Chapter 31 JTable and JTree

1. You can initialize a table using the constructor of JTable. You cannot specify the maximum number of visible rows in a table without scrolling. You can specify the height of a table cell using the setRowHeight method. You can specify the horizontal margin of table cells using the setIntercellSpacing method.

2. To modify table contents visually from the UI, the table cells must be editable with an associated editor for the cell. You must also save the change through the data model. To add or remove a row, you must use a TableModel. The DefaultTableModel class provides methods for adding and removing rows. To add a column, you may use the addColumn method in DefaultTableModel or in DefaultTableColumnModel. To remove a column, you must use the removeTableColumn method from the DefaultTableColumnModel class.

3. JTable has the autoResizingMode property that can be used to auto resize a table column. Possible values are:

   - JTable.AUTO_RESIZE_OFF
   - JTable.AUTO_RESIZE_LAST_COLUMN
   - JTable.AUTO_RESIZE_SUBSEQUENT_COLUMNS
   - JTable.AUTO_RESIZE_NEXT_COLUMN
   - JTable.AUTO_RESIZE_ALL_COLUMNS

4. The properties to show grids, horizontal grids, and vertical grids are showGrid, showHorizontalGrid, and showVerticalGrid. The properties to specify the table row height, and vertical margin are rowHeight and rowMargin. There are no methods to set horizontal margins between the cells, because this type of margin is flexible.

5. By default, a cell object's string representation is displayed and the string can be edited as it was in a text field. JTable maintains a set of predefined renderers and editors, listed in Table 8.1, which can be specified to replace default string renderers and editors. You can create a custom renderer by extending the DefaultTableCellRenderer class.

6. To create a tree, simply use the constructor of JTree. To specify the row height of a tree node, use the rowHeight property. To obtain the default tree model and tree selection model from an instance of JTree, use getModel and getSelectionModel methods.
7. You have to declare a custom table model and implement the `isCellEditable(row, column)` method to return false. By default, all cells are editable.

8. To initialize data in a tree using `TreeModel`, you need to create nodes using the `DefaultMutableTreeNode` class, and set the root with the `TreeModel` (or `DefaultTreeModel`). To add a child to an instance of `DefaultMutableTreeNode`, use the `add` method.

9. To initialize data in a tree using `TreeModel`, you need to create nodes using the `DefaultMutableTreeNode` class, and set the root with the `TreeModel` (or `DefaultTreeModel`). To add a child to an instance of `DefaultMutableTreeNode`, use the `add` method.

10. To enable tree node editing, set `jTree`’s editable property to `true`. By default, it is `false`.

11. To add a node from a tree, use the `add` method from an instance of `DefaultMutableTreeNode` in a parent. To remove a node from a tree, use the `removeNodeFromParent` method from `TreeModel`.

12. To obtain a selected tree node, use the `getLeadSelectedPath` method to get the path, then get the node from the path using the `getLastPathComponent` method.