|  |  |  |
| --- | --- | --- |
| **LAB211 Assignment** | **Type:** | **Short Assignment** |
| **Code:** | **J1.S.P0068** |
| **LOC:** | **37** |
| **Slot(s):** | **1** |

**Title**

Input, sort and display student information.

**Background**

N/A

**Program Specifications**

Write a program that allows user to enter student information (Name, Class, Mark), then use collection to sort students name from A to Z, display student information on screen.

***Function details:***

**Function 1:** Display GUI And Input Data.

* User runs program. Program prompts user to input data.
* Auto next **Function** **2**.

**Function 2:** Perform function

* Use collection to sort name of student from A to Z
* Display the student information on screen
* Exit the program.

***Expectation of User interface:***

====== Collection Sort Program ======

Please input student information

Name: Nghia

Classes: FU1

Mark: 100

Do you want to enter more student information?(Y/N):Y

Please input student information

Name: Lien

Classes: FU1

Mark: 100

Do you want to enter more student information?(Y/N):N

-------------Student 1-------------

Name: Lien

Classes: FU1

Mark: 100.0

-------------Student 2-------------

Name: Nghia

Classes: FU1

Mark: 100.0

2

1

**Guidelines**

**Student must implement methods**

* sortStudent
* display

**in startup code.**

**Hint:**

Create a class called Student and has the following attributes:

private String name;

private float mark;

private String classes;

+ Create the constructors:

* Default constructor
* Constructor with parameters

+ Create the getter/setter for the attributes of Student class

Create Student Comparator class that implements Comparator interface, overrides the **compare** method

+ Use Collection.sort() to sort Student by name from A - Z

Create Main class with the main() method to test the above class, implement as below:

* Require user to enter student information.
* If the mark of student is not a valid number, request user to re-enter.
* Call sortStudent() function to sort the list of students.
* Call display() function to display list of students.

**Function 1:** Enter student information

* Program requires user to enter student name, address, and mark.
* Method name: List <Student> sortStudent (List <Student> students)
  + Input:
* students: list of student being entered
  + Return: list of students that was sorted.

**Function 2:** Display student information

* Program display all student information entered.
* Method name: void display (List <Student> students)
  + Input:
* students: list of students that was sorted.
  + Return: void.