Hashmap

```
import java.util.Map;
import java.util.HashMap;
import java.util.Map.Entry;

class Student {

   public static Map<String, Double> findMaxMinScorers(Map<String, Double> studentMarks)
{
      Map<String, Double> result = new HashMap<>();

      if (studentMarks == null || studentMarks.isEmpty()) {
         return result;  // Return an empty map if input is null or empty
      }

      // Initialize variables to store max and min marks
      double maxMarks = Double.MIN_VALUE;
      double minMarks = Double.MAX_VALUE;

      // First, find the max and min marks
      for (double marks : studentMarks.values()) {
         if (marks > maxMarks) {
            maxMarks = marks;
         }
         if (marks < minMarks) {
            minMarks = marks;
         }
      }
```

```
        // Add students with max marks to the result
        for (Entry<String, Double> entry : studentMarks.entrySet()) {
            if (entry.getValue() == maxMarks) {
                result.put(entry.getKey() + " (Max)", entry.getValue());
            }
        }

        // Add students with min marks to the result
        for (Entry<String, Double> entry : studentMarks.entrySet()) {
            if (entry.getValue() == minMarks) {
                result.put(entry.getKey() + " (Min)", entry.getValue());
            }
        }

        return result;
    }

}

class Tester {

    public static void main(String args[]) {
        Map<String, Double> studentMarks = new HashMap<>();
        studentMarks.put("Lily", 90.0);
        studentMarks.put("Robin", 68.0);
        studentMarks.put("Marshall", 76.5);
```

```
        studentMarks.put("Neil", 67.0);
        studentMarks.put("Ted", 92.0);

        Map<String, Double> maxMinScorers = Student.findMaxMinScorers(studentMarks);

        System.out.println("Details of Top Scorers & Low
Scorers\n==================================");
        for (Entry<String, Double> entry : maxMinScorers.entrySet()) {
            System.out.println(entry.getKey() + " -- " + entry.getValue());
        }

    }
}
```

Output