

        this.auto = auto;
        setTitle("Buying a Car");
        closeStatus = CANCEL;
        setSize(400,200);

        // prevent user from closing window
        setDefaultCloseOperation(WindowConstants.DO_NOTHING_ON_CLOSE);

        // instantiate and display two text fields
        txtName = new JTextField("Fusion",30);
        txtDate = new JTextField(15);
        turbo = new JTextField("True",15);
        txtTrimPackage = new JTextField("ST",15);
        txtCost = new JTextField("10000.00", 15);

        String[] autoStrings = {"Car"};

        combobox = new JComboBox<>(autoStrings);
        txtDate.setText("8/8/2019");
        JPanel textPanel = new JPanel();
        textPanel.setLayout(new GridLayout(7,2));

        textPanel.add(new JLabel(""));
        textPanel.add(combobox);
        textPanel.add(new JLabel(""));
        textPanel.add(new JLabel(""));

        textPanel.add(new JLabel("Name of Car: "));
        textPanel.add(txtName);
        textPanel.add(new JLabel("bought on Date: "));
        textPanel.add(txtDate);
        textPanel.add(new JLabel("Trim Package: "));
        textPanel.add(txtTrimPackage);
        textPanel.add(new JLabel("Turbo: "));
        textPanel.add(turbo);
        textPanel.add(new JLabel("Amount Paid for: "));
        textPanel.add(txtCost);

        getContentPane().add(textPanel, BorderLayout.CENTER);

        // Instantiate and display two buttons
        okButton = new JButton("OK");
        cancelButton = new JButton("Cancel");
        JPanel buttonPanel = new JPanel();
        buttonPanel.add(okButton);
        buttonPanel.add(cancelButton);
        getContentPane().add(buttonPanel, BorderLayout.SOUTH);
        okButton.addActionListener(this);
        cancelButton.addActionListener(this);

        setVisible (true);
    }

    /*************************************************************
     Respond to either button clicks
     @param e the action event that was just fired
     ************************************************************/
    public void actionPerformed(ActionEvent e) {

        JButton button = (JButton) e.getSource();

        // if OK clicked the fill the object
        if (button == okButton) {
            // save the information in the object
            closeStatus = OK;
            SimpleDateFormat df = new SimpleDateFormat("MM/dd/yyyy");
            GregorianCalendar temp = new GregorianCalendar();
            String tempName;
            double tempCost;
            int day;
            int month;
            int year;

            try {
                tempName = txtName.getText();
                tempCost = Double.parseDouble(txtCost.getText());
                String[] dates = txtDate.getText().split("/");
                String months;
                String days;
                String years;
                if (dates.length == 3){
                    months = dates[0];
                    days = dates[1];
                    years = dates[2];
                }
                else
                    throw new IllegalArgumentException();

                month = Integer.parseInt(months);//Converts the Strings into integers
                day = Integer.parseInt(days);
                year = Integer.parseInt(years);
                if (month < 1 || day < 1 || year < 1950 || month > 12)
                    throw new IllegalArgumentException();

                if (!isLeapYear(year)) {
                    if (day > DAYS_IN_MONTH[month])
                        throw new IllegalArgumentException();
                }
                else if (month == 2 && day > 29) {
                    throw new IllegalArgumentException();
                }
                if (tempName.equals(""))
                    throw new Exception();
                if (tempCost <= 0)
                    throw new NumberFormatException();
            }
            catch (NumberFormatException e2){
                JOptionPane.showMessageDialog(null, "Enter a cost above 0");
                return;
            }
            catch (IllegalArgumentException e5){
                JOptionPane.showMessageDialog(null,
                        "Enter a correct date with the format month/day/year");
                return;
            }
            catch (Exception e3){
                JOptionPane.showMessageDialog(null, "Enter the name of the Car");
                return;
            }

            if (combobox.getSelectedIndex() == 1) {
                Date d = null;
                try {
                    d = df.parse(txtDate.getText());
                    temp.setTime(d);

                } catch (ParseException e1) {
```

```java
                    JOptionPane.showMessageDialog(null, "Invalid Date");
                    //unreachable because the date is checked before this
                }
                auto.setBoughtOn(temp);
                auto.setAutoName(txtName.getText());
                ((Car) auto).setTrim(txtTrimPackage.getText());
                auto.setBoughtCost(Double.parseDouble(txtCost.getText()));

            }

            else {
                Date d = null;
                try {
                    d = df.parse(txtDate.getText());
                    temp.setTime(d);

                } catch (ParseException e1) {
                    JOptionPane.showMessageDialog(null, "Invalid Date");
                    //unreachable because the date is checked before this
                }

                auto.setBoughtOn(temp);
                auto.setAutoName(txtName.getText());
                auto.setBoughtCost(Double.parseDouble(txtCost.getText()));
                ((Car) auto).setTrim(txtTrimPackage.getText());

                if (turbo.getText().equalsIgnoreCase("true"))
                    ((Car) auto).setTurbo(true);
                else
                    ((Car) auto).setTurbo(false);

            }
        }

        if (button == cancelButton){
            txtCost.setText("50000");
        }

        // make the dialog disappear
        dispose();
    }

    /************************************************************
     Return a String to let the caller know which button
     was clicked

     @return an int representing the option OK or CANCEL
     ************************************************************/
    public int getCloseStatus(){
        return closeStatus;
    }

    public static boolean isLeapYear(int year) {
        return year % 4 == 0 && (year % 100 != 0 || year % 400 == 0);
    }
}
```