```
/*
        Author's Name:              Kenneth Larot Yamat

        Purpose of Program:         To create a program that automatically creates
                                    trading tickets for a security, for example, buy
                                    and sell orders for shares of an exchange traded fund.

        Date Due:                   11:59 PM on March 4th, 2024
*/

Project Proposal:       To create a program that automatically creates
                        trading tickets for a security, for example, buy
                        and sell orders for shares of an exchange traded fund.

                        A user would only manually enter the first order, either to
                        buy or sell a security, the program would populate and submit
                        a new ticket based on the fulfillment of the previous ticket,
                        the chain of tickets would continue until the user decided to
                        cancel the chain.

A) Background and the needs:

                        This program is needed because there are many securities
                        that are difficult to trade because they are illiquid as
                        a result of large bid and ask spreads, or because they lack
                        volume.

                        The goal is to reduce spreads while increasing volume.

                        Another need is due to the fact that manually performing this task
                        is laborious and prone to error.


B) Function list:

                        getSecurityPrice
                        setSecurityPrice

                        setPurchasePrice
                        getPurchasePrice

                        setLiquidationPrice
                        getLiquidationPrice

                        setAvarageTrueRange
                        getAverageTrueRange

                        setBollingerBandWidth
                        getBollingerBandWidth

                        setAverageDirectionalIndex
                        getAverageDirectionalIndex

C) User interface (UI) design:

    Step 1


                Trade Ticket
                ============

                Security:           [User Input Element]
                Buy or Sell:        [User Input Element]
                Limit:              [User Input Element]
                Quantity:           [User Input Element]
                ATR:                [User Input Element]
                BBW:                [User Input Element]
                ADX:                [User Input Element]

    Step 2


                Your initial [Buy/Sell] Trade ticket for [Security] has been
                submitted at the following price [Limit Price] and quantity [Quantity].

                Subsequent orders will be automatically generated and submitted contingent
                upon the fulfillment of the previous order, with buy limits and sell limits
                based on the Average True Range, Bollinger Band Width, and Average Directional
                Index entered on the initializing ticket.

                Sell orders will be generated with a limit of   [Calculated Amount] above the previously filled ticket
                Buy  orders will be generated with a limit of   [Calculated Amount] below the previously filled ticket

                [   User Input Element   [Accept and Submit]   [Override and Submit]   [Start Over]   ]

    Step 3

                [Ticker Symbol]     [Buy/Sell]          [Order Quantity]            [Limit Price]               [Ticket Status]

                [   HFH.P                   Buy                     1                           86.86                   Open
  ]

    Step 4
                [Ticker Symbol]     [Buy/Sell]          [Order Quantity]            [Limit Price]               [Ticket Status]
```

```
 94
 95                [   HFH.P                   Buy                     1                           86.86                   Filled
   ]
 96
 97    Step 5
 98
 99                [Ticker Symbol]     [Buy/Sell]          [Order Quantity]            [Limit Price]               [Ticket Status]
100
101                [   HFH.P                   Sell                    1                           86.89                   Open
   ]
102
103    Step 6
104
105                [Ticker Symbol]     [Buy/Sell]          [Order Quantity]            [Limit Price]               [Ticket Status]
106
107                [   HFH.P                   Sell                    1                           86.89                   Filled
   ]
108
109    Step 7
110
111                [Ticker Symbol]     [Buy/Sell]          [Order Quantity]            [Limit Price]               [Ticket Status]
112
113                [   HFH.P                   Buy                     1                           86.87                   Open
   ]
114
115    Step 8
116
117                [Ticker Symbol]     [Buy/Sell]          [Order Quantity]            [Limit Price]               [Ticket Status]
118
119                [   HFH.P                   Buy                     1                           86.87                   Filled
   ]
120
121    Notes:      this sequence is based on + 00.03 to Sell orders and - 00.02 to Buy orders for first issue preferred shares for
   the security HFH
122
123 D) Class diagram
124
125
126
127 E) File and database design:
128
129 F) Expectations of project fulfillment:
130
131    a.  [Ticket]                instantiates based on user input.
132
133        [AutoTicket]            instantiates based on fulfillment of previous ticket.
134
135    b.
136
137    c.
138
139    d.  [Arraylist]             Arraylist will be used to log the sequence of trades
140
141    e.  [Exception handling]    a user may enter alphabetical values in a field that requires an int or
142                                double, and vice versa, an invalid data message will prompt the user.
143
144    f.  [Database]
145
146    g.
147        [Documentation]         very detailed and elaborate notes will be included in every program,
148                                class, method, and attribute regarding the purpose, design, development,
149                                and miscellaneous other notes as well.  JavaDoc will see extensive use.
150
151 G) Project Report
152
153    a.
154    b.
155    c.
156    d.
157    e.
158    f.
159    g.
160    h.
161
162
163
```