JavaScript is disabled on your browser.

[Skip navigation links](#gjdgxs)

* [Package](http://docs.google.com/package-summary.html)
* Class
* [Use](http://docs.google.com/class-use/SeatingProblem.html)
* [Tree](http://docs.google.com/package-tree.html)
* [Deprecated](http://docs.google.com/deprecated-list.html)
* [Index](http://docs.google.com/index-files/index-1.html)
* [Help](http://docs.google.com/help-doc.html)
* [All Classes](http://docs.google.com/allclasses.html)
* SEARCH:

JavaScript is disabled on your browser.

* Summary:
* Nested |
* Field |
* [Constr](#30j0zll) |
* [Method](#1fob9te)
* Detail:
* Field |
* [Constr](#3znysh7) |
* [Method](#2et92p0)

Class SeatingProblem

* java.lang.Object
  + SeatingProblem
* public class SeatingProblem  
  extends java.lang.Object  
  Contains all information about the problem, in a format easy for processingAuthor: sunny

### Constructor SummaryConstructors

|  |  |
| --- | --- |
| * + Constructor | * + Description |
| * + SeatingProblem​(java.util.ArrayList<[Student](http://docs.google.com/Student.html)> students, int maxStudentsPerTable) | * + Creates a SeatingProblem instance |

### Method SummaryAll Methods Instance Methods Concrete Methods

|  |  |  |
| --- | --- | --- |
| * + Modifier and Type | * + Method | * + Description |
| * + int | * + [getMaxStudentsPerTable](#tyjcwt)() |  |
| * + int | * + [getPeopleWithPref](#3dy6vkm)() |  |
| * + int | * + [getPrefCount](#1t3h5sf)() |  |
| * + int | * + [getPrefCount](#4d34og8)​(int student) | * + gets number of preferences a certain student has |
| * + java.util.ArrayList<java.lang.Integer> | * + [getPrefs](#2s8eyo1)​(int student) | * + gets students' preferences |
| * + java.util.Random | * + [getRandomSource](#17dp8vu)() | * + should be the only source of randomness used by solver components |
| * + [Student](http://docs.google.com/Student.html) | * + [getStudentByID](#3rdcrjn)​(int id) | * + returns a student object given their id |
| * + int | * + [getStudentCount](#26in1rg)() |  |
| * + boolean | * + [hasPref](#lnxbz9)​(int studentA, int studentB) | * + Checks if a preference exists |

### Methods inherited from class java.lang.Objectequals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

### Constructor Detail

#### SeatingProblem public SeatingProblem​(java.util.ArrayList<[Student](http://docs.google.com/Student.html)> students, int maxStudentsPerTable) Creates a SeatingProblem instanceParameters: students - the students in the problem maxStudentsPerTable - max number of students allowed at one table

### Method Detail

#### getPeopleWithPref public int getPeopleWithPref()Returns: num people with preference

#### getMaxStudentsPerTable public int getMaxStudentsPerTable()Returns: the maximum number of students allowed at one table

#### getStudentCount public int getStudentCount()Returns: the number of students in the problem

#### getPrefCount public int getPrefCount()Returns: the total number of preferences in this problem (number of edges)

#### getPrefCount public int getPrefCount​(int student) gets number of preferences a certain student hasParameters: student - the id of the student Returns: the num of preferences that guy has

#### hasPref public boolean hasPref​(int studentA, int studentB) Checks if a preference existsParameters: studentA - the student studentB - the other student Returns: true if studentA has a preference for studentB

#### getStudentByID public [Student](http://docs.google.com/Student.html) getStudentByID​(int id) returns a student object given their idParameters: id - the id of student Returns: the student object

#### getPrefs public java.util.ArrayList<java.lang.Integer> getPrefs​(int student) gets students' preferencesParameters: student - the student to get preferences from Returns: the list of preferences of that student

#### getRandomSource public java.util.Random getRandomSource() should be the only source of randomness used by solver componentsReturns: a random number generator

[Skip navigation links](#35nkun2)

* [Package](http://docs.google.com/package-summary.html)
* Class
* [Use](http://docs.google.com/class-use/SeatingProblem.html)
* [Tree](http://docs.google.com/package-tree.html)
* [Deprecated](http://docs.google.com/deprecated-list.html)
* [Index](http://docs.google.com/index-files/index-1.html)
* [Help](http://docs.google.com/help-doc.html)
* [All Classes](http://docs.google.com/allclasses.html)

JavaScript is disabled on your browser.

* Summary:
* Nested |
* Field |
* [Constr](#30j0zll) |
* [Method](#1fob9te)
* Detail:
* Field |
* [Constr](#3znysh7) |
* [Method](#2et92p0)