

Seating Chart Generator

Monsi Magal and Gauri Iyer
Block 1

Description of Project

- Seating chart generator
 - “Random” chart generator isn’t random enough
- Students and teachers tend to get bothered when the same students keep ending up together
- Teachers input a text file of just names and set up the program by selecting how many rows and columns they want.
 - As of right now, our code only works with rows/columns, later implementations could include groups.

Code Breakdown

Class: Tester

- main
- loadDataFromFile
- generateChart
 - convertToStudents
 - removeFromAllStudents
 - inArrayList
 - findNextStudent
 - displayClassroom
- generateChartAgain

Class: Student

- constructor
- getName
- getPossibleStudents
- getPossibleStudents1
- removeStudent
- removeStudentFromWithout
- toString
- setEqualArrays

Some things to understand

- Each Student has an ArrayList of Strings of who they can sit next to.
 - It is first initialized with everyone in StudentList (minus themselves)
 - Each time a Student was seated in the classroom, they would be removed from everyone's ArrayList
- By the time the classroom has been filled, everyone's ArrayList is empty.
- If the inputted number of seats is greater than the remaining seats, those seats are filled with null.

The Problem

- We have two methods that create the classroom, generateChart and generateChartAgain
- generateChartAgain loops through the classroom generated previously and gets the students out.
- There's two problems with this.
 - 1. At the end when there's empty seats, the null that fills those seats also gets added.
 - 2. Each student we got out had an EMPTY ArrayList of possibleStudents.

Potential Solutions

- One possible solution is while we loop through the classroom in generateChartAgain to create an if statement.

```
if (classroom[r][c] != null) {  
    //add them to the array list  
}
```

An attempted solution

- Have two ArrayLists in Student
 - One where we remove who they've sat next to and also who has already been seated (called `studentListWithRemoval`).
 - One where we only remove who they've sat next to (called `studentListWithout`).
- At the end of the current generation, we tried to set the first array the same as the second.

If we had more time

- Our code was supposedly complete
 - If we had more time we would've removed the bugs entirely
- Graphics
 - easy to understand for teachers
- Groups instead of row / column
 - “5 groups of four & 2 groups of three” instead of “4 rows & 3 cols”
- Special restrictions
 - a student can't sit in front
 - a student cannot sit next to another person
 - a student must sit in a specified position