```java
package rolodex;

import javax.swing.*;
import javax.swing.border.*;
import java.awt.*;
import java.awt.event.*;
import mvp.*;

/****
 *
 * Class ChangeCardDialog is a view for displaying a selected card to be
 * changed.  Since it is very similar in content to the dialog for adding a new
 * card, ChangeCardDialog extends <a href="AddCardDialog.html"> AddCardDialog,
 * </a> q.v.  The differences in the change vs add dialogs are the message at
 * the top of the dialog, and the use of ChangeCardButtons instead of
 * AddCardButtons.  The OK button listener in ChangeCardButtons calls
 * Rolodex.change instead of Rolodex.add.
 *
 */

public class ConfirmChangeDialog extends AddCardDialog {

    /*
     * Construct this by calling the parent constructor and then specializing
     * the buttons subview.  I.e., change the AddCardButtons to
     * ConfirmChangeButtons.
     */
    public ConfirmChangeDialog(Rolodex r, RolodexMenuUI rmui) {

        /*
         * Call the parent constructor.
         */
        super(r, rmui);

        /*
         * Clear set the parent AddCardButtons data member to
         */
        acb = ccb = new ConfirmChangeButtons(r, rmui, this);
    }

    /**
     * Compose this simply by calling the parent compose changing the message
     * label at the top of the dialog.
     */
    public void compose() {

        /*
         * Call the parent compose function.
         */
        super.compose();

        /*
         * Change the prompting message text.
         */
        top_label.setText("Change the following card:");
    }

    /**
     * Display the card selected by the user for changing.  Also, save the card
     * id entered by the user in the change card dialog.  This id value will be
     * used subsequently by <a href=
     * "OKConfirmChangeButtonListener.html#actionPerformed(java.awt.event.ActionEvent)"
     * > OKConfirmChangeButtonListener.actionPerformed,</a> q.v.

     */
    public void showCardAndSaveId(Card c, Id id) {
        name_jtf.setText(c.getName().stringValue());
        id_jtf.setText(c.getId().integerValue().toString());
        age_jtf.setText(c.getAge().integerValue().toString());
        sex_jtf.setText(c.getSex().stringValue());
        addr_jtf.setText(c.getAddr().stringValue());

        this.id = id;
    }

    /**
     * Return the id of the card to be changed.  This is the original card id
     * entered by the user in the change card dialog.
     */
    public Id getOriginalId() {
        return id;
    }


    /** Id of card to changed entered by user. */
    protected Id id;

    /** Subview containing add card buttons OK, Clear, and Cancel */
    protected ConfirmChangeButtons ccb;

}
```