

Class Projects

java.lang.Object  
Projects

public class Projects  
extends java.lang.Object

Constructor Summary

Constructors

Constructor and Description

Projects()

Method Summary

All MethodsInstance MethodsConcrete Methods

Modifier and Type	Method and Description
Project	<b>addContributorToProject</b> (User user, Project project, java.lang.String contributorUsername) Adds a contributor to a project
boolean	<b>addProject</b> (User user, java.lang.String projectName, java.lang.String abstractText, java.lang.String ownerUsername) Adds a project to the database if the user trying is of type Faculty and has logged in.
Project	<b>deleteContributorFromProject</b> (User user, Project project, java.lang.String contributorUsername) Removes a contributor from a project
boolean	<b>deleteProject</b> (User user, Project project) Deletes a project
Project	<b>editProjectAbstract</b> (User user, Project project, java.lang.String newAbstractText) Edits a project abstract text
Project	<b>editProjectName</b> (User user, Project project, java.lang.String newName) Edits a project name
Project	<b>editProjectOwner</b> (User user, Project project, java.lang.String newUsername) Edits a project's owner
java.util.ArrayList<java.lang.String>	<b>findKeywordsByProjectID</b> (int projID) Finds all keywords associated with a given project ID number
Project	<b>findProjectByID</b> (java.lang.Integer projID) Returns a single project given the primary key projID
java.util.ArrayList<Project>	<b>findProjectsByContributor</b> (java.lang.Integer contributorUserID) finds any projects in the database by a given contributor
java.util.ArrayList<Project>	<b>findProjectsByKeyword</b> (java.lang.String keyword) finds all projects in the database that have a given keyword
java.util.ArrayList<Project>	<b>findProjectsByName</b> (java.lang.String projectName) find all projects in database by a given name
java.util.ArrayList<Project>	<b>findProjectsByOwner</b> (java.lang.Integer ownerID) Finds any projects in the database by the owner ID
private java.util.ArrayList<java.util.ArrayList<java.lang.String>>	<b>resultSetFormatter</b> (java.sql.ResultSet rs) Formats a given result set into a 2D array list

Methods inherited from class java.lang.Object

clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Detail

**Projects**

```
public Projects()
```

**Method Detail****resultSetFormatter**

```
private java.util.ArrayList<java.util.ArrayList<java.lang.String>> resultSetFormatter(java.sql.ResultSet rs)
```

Formats a given result set into a 2D array list

**Parameters:**

rs - the result set to format

**Returns:**

the 2D array list that has only strings

**findProjectsByName**

```
public java.util.ArrayList<Project> findProjectsByName(java.lang.String projectName)
```

find all projects in database by a given name

**Parameters:**

projectName - the name of a project

**Returns:**

a list of projects with a given name

**findProjectsByOwner**

```
public java.util.ArrayList<Project> findProjectsByOwner(java.lang.Integer ownerId)
```

Finds any projects in the database by the owner ID

**Parameters:**

ownerID - the user ID of the owner to search for owned projects

**Returns:**

a list of projects owned by a given owner userID, or null if no results

**findProjectByID**

```
public Project findProjectByID(java.lang.Integer projID)
```

Returns a single project given the primary key projID

**Parameters:**

projID - the unique key that links to a single project in the database

**Returns:**

a single project object or null if no results from db

**findProjectsByContributor**

```
public java.util.ArrayList<Project> findProjectsByContributor(java.lang.Integer contributorUserID)
```

finds any projects in the database by a given contributor

**Parameters:**

contributorUserID - the user ID to get associated projects from

**Returns:**

a list of projects

**findProjectsByKeyword**

```
public java.util.ArrayList<Project> findProjectsByKeyword(java.lang.String keyword)
```

finds all projects in the database that have a given keyword

**Parameters:**

keyword - the keyword to get all associated projects with

**Returns:**

a list of projects or null if no results

#### findKeywordsByProjectID

```
public java.util.ArrayList<java.lang.String> findKeywordsByProjectID(int projID)
```

Finds all keywords associated with a given project ID number

**Parameters:**

projID - the projID of a project

**Returns:**

a list of keywords

#### addProject

```
public boolean addProject(User user,  
                           java.lang.String projectName,  
                           java.lang.String abstractText,  
                           java.lang.String ownerUsername)
```

Adds a project to the database if the user trying is of type Faculty and has logged in.

**Parameters:**

user - the user trying to add a project (must be of type "F" for faculty

projectName - the name of the project

abstractText - the content of the project

ownerUsername - the owner's unique username

**Returns:**

true if the add was successful, false otherwise

#### editProjectName

```
public Project editProjectName(User user,  
                                Project project,  
                                java.lang.String newName)
```

Edits a project name

**Parameters:**

user - the user trying to change a project

project - the project to change

newName - the new name

**Returns:**

the updated project

#### editProjectAbstract

```
public Project editProjectAbstract(User user,  
                                    Project project,  
                                    java.lang.String newAbstractText)
```

Edits a project abstract text

**Parameters:**

user - the user trying to change a project

project - the project to change

newAbstractText - the new abstract text

**Returns:**

the updated project

#### editProjectOwner

```
public Project editProjectOwner(User user,  
                                 Project project,  
                                 java.lang.String newUsername)
```

Edits a project's owner

**Parameters:**

user - the user trying to change a project

project - the project to change

newUsername - the new user to change to

**Returns:**

the updated project

### deleteProject

```
public boolean deleteProject(User user,  
                             Project project)
```

Deletes a project

**Parameters:**

user - the user trying to change a project

project - the project to change

**Returns:**

true if successful, false otherwise

### addContributorToProject

```
public Project addContributorToProject(User user,  
                                       Project project,  
                                       java.lang.String contributorUsername)
```

Adds a contributor to a project

**Parameters:**

user - the user trying to add a contributor to their project

project - the project to add a contributor to

contributorUsername - the username of the contributor

**Returns:**

the updated project

### deleteContributorFromProject

```
public Project deleteContributorFromProject(User user,  
                                             Project project,  
                                             java.lang.String contributorUsername)
```

Removes a contributor from a project

**Parameters:**

user - the user trying to remove a contributor from their project

project - the project to remove a contributor from

contributorUsername - the username of the contributor

**Returns:**

the updated project