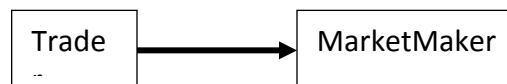


Testing Report: Project 3

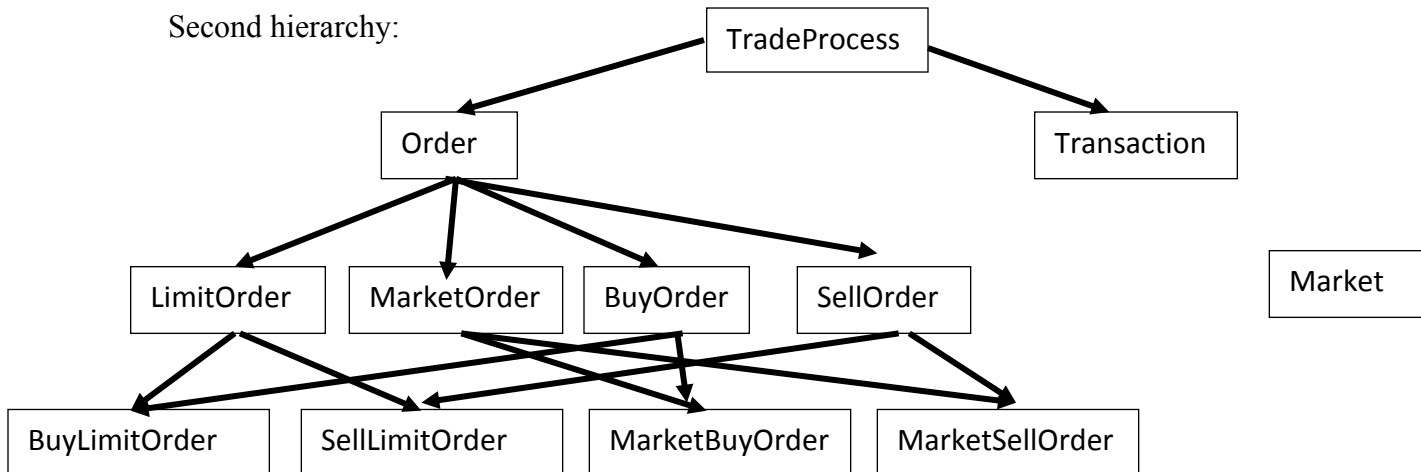
In this project, I create 2 hierarchies.

- The first Hierarchy included **Trader** and **MarketMaker**
- The first Hierarchy included **TraderProcess** (abstract), **Order** (abstract), **Transaction**, **LimitOrder** (interface), **MarketOrder** (interface), **BuyLimitOrder**, **SellLimitOrder**, **MarketBuyOrder**, **MarketSellOrder**, **BuyOrder** (interface), **SellOrder** (interface).

First hierarchy:



Second hierarchy:



Tester

In this project, I create 3 tester for each Hierarchy and Market class.

1. TraderHierarchyTest

In this test, I create 5 instances:

```
Trader trader1 = new Trader("Harley");  
Trader trader2 = new Trader("Harold");  
Trader trader3 = new Trader("Harley");  
MarketMaker marketMaker1 = new MarketMaker("Jason", 1000, 100.0);  
MarketMaker marketMaker2 = new MarketMaker("Harold", 500, 20.0);
```

- *Test the getName() method*

I tested the method by using getName() for each type. The returned values are match with expected values

- ***Test the setName() method***

I test the method by setting new name for each instance. Then I used the getName() method to call the name of the instances.

- ***Test the getDefaultOrderSize() method***

I test the method by using getDefaultOrderSize() for only marketMaker1 and marketMaker2. The returned values match the expected values.

- ***Test the setDefaultOrderSize() method***

I test the method by setting new default order size to marketMaker1 and marketMaker2. Then I used the getDefaultOrderSize() to return the values. The returned values match the expected values

- ***Test the getPriceOffset() method***

I test the method by using getPriceOffset() on marketMaker1 and marketMaker2. The returned values match the expected values

- ***Test the setPriceOffset() method***

I test the method by setting new price offset to marketMaker1 and marketMaker2. Then I used the getPriceOffset() to return the values. The returned values match the expected values.

- ***Test the toString() method***

I format the output string by calling all values of each instance. The returned values match the expected values.

- ***Test the equals() method***

I tested if there's identical orders. The returned values match the expected values.

2. TradeProcessHierarchyTester

I create 5 instances for BuyLimitOrder, SellLimitOrder, MarketBuyOrder, MarketSellOrder, Transaction.

```
BuyLimitOrder bLimitOrder = new BuyLimitOrder('A', 100, 50, true, true, trader1);  
SellLimitOrder sLimitOrder = new SellLimitOrder('B', 400, 32, true, true, trader2);  
MarketBuyOrder bMarketOrder = new MarketBuyOrder('C', 2000, 100, marketMaker1);  
MarketSellOrder sMarketOrder = new MarketSellOrder('D', 400, 50, marketMaker2);  
Transaction successOrder = new Transaction('E', 100, 50, trader1, marketMaker1, 23);
```

- ***Test the getStockSymbol() method***

I test the method by using getStockSymbol() on all instances. The returned values match the expected values.

- ***Test the `getNumberShares()` method***

I test the method by using `getNumberShares()` on all instances. The returned values match the expected values.

- ***Test the `setNumberShares()` method***

I test the method by setting new number shares to `bLimitOrder`, `sLimitOrder`, `bMarketOrder`, `sMarketOrder`. Then I used the `getNumberShares()` method to return the values. The returned values match the expected values.

- ***Test the `getPrice()` method***

I test the method by using `getPrice()` on all instances. The returned values match the expected values.

- ***Test the `setPrice()` method***

I test the method by setting new price to `bLimitOrder`, `sLimitOrder`, `bMarketOrder`, `sMarketOrder`. Then I used the `getPrice()` method to return the values. The returned values match the expected values.

- ***Test the `isAllOrNone()` method***

I test the method by using `isAllOrNone` on `bLimitOrder`, `sLimitOrder`, `bMarketOrder`, `sMarketOrder`. The returned values match the expected values.

- ***Test the `isDayOrder()` method***

I test the method by using `isDayOrder()` on `bLimitOrder`, `sLimitOrder`, `bMarketOrder`, `sMarketOrder`. The returned values match the expected values.

- ***Test the `getTrader()` method***

I test the method by using `getTrader()` on `bLimitOrder`, `sLimitOrder`, `bMarketOrder`, `sMarketOrder`. The returned values match the expected values.

- ***Test the `toString()` method***

I format the output string by calling all values of each instance. The returned values match the expected values.

- ***Test the `equals()` method***

I tested if there's identical orders. The returned values match the expected values.

3. MarketTester

I first create 3 new instances for `Trader`, 1 new instance for `MarketMaker`, 4 new instances for `LLNode<Order>`, 4 new `Markets`.

```
Trader trader1 = new Trader("Harley");
Trader trader2 = new Trader("Harold");
Trader trader3 = new Trader("Jason");
MarketMaker trader4 = new MarketMaker("Jen", 1000, 125);
```

```

LLNode<Order> buyNode = new LLNode<Order> (new MarketBuyOrder('M', 100,90,
trader3), null);
LLNode<Order> buyList = new LLNode<Order> (new BuyLimitOrder('N', 120, 253, true, true,
trader1), null);
LLNode<Order> sellNode = new LLNode<Order> (new MarketSellOrder('O', 500, 200,
trader4), null);
LLNode<Order> sellList = new LLNode<Order> (new SellLimitOrder('P', 235, 100, true, true,
trader2), null);

```

```

Market market 1 = new Market ('M', buyList, sellList);
Market market2 = new Market ('N', buyList, sellNode);
Market market3 = new Market ('O', buyNode, sellList);
Market market4 = new Market('P', buyNode, sellNode);

```

- ***Test the getStockSymbol() method***

I tested the method by using getStockSymbol() on 4 market instances. The returned values match the expected values.

- ***Test the getSellOrder() method***

I tested the method by using getSellOrder() on 4 market instances. The returned values match the expected values.

- ***Test the getBuyOrder() method***

I tested the method by using getBuyOrder() on 4 market instances. The returned values match the expected values.

- ***Test the getOpenOrders() method***

I tested the method by using the getOpenOrders() on market1. The returned values match the expected values.

Note: Because 4 markets work in the same way, the following method is only tested for market1.

- ***Test the getCurrentPrice() method***

I test the method by using the getCurrentPrice() method on market1. The returned values match the expected values.

- ***Test the isOpen() method***

I open 3 more markets.

```

Market market5 = new Market ('S', null, null);
Market market6 = new Market ('T', buyList, null);
Market market7 = new Market ('W', null, sellNode);

```

From here, I tested the isOpen() method on all markets. The returned values match the expected values.

- ***Test the isValidOrder() method***

I tested the method by testing it on all 4 markets. The returned values match the expected values.

Note: Because 4 markets work in the same way, the following methods are only tested for market1.

- ***Test the addOrder() method***

I create 4 more orders that have the same stock symbol as market1.

```
Order order1 = new BuyLimitOrder ('M', 123, 68, true, true, trader4);
```

```
Order order2 = new SellLimitOrder ('M', 214, 45, true, true, trader2);
```

```
Order order3 = new BuyMarketOrder ('M', 324, 53, trader3);
```

```
Order order4 = new SellMarketOrder ('M', 6546, 134, trader4);
```

I tested the method by adding order1, order2, order3, order4 to the market. Then I called the node using getElement() and getNext(). The returned values match the expected values.

- ***Test the removeOrder() method***

I create 4 more orders that have the same stock symbol as market1.

```
Order order1 = new BuyLimitOrder ('M', 123, 68, true, true, trader4);
```

```
Order order2 = new SellLimitOrder ('M', 214, 45, true, true, trader2);
```

```
Order order3 = new BuyMarketOrder ('M', 324, 53, trader3);
```

```
Order order4 = new SellMarketOrder ('M', 6546, 134, trader4);
```

I tested the method by removing order1, order2, order3, order4 to the market. Then I called the node using getElement() and getNext(). The returned values match the expected values.

- ***Test the matchingOrder() method***

I tested the method by comparing buyList, buyNode, sellNode, sellList to each other. The returned values match the expected values.

- ***Test the placeOrder() method***

I test the method by creating more input order and 1 LLNode<Transaction> that is expected output.

```
Order inputOrder1 = new BuyLimitOrder ('M', 123, 68, true, true, trader4);
```

```
Order inputOrder2 = new SellLimitOrder('M', 100, 100, true, true, trader3);
```

```
LLNode<Transaction> transaction = new LLNode<Transaction> (new Transaction('M', 100, 100, trader1, trader3, 0), null);
```

I then test the NoSuchElementException, if there's no match, and if there's a match. The returned values match the expected values.

- ***Test the toString() method***

I format the output string by calling all values of each instance. The returned values match the expected values.

- ***Test the equals() method***

I tested if there's identical orders. The returned values match the expected values.