

```

package doc.tutorial;

import java.awt.Container;

public class TableDisplay extends AbstractPlaceableActor {
    private JTable _table;
    private Tableau _tableau;

    public TableDisplay(CompositeEntity container, String name)
        throws IllegalArgumentException, NameDuplicationException {
        super(container, name);
    }

    public void initialize() throws IllegalArgumentException {
        super.initialize();
        if (_table == null) {
            Effigy containerEffigy = Configuration.findEffigy(toplevel());
            try {
                _tableau = new Tableau(containerEffigy, "tableau");
            } catch (NameDuplicationException e) {
                throw new IllegalArgumentException(this, e, "Failed to create tableau.");
            }
            _frame = new TableauFrame(_tableau);
            setFrame(_frame);
            _tableau.setFrame(_frame);
            place(_frame.getContentPane());
            _frame.pack();
        }
        if (_frame != null) {
            ((TableauFrame)_frame).show();
            _frame.toFront();
        }
    }

    public void place(Container container) {
        if (container == null) {
            if (_frame != null) {
                _frame.dispose();
            }
            _frame = null;
            _table = null;
            if (_tableau != null) {
                try {
                    _tableau.setContainer(null);
                } catch (Exception e) {
                    throw new InternalErrorException(e);
                }
            }
            _tableau = null;
        } else {
            TableModel dataModel = new AbstractTableModel() {
                public int getColumnCount() { return 10; }
                public int getRowCount() { return 10; }
                public Object getValueAt(int row, int col) { return new Integer(row*col); }
            };
            _table = new JTable(dataModel);
            container.add(_table);
        }
    }
}

```