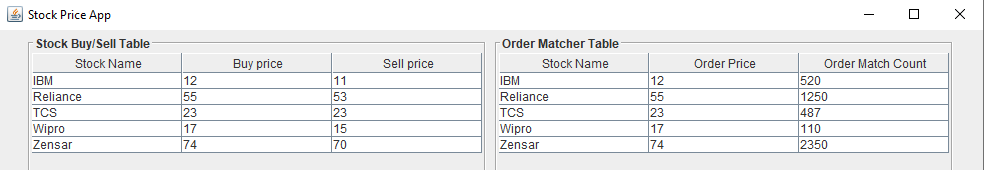
**Stock Price Application**



Stock price application is used to display current buy/sell price of stocks & it also does order matching. In the above diagram, there are 2 tables ‘Stock Buy/Sell Table’ & ‘Order Matcher Table’. The ‘Stock Buy/Sell Table renders the latest buy & sell prices of different stocks like IBM, Zensar etc. Note that these prices change frequently. The second table ‘Order Matcher Table’ matches the order if buy price is equals to or more than sell price. Thus, ‘Order Price’ also changes frequently. Write an application that simulates Stock Price Application. Steps to follow:

1. Create separate threads for producing Buy price, Sell price & Order matching.
2. Buy/Sell price threads will continuously produce values using random number & render them into the table.
3. Order Matcher thread will pick up recent 5 buy prices & 5 sell prices for a every stock, calculate average buy price & average sell price for that stock & then if buy price >= sell price then displays the buy price into ‘Order Price’ column as well as increment total matched order count under ‘Order Match Count’ column.

**Note:**

* You can use repaint() method to update the UI.
* Stock names like IBM, Zensar etc are fixed. They are sorted in ascending order.
* First implement the application using normal thread class & then try it with Thread pool using executor framework.
* Please copy the basic UI code given below & then add your code inside.

*package com.zensar.stockapp;*

*import java.awt.Container;*

*import java.awt.FlowLayout;*

*import javax.swing.BorderFactory;*

*import javax.swing.JFrame;*

*import javax.swing.JScrollPane;*

*import javax.swing.JTable;*

*public class StockMatcherApp extends JFrame {*

*JTable buySellTable, orderMatcherTable;*

*public StockMatcherApp()*

*{*

*Container contentPane = getContentPane();*

*contentPane.setLayout(new FlowLayout());*

*String[][] stockData = {*

*{ "IBM", "12", "11"},*

*{ "Reliance", "55", "53" },*

*{ "TCS", "23", "23" },*

*{ "Wipro", "17", "15" },*

*{ "Zensar", "74", "70" }*

*};*

*String[] stockColumnNames = { "Stock Name", "Buy price", "Sell price" };*

*String[] orderMatchColumnNames = {"Stock Name", "Order Price", "Order Match Count"};*

*String[][] orderMatchData = {*

*{ "IBM", "12", "520"},*

*{ "Reliance", "55", "1250" },*

*{ "TCS", "23", "487" },*

*{ "Wipro", "17", "110" },*

*{ "Zensar", "74", "2350" }*

*};*

*// Initializing the JTable*

*buySellTable = new JTable(stockData, stockColumnNames);*

*orderMatcherTable = new JTable(orderMatchData, orderMatchColumnNames);*

*JScrollPane spBuySell = new JScrollPane(buySellTable);*

*spBuySell.setBorder(BorderFactory.createTitledBorder(BorderFactory.createEtchedBorder(), "Stock Buy/Sell Table"));*

*JScrollPane spOrderMatcher = new JScrollPane(orderMatcherTable);*

*spOrderMatcher.setBorder(BorderFactory.createTitledBorder(BorderFactory.createEtchedBorder(), "Order Matcher Table"));*

*contentPane.add(spBuySell);*

*contentPane.add(spOrderMatcher);*

*}*

*public static void main(String[] args)*

*{*

*JFrame jframe = new StockMatcherApp();*

*jframe.setBounds(200, 200, 1000, 300);*

*jframe.setTitle("Stock Price App");*

*jframe.setVisible(true);*

*}*

*}*