```java
package rolodex;

import mvp.*;

/****
 *
 * Class Main contains the main Java function for the Rolodex system.  It
 * constructs instances of the model, view, and process classes for the Rolodex
 * and gives control to the view, which will manage communication with the end
 * user.
 *                                                                          <p>
 * This is an initial implementation for GUI testing purposes, with no process
 * classes yet defined.
 *
 */

public class Main {

    /**
     * Construct models, construct views, compose views, and fire the pupply
     * up.
     */
    public static void main(String[] args) {
        Screen s;                       // The GUI screen
        RolodexTool rt;                 // The top-level Rolodex model
        RolodexMenuUI rui;              // The top-level Rolodex view

        /*
         * Construct the GUI screen, thereby initializing the GUI system.  In
         * this Java-based implementation, the GUI screen is managed entirely
         * by the Java runtime environment, so the only thing to do in the way
         * of screen construction is to set its look and feel, if desired.  In
         * other GUI toolkits, screen construction may involve more substantial
         * work.
         */
        s = new Screen();

        /*
         * Construct the top-level Rolodex model.  It will in turn construct
         * all subsidiary model classes and the data objects they require.
         * Note that the view parameter is null, since the view has not yet
         * been constructed.  After it is, we call the rt.setView method, which
         * is inherited from class Model.
         */
        rt = new RolodexTool(null);

        /*
         * Construct and compose the Rolodex Tool view.  Compose will lay out
         * all GUI components, including dialogs that appear during the course
         * of user interaction.
         */
        rui = new RolodexMenuUI(s, rt);
        rui.compose();

        /*
         * Store the view in the model to enable two-way communication between
         * the model and the view.
         */
        rt.setView(rui);

        /*
         * Display the top-level view window on the UI screen.
         */
        rui.show();

        /*
         * In Java, no call to View.run is necessary, since showing any window
         * on the screen automatically starts the event loop.  The program
         * subsequently exits when the System.exit function is called, e.g.,
         * in response to the user choosing the File Quit menu item.
         */

    }

}
```