

# MAPS – STUDENT GRADE MAP (PART 2)

**Lab Description :** Reimplement Part 1 so that the keys of the map are objects of class Student. A student should have a first name, a last name, and a unique integer ID. For grade changes and removals, lookup should be by ID. The printout should be sorted by last name. If two students have the same last name, then use the first name as a tie. If the first names are also identical, then use the integer ID breaker – you will need to implement Comparable for this. *Hint:* Use two maps.

**Lab Directions :** First, create a Student Class to satisfy the requirements detailed above. Then, adjust the following methods from your implementation completed in Part 1:

```
/**
 Prints the options menu and returns the letter that the user types does not check for
 invalid selection
 @return the next token on the input stream
 */
public static String printMenuAndGetChoice()

/**
 Prints the students and grades
 @param gradeMap the map to print
 */
public static void printGrades(Map<String, String> gradeMap)

/**
 Modifies an entry based on user input. Prints an error if an invalid student is
 modified
 @param gradeMap the map to modify
 @param idToStudentMap the map to associate student id with a student
 */
public static void modifyStudent(Map<Student, String> gradeMap,
 Map<Integer, Student> idToStudentMap)

/**
 Removes a student from the map based on user input
 @param gradeMap the map to remove the student from
 @param idToStudentMap the map to associate student id with a student
 */
public static void removeStudent(Map<Student, String> gradeMap,
 Map<Integer, Student> idToStudentMap)

/**
 Adds a student based on user input. Prints an error if a student
 is added that already exists in the map.
 @param idToGradeMap the map to add the student to
 @param idToStudentMap the map to associate student id with a student
 */
public static void addStudent(Map<Student, String> idToGradeMap,
 Map<Integer, Student> idToStudentMap)
```

**Extension :** Supply compatible hashCode and equals methods to the Student class. Test the hash code by adding Student objects to a hash set.