```java
/**
 * @author Harley Phung
 * Testing using junit to test the Trader hierachy
 */
import org.junit.*;
import static org.junit.Assert.*;
import java.util.NoSuchElementException;

public class TraderHierachyTester {
    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
     */
    @Test
    public void testGetName() {
        //Trader
        assertEquals("Harley", trader1.getName());
        assertEquals("Harold", trader2.getName());
        assertEquals("Harley", trader3.getName());
        //Market maker
        assertEquals("Jason", marketMaker1.getName());
        assertEquals("Harold", marketMaker2.getName());
    }

    /**
     * Test the setName() method
     */
    @Test
    public void testSetname() {

        //Trader
        trader1.setName("Harley Phung");
        assertEquals("Harley Phung", trader1.getName());
        trader2.setName("Harold Connamacher");
        assertEquals("Harold Connamacher", trader2.getName());
        trader3.setName("Harley Phung");
        assertEquals("Harley Phung", trader3.getName());
        //Market maker
        marketMaker1.setName("Jason Hans");
        assertEquals("Jason Hans", marketMaker1.getName());
        marketMaker2.setName("Harold Connamacher");
        assertEquals("Harold Connamacher", marketMaker2.getName());
    }

    /**
     * Test the getDefaultOrderSize() method
     */
    @Test
    public void testGetDefaultOrderSize() {
        assertEquals(1000, marketMaker1.getDefaultOrderSize());
        assertEquals(500, marketMaker2.getDefaultOrderSize());
    }

    /**
```

```
     * Test the setDefaultOrderSize() method
     */
    @Test
    public void testSetDefaultOrderSize() {
        marketMaker1.setDefaultOrderSize(1500);
        assertEquals(1500, marketMaker1.getDefaultOrderSize());
        marketMaker2.setDefaultOrderSize(750);
        assertEquals(750, marketMaker2.getDefaultOrderSize());
    }

    /**
     * Test the getPriceOffset() method
     */
    @Test
    public void testGetPriceOffset() {
        assertEquals(100.0, 100.0, marketMaker1.getPriceOffset());
        assertEquals(20.0, 20.0, marketMaker2.getPriceOffset());
    }

    /**
     * Test the setPriceOffset() method
     */
    @Test
    public void testSetPriceOffset() {
        marketMaker1.setPriceOffset(125.0);
        assertEquals(125.0, 125.0, marketMaker1.getPriceOffset());
        marketMaker2.setPriceOffset(50.0);
        assertEquals(50.0, 50.0, marketMaker2.getPriceOffset());
    }

    /**
     * Test the toString() method
     */
    @Test
    public void testToString() {
        //Trader
        assertEquals("the name of the trader: Harley", trader1.toString());
        assertEquals("the name of the trader: Harold", trader2.toString());
        assertEquals("the name of the trader: Harley", trader3.toString());
        //Market maker
        assertEquals("The market maker name is: Jason", marketMaker1.toString());
        assertEquals("The market maker name is: Harold", marketMaker2.toString());
    }

    /**
     * Test the equals() method
     */
    @Test
    public void testEquals() {
        //Trader
        assertTrue(trader1.equals(trader3));
        assertFalse(trader1.equals(trader2));
        assertFalse(trader2.equals(trader3));
        //Market maker
        assertFalse(marketMaker1.equals(marketMaker2));
    }
}
```