Assignment 3

Generated by Doxygen 1.8.13

Contents

Chapter 1

Namespace Index

1.1 Packages

Here are the packages with brief descriptions (if available):

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

Chapter 3

Class Index

3.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

Chapter 4

File Index

4.1 File List

Here is a list of all documented files with brief descriptions:

Chapter 5

Namespace Documentation

5.1 Package src

Classes

- class AttributeT
- class CourseT

  Class for initializing and operating on CourseT objects.
- enum IndicatorT
- class LOsT
- interface **Measures**
- class Norm
- class ProgramT
- class Services

5.1.1 Detailed Description

Author: Mohammad Omar Zahir - zahirm1 Revised: March 29 2021

Description: File for the IndicatorT enumerator

Author: Mohammad Omar Zahir Revised: 29 March, 2021

Description: Contains the Abstract Data Type for creating and operating with AttributeT objects

Author: Mohammad Omar Zahir - zahirm1 Revised: March 29 2021

Description: Contains the interface for Measures that will be implemented and overriden by other classes

Author: Mohammad Omar Zahir - zahirm1 Revised: March 29 2021

Description: File for the services module that contains a functions to determine the normal

Author: Mohammad Omar Zahir Revised: 29 March, 2021

Description: File for the Norm Abstract object that specifies various boolean properties for other methods

Author: Mohammad Omar Zahir - zahirm1 Revised: March 29 2021

Description: Contains the Abstract Data Type for creating and operating with LOsT objects

Author: Mohammad Omar Zahir Revised: 29 March, 2021

Description: Contains the Abstract Data Type for creating and operating with ProgramT objects on the object

Chapter 6

Class Documentation

6.1 src.AttributeT Class Reference

Public Member Functions

- **AttributeT** (String attribName, IndicatorT[ ] indicators)
- String **getName** ()
- IndicatorT [ ] **getIndicators** ()

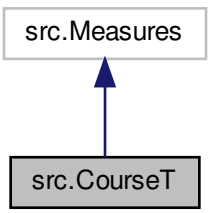
The documentation for this class was generated from the following file:

- src/AttributeT.java

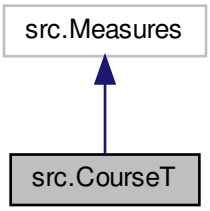
6.2 src.CourseT Class Reference

Class for initializing and operating on CourseT objects.

Inheritance diagram for src.CourseT:



Collaboration diagram for src.CourseT:



Public Member Functions

- CourseT (String courseName, IndicatorT[ ] indicators)

  Method for constructing the CourseT object.
- String getName ()

  Getter for returning the name of the CourseT object.
- IndicatorT [ ] getIndicators ()

  Getter for returning the array of IndicatorT objects given at the time of initialization.
- LOsT [ ] getLOs (IndicatorT indicator)

  Method for retrieving the LOsT objects that are mapped to a given IndicatorT object.
- void addLO (IndicatorT indicator, LOsT outcome)

  Method for adding/mapping a given LOsT object to a given IndicatorT object.
- void delLO (IndicatorT indicator, LOsT outcome)

  Method for removing a given LOsT object that is mapped to a given IndicatorT object.
- boolean member (IndicatorT indicator, LOsT[ ] outcomes)

  Method for checking if a given IndicatorT object exists with the given amount of LOsT objects mapped to it.
- double [ ] measures ()

  Implements the measure method from the Measure module that accepts no input.
- double [ ] measures (IndicatorT ind)

  Implements the measure method from the Measure module that accepts an IndicatorT object.
- double [ ] measures (AttributeT att)

  Implements the measure method from the Measure module that accepts an IndicatorT object.

6.2.1 Detailed Description

Class for initializing and operating on CourseT objects.

Consists of various functions, including the constructor, useful methods, getters, as well as the implementations of the measures methods from the Measures module.

6.2.2 Constructor & Destructor Documentation

6.2.2.1 CourseT()

```
src.CourseT.CourseT (
            String courseName,
            IndicatorT [] indicators )
```

Method for constructing the CourseT object.

Parameters

courseName	A String to represent the name.
indicators	An array of indicators to be stored in the course object.

The constructor makes use of a Hashmap that maps an empty array of LOs objects to each of the indicators as separate keys.

6.2.3 Member Function Documentation

6.2.3.1 addLO()

```
void src.CourseT.addLO (
            IndicatorT indicator,
            LOsT outcome )
```

Method for adding/mapping a given LOsT object to a given IndicatorT object.

Parameters

indicator	An IndicatorT object for mapping the LOs object to.
outcome	An LOsT object for mapping to the given IndicatorT object.

If the IndicatorT object is found, the LOs object is only added if another LOs object of the same name does not exist there. If the IndicatorT object does not exist, or the LOsT object already exists mapped to the IndicatorT object, nothing is done.

6.2.3.2 delLO()

```
void src.CourseT.delLO (
            IndicatorT indicator,
            LOsT outcome )
```

Method for removing a given LOsT object that is mapped to a given IndicatorT object.

Parameters

indicator	An IndicatorT object that has the LOs object mapped to it.
outcome	An LOsT object that is mapped to the given IndicatorT object.

If the IndicatorT object is found, the LOs object is only removed if an LOs object of the same name exists there. If the IndicatorT object does not exist, or the LOsT object of the same name does not exist mapped to the IndicatorT object, nothing is done.

6.2.3.3 getIndicators()

```
IndicatorT [] src.CourseT.getIndicators ( )
```

Getter for returning the array of IndicatorT objects given at the time of initialization.

The IndicatorT objects from the class's Hashmap.

Returns

Array of IndicatorT objects for the Course.

6.2.3.4 getLOs()

```
LOsT [] src.CourseT.getLOs (
            IndicatorT indicator )
```

Method for retrieving the LOsT objects that are mapped to a given IndicatorT object.

Parameters

indicator	An IndicatorT object for which the LOs objects that are mapped to it are to be returned.

If the IndicatorT object is found, the LOs objects that are mapped to it are returned. If the IndicatorT object does not exist, or there are no LOsT objects mapped to it nothing, an empty array of LOsT objects is returned.

Returns

Array of LOsT objects that correspond to the given IndicatorT object.

6.2.3.5 getName()

```
String src.CourseT.getName ( )
```

Getter for returning the name of the CourseT object.

Returns

The name of the course as a String.

6.2.3.6 measures() [1/3]

```
double [] src.CourseT.measures ( )
```

Implements the measure method from the Measure module that accepts no input.

This method is not supported in this class. If invoked, an UnsupportedOperationException will be thrown.

Exceptions

UnsupportedOperationException	

6.2.3.7 measures() [2/3]

```
double [] src.CourseT.measures (
            IndicatorT ind )
```

Implements the measure method from the Measure module that accepts an IndicatorT object.

Parameters

ind	An IndicatorT object.

Overrides the measure method in the Measure module.

Returns

The sum of all the LOs objects for a given IndicatorT object in the course, or the normal of this sum if that is specified by a boolean value of nInd in the Norm.

6.2.3.8 measures() [3/3]

```
double [] src.CourseT.measures (
            AttributeT att )
```

Implements the measure method from the Measure module that accepts an IndicatorT object.

Parameters

att	An AttributeT object.

Overrides the measure method in the Measure module.

Returns

The sum of all the LOs objects for all the IndicatorT objects that correspond to the AttributeT object provided, or the normal of this sum if that is specified by a boolean value of nAtt in the Norm.

6.2.3.9 member()

```
boolean src.CourseT.member (
            IndicatorT indicator,
            LOsT [] outcomes )
```

Method for checking if a given IndicatorT object exists with the given amount of LOsT objects mapped to it.

Parameters

indicator	An IndicatorT object that should have the LOs objects mapped to it.
outcomes	A array of LOsT objects that should be mapped to the given IndicatorT object.

Returns

A boolean indicating whether the given LOsT objects belong to the given IndicatorT object for this course.

The documentation for this class was generated from the following file:

- src/CourseT.java

6.3 src.IndicatorT Enum Reference

Public Attributes

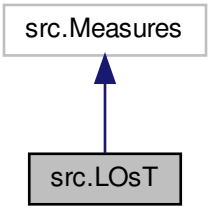
- **math**
- **specEngKnow**
- **assumpt**
- **suitableFund**
- **recogTheory**
- **modelSelect**
- **estOutcomes**
- **desProcess**
- **desPrinciples**
- **openEnded**
- **ideaGeneration**
- **healthSafety**
- **standards**
- **tools**
- **engInSoc**

The documentation for this enum was generated from the following file:

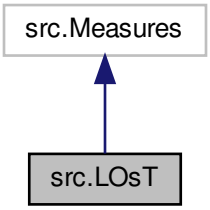
- src/IndicatorT.java

6.4 src.LOsT Class Reference

Inheritance diagram for src.LOsT:



Collaboration diagram for src.LOsT:



Public Member Functions

- **LOsT** (String topic, int nblw, int nmrg, int nmts, int nexc) throws IllegalArgumentException
- String **getName** ()
- boolean **equals** (Object o)
- double [ ] **measures** ()
- double [ ] **measures** (IndicatorT ind)
- double [ ] **measures** (AttributeT att)

The documentation for this class was generated from the following file:

- src/LOsT.java

6.5 src.Norm Class Reference

Static Public Member Functions

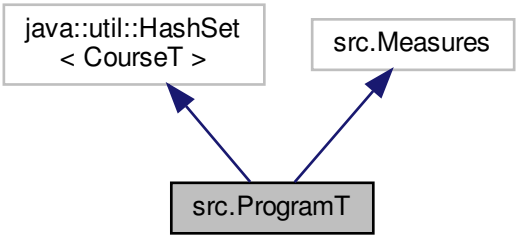
- static void **setNorms** (boolean nLOs, boolean nInd, boolean nAtt)
- static boolean **getNLOs** ()
- static boolean **getNInd** ()
- static boolean **getNAtt** ()
- static void **setNLOs** (boolean nLOs)
- static void **setNInd** (boolean nInd)
- static void **setNAtt** (boolean nAtt)

The documentation for this class was generated from the following file:

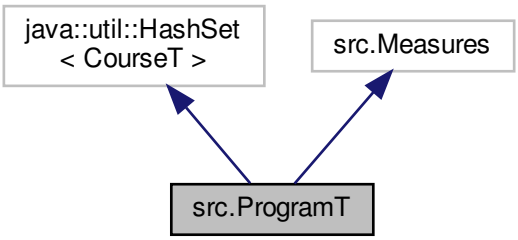
- src/Norm.java

6.6 src.ProgramT Class Reference

Inheritance diagram for src.ProgramT:



Collaboration diagram for src.ProgramT:



Public Member Functions

- double [ ] **measures** ()
- double [ ] **measures** (IndicatorT ind)
- double [ ] **measures** (AttributeT att)

The documentation for this class was generated from the following file:

- src/ProgramT.java

6.7 src.Services Class Reference

Static Public Member Functions

- static double [ ] **normal** (double[ ] v)

The documentation for this class was generated from the following file:

- src/Services.java

Chapter 7

File Documentation

7.1 src/CourseT.java File Reference

Contains the Abstract Data Type for creating and operating with CourseT objects.

Classes

- class src.CourseT

  Class for initializing and operating on CourseT objects.

Packages

- package src

7.1.1 Detailed Description

Contains the Abstract Data Type for creating and operating with CourseT objects.

Author

Mohammad Omar Zahir - zahirm1

Date

March 29, 2021

Index