```
/****
 *
 * Implementation of find-card-dialog.h.
 *
 */

#include "card-list.h"
#include "find-card-dialog.h"
#include "label.h"
#include "rolodex-menu-ui.h"
#include <InterViews/box.h>
#include <InterViews/button.h>
#include <InterViews/frame.h>
#include <InterViews/glue.h>
#include <InterViews/streditor.h>
#include <stdlib.h>


FindCardDialog::FindCardDialog(Rolodex* r, RolodexMenuUI* rmui,
    FoundCardsDialog* fcsd) :
        View(NULL, r) {
    okfb = new OKFindButton(r, rmui, this, fcsd);
    cfb = new CancelFindButton(rmui);
}

FindCardDialog::~FindCardDialog() {
}

void FindCardDialog::Compose() {
    /*
     * Make the window of this a VBox.
     */
    w = new VBox();
    ((VBox*) w)->Align(Center);

    /*
     * Stick some stretchable glue spacing above the prompting message.
     */
    ((VBox*) w)->Insert(new VGlue(round(0.20 * inch), 0, round(0.50 * inch)));

    /*
     * Insert the prompting message into the top of the vbox.  The prompting
     * message itself is in an hbox with some glue on each side.  This glue
     * will enable horizontal stretching of the message, which stretchability
     * will transfer out to the entire scene that the message is in.
     */
    ((VBox*) w)->Insert(new Label(
        "Enter Name of Card(s) to Find:",
            "times", "bold", "r", 18));

    /*
     * Stick some stretchable glue spacing between the message and the field
     * editors below.
     */
    ((VBox*) w)->Insert(new VGlue(round(0.20 * inch), 0, round(0.50 * inch)));

    /*
     * Insert the string editor to enter the card id.  The subsequent call to
     * Message("") clears the blank chars out after the editor is built and
     * sized.  In this way, the editor properly sized and initially empty when
     * user frist types in it.
     */
    ((VBox*) w)->Insert(new ShadowFrame(sen =
         new StringEditor(bs, "                         ")));
    sen->Message("");

    /*
     * Compose the OK and Cancel dialog buttons and insert into the window.
     */
    ComposeButtons();
}

void FindCardDialog::ComposeButtons() {
    /*
     * Make an HBox to hold the buttons
     */
    hb = new HBox();

    /*
     * Enter the buttons into the box, delimited with glue.
     */
    hb->Insert(new HGlue(round(0.2 * inch), hfil, hfil));
    hb->Insert(okfb);
    hb->Insert(new HGlue(round(0.2 * inch), hfil, hfil));
    hb->Insert(cfb);
    hb->Insert(new HGlue(round(0.2 * inch), hfil, hfil));

    /*
     * Put the hbox into the window, with some glue.
     */
    ((VBox*)w)->Insert(new VGlue(round(0.2 * inch), 0, vfil));
    ((VBox*)w)->Insert(hb);
    ((VBox*)w)->Insert(new VGlue(round(0.2 * inch), 0, vfil));
}

String* FindCardDialog::GetName() {
    return new String(sen->Text());
}

OKFindButton::OKFindButton(Rolodex* r, RolodexMenuUI* rmui,
    FindCardDialog* fcd, FoundCardsDialog* fcsd) :
        PushButton("OK", new ButtonState(), true) {
                // ^^^   ^^^^^^^^^^^^^^^^^  ^^^^ Default starting state; since
                // ^^^   ^^^^^^^^^^^^^^^^^  ^^^^ we will not use ButtonStates,
                // ^^^   ^^^^^^^^^^^^^^^^^  ^^^^ our starting state is always
                // ^^^   ^^^^^^^^^^^^^^^^^  ^^^^ true
                // ^^^   ^^^^^^^^^^^^^^^^^
                // ^^^   ^^^^^^^^^^^^^^^^^ Dummy value for the ButtonState of
                // ^^^   ^^^^^^^^^^^^^^^^^ this button.  We will not normally
                // ^^^   ^^^^^^^^^^^^^^^^^ use ButtonState objects
                // ^^^
                // ^^^ Text label inside button
```

```
113
114     this->r = r;
115     this->rmui = rmui;
116     this->fcd = fcd;
117     this->fcsd = fcsd;
118 }
119
120 void OKFindButton::Press() {
121     CardList* cl = r->Find(fcd->GetName());
122     if (cl->Len() == 0)
123         rmui->ShowNoFoundDialog();
124     else
125         rmui->ShowFoundDialog();
126         fcsd->ShowResults(cl);
127 }
128
129 CancelFindButton::CancelFindButton(RolodexMenuUI* rmui) :
130         PushButton("Cancel", new ButtonState(), true) {
131     this->rmui = rmui;
132 }
133
134 void CancelFindButton::Press() {
135     rmui->ShowEmptyDialog();
136 }
```