```java
/**
 * A Test class for {@link MarketPlace}.
 *
 * @author Silas Agnew
 * @version December 4, 2017
 */

import java.util.Random;

public class MarketTest
{
    // Test constants
    private static final int    NUM_CASHIERS = 4;
    private static final double AVG_ARRIVAL  = 3.0;
    private static final double AVG_SERVICE   = 5.0;

    public static void main(String[] args)
    {
        Random rand = new Random();

        System.out.println("-Starting Test-");

        MarketPlace store = new MarketPlace();
        store.setParameters(NUM_CASHIERS, AVG_SERVICE, AVG_ARRIVAL, true);

        // Check parameters
        assert (store.getNumCashiers() == NUM_CASHIERS) :
                "Cashier count not set. Expected: "
                + NUM_CASHIERS + " Got: " + store.getNumCashiers();
        assert (Math.abs(store.getServiceTime() - AVG_SERVICE) < .001) :
                "Service time not set. Expected: "
                + AVG_SERVICE + " Got: " + store.getServiceTime();
        assert (Math.abs(store.getArrivalTime() - AVG_ARRIVAL) < .001) :
                "Arrival time not set. Expected: "
                + AVG_ARRIVAL + " Got: " + store.getArrivalTime();

        // Time method
        String time = store.formatTime(542);
        assert (time.equals("9:02am")) :
                "Time Format incorrect. Expected: 9:02am, Got: " + time;

        time  = store.formatTime(720);
        assert (time.equals("12:00pm")) :
                "Time Format incorrect. Expected: 12:00pm, Got: " + time;

        // Run test simulations
        for (int i = 0; i < 3; i++)
        {
            double arrival = rand.nextDouble() * 3.5 + 3;

            store.setParameters(NUM_CASHIERS, AVG_SERVICE, arrival, true);
            store.startSimulation();
            System.out.println(store.getReport());
            int before = store.getNumCustomersServed();

            arrival = rand.nextDouble() * 3.5;
```

```
57
58              store.setParameters(NUM_CASHIERS, AVG_SERVICE, arrival, true);
59              store.startSimulation();
60              System.out.println(store.getReport());
61              int after = store.getNumCustomersServed();
62
63              assert (before < after) :
64                      "Smaller Arrival time should produce more customers" +
65                              "\nBefore: " + before + " After: " + after;
66          }
67
68          System.out.println("-Test Finished-");
69      }
70  }
71
```