```java
package MULTISET_Testing;
import MULTISET.*;

/**
 * This program is a test harness for the MULTISET package.
 *
 * @author Matt Laidman
 * @version 1.0 (March 2014)
 */
public class TestHarness {


    @SuppressWarnings("unchecked")
    public TestHarness() {
        MultiSet data = new MyBag(new KeyedChar[] {new KeyedChar('a'), new KeyedChar('c'), new KeyedChar('b'), new KeyedChar('d'), new KeyedChar('c')});
        MultiSet data2 = new MySet(new KeyedChar[] {new KeyedChar('b'), new KeyedChar('e'), new KeyedChar('f'), new KeyedChar('a'), new KeyedChar('f')});
        MultiSet data3 = new MySet(new KeyedInt[] {new KeyedInt(2), new KeyedInt(5), new KeyedInt(1),
        MultiSet data4 = new MySet();
        MultiSet data5 = new MyBag();
        data.add(new KeyedChar('f'));
        data.add(new KeyedChar('e'));
        data.add(new KeyedChar('b'));
        data3.add(new KeyedInt(4));

        Iterator multiIterator = data.iterator();
        System.out.print("Data 1 (MyBag): ");
        while (multiIterator.hasNext()) {
            Keyed value = multiIterator.next();
            System.out.print(value.getKey());
        }
        System.out.println();
        multiIterator = data2.iterator();
        System.out.print("Data 2 (MySet): ");
        while (multiIterator.hasNext()) {
            Keyed value = multiIterator.next();
            System.out.print(value.getKey());
        }
        System.out.println();
        multiIterator = data3.iterator();
        System.out.print("Data 3 (MySet): ");
        while (multiIterator.hasNext()) {
            Keyed value = multiIterator.next();
            System.out.print(value.getKey());
        }
        System.out.println();
        multiIterator = data4.iterator();
        System.out.print("Data 4 (MySet): ");
        while (multiIterator.hasNext()) {
            Keyed value = multiIterator.next();
            System.out.print(value.getKey());
        }
        System.out.println();
        multiIterator = data5.iterator();
        System.out.print("Data 5 (MyBag): ");
        while (multiIterator.hasNext()) {
            Keyed value = multiIterator.next();
            System.out.print(value.getKey());
        }
        System.out.println("\n");

        System.out.print("Data 1 Cardinality: ");
        System.out.println(data.cardinality() + "\n");

        System.out.print("Data 1 Multiplicity of \'b\': ");
        System.out.println(data.multiplicity(new KeyedChar('b')) + "\n");

        multiIterator = data.union(data).iterator();
        System.out.print("Data 1 Union Data 1: ");
        while (multiIterator.hasNext()) {
            Keyed value = multiIterator.next();
            System.out.print(value.getKey());
        }
        System.out.println("\n");

        multiIterator = data3.union(data3).iterator();
```

```
 76          System.out.print("Data 3 Union Data 3: ");
 77          while (multiIterator.hasNext()) {
 78              Keyed value = multiIterator.next();
 79              System.out.print(value.getKey());
 80          }
 81          System.out.println("\n");
 82
 83          multiIterator = data.union(data2).iterator();
 84          System.out.print("Data 1 Union Data 2: ");
 85          while (multiIterator.hasNext()) {
 86              Keyed value = multiIterator.next();
 87              System.out.print(value.getKey());
 88          }
 89          System.out.println("\n");
 90
 91          multiIterator = data2.union(data).iterator();
 92          System.out.print("Data 2 Union Data 1: ");
 93          while (multiIterator.hasNext()) {
 94              Keyed value = multiIterator.next();
 95              System.out.print(value.getKey());
 96          }
 97          System.out.println("\n");
 98
 99          multiIterator = data.intersection(data2).iterator();
100          System.out.print("Data 1 Intersection Data 2: ");
101          while (multiIterator.hasNext()) {
102              Keyed value = multiIterator.next();
103              System.out.print(value.getKey());
104          }
105          System.out.println("\n");
106
107          multiIterator = data2.intersection(data).iterator();
108          System.out.print("Data 2 Intersection Data 1: ");
109          while (multiIterator.hasNext()) {
110              Keyed value = multiIterator.next();
111              System.out.print(value.getKey());
112          }
113          System.out.println("\n");
114
115          multiIterator = data3.intersection(data3).iterator();
116          System.out.print("Data 3 Intersection Data 3: ");
117          while (multiIterator.hasNext()) {
118              Keyed value = multiIterator.next();
119              System.out.print(value.getKey());
120          }
121          System.out.println("\n");
122
123          System.out.print("Data 2 Equal Data 1: " + data2.equal(data));
124          System.out.println("\n");
125
126          System.out.print("Data 2 Equal Data 2: " + data2.equal(data2));
127          System.out.println("\n");
128
129          System.out.print("Data 3 Is Empty: " + data3.isEmpty());
130          System.out.println("\n");
131
132          System.out.print("Data 4 Is Empty: " + data4.isEmpty());
133          System.out.println("\n");
134
135          System.out.print("Data 5 Is Empty: " + data5.isEmpty());
136          System.out.println("\n");
137      }
138
139
140      public static void main(String[] args) {
141          new TestHarness();
142      }
143  }
```