Packages

Package	Description
Controller	
Model	
View	

Hierarchy For All Packages

Package Hierarchies:
Controller, Model, View

Class Hierarchy

- java.lang.Object
  - Controller.**Controller** (implements Controller.IController)
  - Model.**Dose** (implements Model.IDose, java.io.Serializable)
  - Model.**Medicine** (implements Model.IMedicine, java.io.Serializable)
  - View.**View**

Interface Hierarchy

- Controller.**IController**
- Model.**IDose**
- Model.**IMedicine**

Deprecated API

Contents

A C D G I L M N P R S T U V

A

addDose(boolean) - Static method in class View.View

Creates Dose from user input.

addDose(LocalDateTime, double, boolean) - Method in class Controller.Controller

addDose(LocalDateTime, double, boolean) - Method in interface Controller.IController

addDose(LocalDateTime, double, Boolean) - Method in interface Model.IMedicine

Add a Dose to doses array.

addDose(LocalDateTime, double, Boolean) - Method in class Model.Medicine

amountDose - Variable in class Model.Dose

Amount of dose.

A C D G I L M N P R S T U V

How This API Document Is Organized

This API (Application Programming Interface) document has pages corresponding to the items in the navigation bar, described as follows.

Overview

The Overview page is the front page of this API document and provides a list of all packages with a summary for each. This page can also contain an overall description of the set of packages.

Package

Each package has a page that contains a list of its classes and interfaces, with a summary for each. This page can contain six categories:

- Interfaces (italic)
- Classes
- Enums
- Exceptions
- Errors
- Annotation Types

Class/Interface

Each class, interface, nested class and nested interface has its own separate page. Each of these pages has three sections consisting of a class/interface description, summary tables, and detailed member descriptions:

- Class inheritance diagram
- Direct Subclasses
- All Known Subinterfaces
- All Known Implementing Classes
- Class/interface declaration
- Class/interface description
- Nested Class Summary
- Field Summary
- Constructor Summary
- Method Summary
- Field Detail
- Constructor Detail
- Method Detail

Each summary entry contains the first sentence from the detailed description for that item. The summary entries are alphabetical, while the detailed descriptions are in the order they appear in the source code. This preserves the logical groupings established by the programmer.

Annotation Type

Each annotation type has its own separate page with the following sections:

- Annotation Type declaration
- Annotation Type description
- Required Element Summary
- Optional Element Summary
- Element Detail

Enum

Each enum has its own separate page with the following sections:

- Enum declaration
- Enum description
- Enum Constant Summary
- Enum Constant Detail

Tree (Class Hierarchy)

There is a Class Hierarchy page for all packages, plus a hierarchy for each package. Each hierarchy page contains a list of classes and a list of interfaces. The classes are organized by inheritance structure starting with `java.lang.Object`. The interfaces do not inherit from `java.lang.Object`.

- When viewing the Overview page, clicking on "Tree" displays the hierarchy for all packages.
- When viewing a particular package, class or interface page, clicking "Tree" displays the hierarchy for only that package.

Deprecated API

The Deprecated API page lists all of the API that have been deprecated. A deprecated API is not recommended for use, generally due to improvements, and a replacement API is usually given. Deprecated APIs may be removed in future implementations.

Index

The Index contains an alphabetic list of all classes, interfaces, constructors, methods, and fields.

Prev/Next

These links take you to the next or previous class, interface, package, or related page.

Frames/No Frames

These links show and hide the HTML frames. All pages are available with or without frames.

All Classes

The All Classes link shows all classes and interfaces except non-static nested types.

Serialized Form

Each serializable or externalizable class has a description of its serialization fields and methods. This information is of interest to re-implementors, not to developers using the API. While there is no link in the navigation bar, you can get to this information by going to any serialized class and clicking "Serialized Form" in the "See also" section of the class description.

Constant Field Values

The Constant Field Values page lists the static final fields and their values.

This help file applies to API documentation generated using the standard doclet.

Packages

Package	Description
Controller	
Model	
View	

All Classes

Controller

Dose

IController

IDose

IMedicine

Medicine

View

Package Controller

Interface Summary

Interface	Description
IController	

Class Summary

Class	Description
Controller	

Package Model

Interface Summary

Interface	Description
IDose	
IMedicine	

Class Summary

Class	Description
Dose	
Medicine	

Package View

Class Summary

Class	Description
View	

Hierarchy For Package Controller

Package Hierarchies:

All Packages

Class Hierarchy

- java.lang.Object
  - Controller.**Controller** (implements Controller.IController)

Interface Hierarchy

- Controller.**IController**

Hierarchy For Package Model

Package Hierarchies:
All Packages

Class Hierarchy

- java.lang.Object
    - Model.**Dose** (implements Model.IDose, java.io.Serializable)
    - Model.**Medicine** (implements Model.IMedicine, java.io.Serializable)

Interface Hierarchy

- Model.**IDose**
- Model.**IMedicine**

Hierarchy For Package View

Package Hierarchies:

All Packages

Class Hierarchy

- java.lang.Object
  - View.**View**

Package Controller

Class Controller

java.lang.Object
    Controller.Controller

All Implemented Interfaces:

`IController`

```
public class Controller
extends java.lang.Object
implements IController
```

Field Summary

Fields

Modifier and Type	Field	Description
`private IMedicine`	`medicine`	Medicine file
`private java.time.ZoneOffset`	`timezone`	Defines time zone.

Constructor Summary

Constructors

Constructor	Description
`Controller()`	

Method Summary

All Methods **Static Methods** **Instance Methods** **Concrete Methods**

Modifier and Type	Method	Description
void	**addDose**(java.time.LocalDateTime dateTimeTakeDose, double amount, boolean isTest)	
java.time.LocalDate	**createLocalDate**(java.lang.String date)	Creates a LocalDate instance by parsing a String.
java.time.LocalTime	**createLocalTime**(java.lang.String time)	Creates a LocalTime instance by parsing a String.
java.lang.Double	**getConcentrationAtTime**(**IDose** doseIn, java.time.LocalDateTime dateTimeAtIn)	Calculate concentration amount at specified time.
IMedicine	**getMedicine**()	Retrieves the Medicine instance.
java.time.LocalDateTime	**getPeakLevel**(**IDose** dose)	Calculates when Dose is going to reach its peak level of concentration.
java.lang.Double	**getSumConcentrationsAtTime**(java.time.LocalDateTime dateTime)	Sums all dose amount at a time specified by the user
java.time.LocalDateTime	**getWhenToDose**(**IDose** dose, double amountDoseAt)	Calculates when to give next dose to reach a certain concentration amount.
void	**loadFile**(java.lang.String filename)	Load a saved Medicine file.
void	**newFile**(java.lang.String nameMedicine, java.time.LocalTime timeMaxMedicine, java.time.LocalTime timeHalfLifeMedicine)	Create a new Medicine instance.

`static java.lang.Double`	**parseValidAmount**()	Verifies if parsed input is a valid number amount.
`static java.lang.Integer`	**parseValidInt**()	Verifies if parsed input is a valid integer.
`void`	**removeAllDoses**()	Delete all doses from the dosages array.
`void`	**removeDose**(int index)	Remove a dose by its index.
`void`	**removeTestDoses**()	Removes all test doses.
`void`	**saveFile**(java.lang.String filename)	Save Medicine to a file.

Methods inherited from class java.lang.Object

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

Field Detail

timezone

private java.time.ZoneOffset timezone

Defines time zone. Default: -05:00

medicine

private IMedicine medicine

Medicine file

Constructor Detail

Controller

public Controller()

Method Detail

getMedicine

public IMedicine getMedicine()

Description copied from interface: `IController`

Retrieves the Medicine instance.

Specified by:

`getMedicine` in interface `IController`

Returns:

Medicine instance.

addDose

```
public void addDose(java.time.LocalDateTime dateTimeTakeDose,
                    double amount,
                    boolean isTest)
```

Specified by:

`addDose` in interface `IController`

removeDose

public void removeDose(int index)

Description copied from interface: `IController`

Remove a dose by its index.

Specified by:

`removeDose` in interface `IController`

Parameters:

`index` - index of dose.

removeAllDoses

```
public void removeAllDoses()
```

Description copied from interface: `IController`

Delete all doses from the dosages array.

Specified by:

`removeAllDoses` in interface `IController`

getConcentrationAtTime

```
public java.lang.Double getConcentrationAtTime(IDose doseIn,
                                               java.time.LocalDateTime dateTimeAtIn)
```

Description copied from interface: `IController`

Calculate concentration amount at specified time.

Specified by:

`getConcentrationAtTime` in interface `IController`

Parameters:

`doseIn` - Dose instance.

`dateTimeAtIn` - Specified time.

Returns:

Concentration amount at specific time.

getSumConcentrationsAtTime

```
public java.lang.Double getSumConcentrationsAtTime(java.time.LocalDateTime dateTime)
```

Description copied from interface: `IController`

Sums all dose amount at a time specified by the user

Specified by:

getSumConcentrationsAtTime in interface IController

Parameters:

dateTime - Specified time

Returns:

Sum of dose amounts at specified time.

saveFile

```
public void saveFile(java.lang.String filename)
```

Description copied from interface: IController

Save Medicine to a file. By default, save directory is user's home directory. Dose and Medicine MUST implement Serializable for this feature to work.

Specified by:

saveFile in interface IController

Parameters:

filename - Name of save file to be saved.

loadFile

```
public void loadFile(java.lang.String filename)
```

Description copied from interface: IController

Load a saved Medicine file. By default, load directory is user's home directory. Dose and Medicine MUST implement Serializable for this feature to work.

Specified by:

loadFile in interface IController

Parameters:

filename - Name of the file to be loaded.

newFile

```
public void newFile(java.lang.String nameMedicine,
                    java.time.LocalTime timeMaxMedicine,
                    java.time.LocalTime timeHalfLifeMedicine)
```

Description copied from interface: IController

Create a new Medicine instance.

Specified by:

newFile in interface IController

Parameters:

nameMedicine - Name of medicine

timeMaxMedicine - TMax of medicine.

timeHalfLifeMedicine - Half life of medicine.

createLocalDate

public java.time.LocalDate createLocalDate(java.lang.String date)

Description copied from interface: IController

Creates a LocalDate instance by parsing a String.

Specified by:

createLocalDate in interface IController

Parameters:

date - Date as a String.

Returns:

LocalDate instance.

createLocalTime

public java.time.LocalTime createLocalTime(java.lang.String time)

Description copied from interface: IController

Creates a LocalTime instance by parsing a String.

Specified by:

createLocalTime in interface IController

Parameters:

time - Time as a String.

Returns:

LocalTime instance.

parseValidAmount

public static java.lang.Double parseValidAmount()

Verifies if parsed input is a valid number amount.

Returns:

valid amount.

parseValidInt

public static java.lang.Integer parseValidInt()

Verifies if parsed input is a valid integer.

Returns:

valid integer.

removeTestDoses

public void removeTestDoses()

Description copied from interface: `IController`

Removes all test doses.

Specified by:

`removeTestDoses` in interface `IController`

getPeakLevel

public java.time.LocalDateTime getPeakLevel(IDose dose)

Description copied from interface: `IController`

Calculates when Dose is going to reach its peak level of concentration.

Specified by:

`getPeakLevel` in interface `IController`

Parameters:

`dose` - Dose instance.

Returns:

Time when Dose reaches its peak level.

getWhenToDose

```
public java.time.LocalDateTime getWhenToDose(IDose dose,
                                             double amountDoseAt)
```

Description copied from interface: `IController`

Calculates when to give next dose to reach a certain concentration amount.

Specified by:

`getWhenToDose` in interface `IController`

Parameters:

`dose` - Dose instance.

`amountDoseAt` - Desired dose concentration amount.

Returns:

Time when doses reaches specified amount.

OVERVIEW PACKAGE CLASS TREE DEPRECATED INDEX HELP

PREV CLASS **NEXT CLASS** FRAMES NO FRAMES ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

OVERVIEW PACKAGE CLASS TREE DEPRECATED INDEX HELP

PREV CLASS NEXT CLASS FRAMES NO FRAMES ALL CLASSES SEARCH: Search

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Package Controller

Interface IController

All Known Implementing Classes:

`Controller`

```
public interface IController
```

Method Summary

All Methods | **Instance Methods** | **Abstract Methods**

Modifier and Type	Method	Description
`void`	**`addDose`**(java.time.LocalDateTime dateTimeTakeDose, double amount, boolean isTest)	
`java.time.LocalDate`	**`createLocalDate`**(java.lang.String date)	Creates a LocalDate instance by parsing a String.
`java.time.LocalTime`	**`createLocalTime`**(java.lang.String time)	Creates a LocalTime instance by parsing a String.
`java.lang.Double`	**`getConcentrationAtTime`**(**`IDose`** doseIn, java.time.LocalDateTime dateTimeAtIn)	Calculate concentration amount at specified time.
`IMedicine`	**`getMedicine`**()	Retrieves the Medicine instance.
`java.time.LocalDateTime`	**`getPeakLevel`**(**`IDose`** dose)	Calculates

		when Dose is going to reach its peak level of concentration.
`java.lang.Double`	**getSumConcentrationsAtTime**(java.time.LocalDateTime dateTime)	Sums all dose amount at a time specified by the user
`java.time.LocalDateTime`	**getWhenToDose**(**IDose** dose, double amountDoseAt)	Calculates when to give next dose to reach a certain concentration amount.
`void`	**loadFile**(java.lang.String filename)	Load a saved Medicine file.
`void`	**newFile**(java.lang.String name, java.time.LocalTime tMax, java.time.LocalTime halfLife)	Create a new Medicine instance.
`void`	**removeAllDoses**()	Delete all doses from the dosages array.
`void`	**removeDose**(int index)	Remove a dose by its index.
`void`	**removeTestDoses**()	Removes all test doses.
`void`	**saveFile**(java.lang.String filename)	Save Medicine to a file.

Method Detail

getMedicine

IMedicine getMedicine()

Retrieves the Medicine instance.

Returns:

Medicine instance.

addDose

```
void addDose(java.time.LocalDateTime dateTimeTakeDose,
             double amount,
             boolean isTest)
```

removeDose

```
void removeDose(int index)
```

Remove a dose by its index.

Parameters:

`index` - index of dose.

removeAllDoses

```
void removeAllDoses()
```

Delete all doses from the dosages array.

saveFile

```
void saveFile(java.lang.String filename)
```

Save Medicine to a file. By default, save directory is user's home directory. Dose and Medicine MUST implement Serializable for this feature to work.

Parameters:

`filename` - Name of save file to be saved.

loadFile

```
void loadFile(java.lang.String filename)
```

Load a saved Medicine file. By default, load directory is user's home directory. Dose and Medicine MUST implement Serializable for this feature to work.

Parameters:

`filename` - Name of the file to be loaded.

newFile

```
void newFile(java.lang.String name,
             java.time.LocalTime tMax,
             java.time.LocalTime halfLife)
```

Create a new Medicine instance.

Parameters:

`name` - Name of medicine

`tMax` - TMax of medicine.

`halfLife` - Half life of medicine.

getConcentrationAtTime

```
java.lang.Double getConcentrationAtTime(IDose doseIn,
                                        java.time.LocalDateTime dateTimeAtIn)
```

Calculate concentration amount at specified time.

Parameters:

`doseIn` - Dose instance.

`dateTimeAtIn` - Specified time.

Returns:

Concentration amount at specific time.

getSumConcentrationsAtTime

```
java.lang.Double getSumConcentrationsAtTime(java.time.LocalDateTime dateTime)
```

Sums all dose amount at a time specified by the user

Parameters:

`dateTime` - Specified time

Returns:

Sum of dose amounts at specified time.

createLocalDate

java.time.LocalDate createLocalDate(java.lang.String date)

Creates a LocalDate instance by parsing a String.

Parameters:

`date` - Date as a String.

Returns:

LocalDate instance.

createLocalTime

java.time.LocalTime createLocalTime(java.lang.String time)

Creates a LocalTime instance by parsing a String.

Parameters:

`time` - Time as a String.

Returns:

LocalTime instance.

removeTestDoses

void removeTestDoses()

Removes all test doses.

getPeakLevel

java.time.LocalDateTime getPeakLevel(IDose dose)

Calculates when Dose is going to reach its peak level of concentration.

Parameters:

`dose` - Dose instance.

Returns:

Time when Dose reaches its peak level.

getWhenToDose

```
java.time.LocalDateTime getWhenToDose(IDose dose,
                                      double amountDoseAt)
```

Calculates when to give next dose to reach a certain concentration amount.

Parameters:

`dose` - Dose instance.

`amountDoseAt` - Desired dose concentration amount.

Returns:

Time when doses reaches specified amount.

OVERVIEW PACKAGE CLASS TREE DEPRECATED INDEX HELP

PREV CLASS NEXT CLASS FRAMES NO FRAMES ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

OVERVIEW PACKAGE CLASS TREE DEPRECATED INDEX HELP

PREV CLASS **NEXT CLASS** FRAMES NO FRAMES ALL CLASSES SEARCH: Search

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Package Model

Class Dose

java.lang.Object
    Model.Dose

All Implemented Interfaces:

`java.io.Serializable`, `IDose`

```
public class Dose
extends java.lang.Object
implements java.io.Serializable, IDose
```

See Also:

Serialized Form

Field Summary

Fields

Modifier and Type	Field	Description
`private java.lang.Double`	**`amountDose`**	Amount of dose.
`private java.time.LocalDateTime`	**`dateTimeTakeDose`**	Keep time when dose is taken.
`private java.lang.Boolean`	**`isTestDose`**	Defines whether dose is a test dose Default: false.

Constructor Summary

Constructors

Constructor	Description
`Dose`()	

Method Summary

All Methods | Instance Methods | Concrete Methods

Modifier and Type	Method	Description
`void`	**`createDose`**(java.time.LocalDateTime dateTimeTakeDose, double amountDose, java.lang.Boolean isTestDose)	Creates a Dose instance
`java.lang.Double`	**`getAmountDose`**()	Concentration amount of the dose taken by patient.
`java.time.LocalDateTime`	**`getDateTimeTakeDose`**()	Time when the dose is taken by patient.
`boolean`	**`getIsTestDose`**()	Identifies dose type.
`java.lang.String`	**`toString`**()	String representation of Dose.

Methods inherited from class java.lang.Object

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

Field Detail

dateTimeTakeDose

private java.time.LocalDateTime dateTimeTakeDose

Keep time when dose is taken. Default: current time.

amountDose

private java.lang.Double amountDose

Amount of dose. Default: 1.

isTestDose

```
private java.lang.Boolean isTestDose
```

Defines whether dose is a test dose Default: false.

Constructor Detail

Dose

```
public Dose()
```

Method Detail

createDose

```
public void createDose(java.time.LocalDateTime dateTimeTakeDose,
                       double amountDose,
                       java.lang.Boolean isTestDose)
```

Description copied from interface: `IDose`

Creates a Dose instance

Specified by:

`createDose` in interface `IDose`

Parameters:

`dateTimeTakeDose` - Date and time when dose is taken.

`amountDose` - Amount of dose taken.

`isTestDose` - Defines whether the dose is a test dose.

getAmountDose

```
public java.lang.Double getAmountDose()
```

Description copied from interface: `IDose`

Concentration amount of the dose taken by patient.

Specified by:

`getAmountDose` in interface `IDose`

Returns:

Concentration amount of Dose.

getDateTimeTakeDose

public java.time.LocalDateTime getDateTimeTakeDose()

Description copied from interface: `IDose`

Time when the dose is taken by patient.

Specified by:

`getDateTimeTakeDose` in interface `IDose`

Returns:

LocalTime Time of Dose.

toString

public java.lang.String toString()

Description copied from interface: `IDose`

String representation of Dose.

Specified by:

`toString` in interface `IDose`

Overrides:

`toString` in class `java.lang.Object`

Returns:

Dose type, dose amount, and dose taken time.

getIsTestDose

public boolean getIsTestDose()

Description copied from interface: `IDose`

Identifies dose type.

Specified by:

`getIsTestDose` in interface `IDose`

Returns:

True = TestDose (what-if dose); False = Dose (actual dose);

OVERVIEW PACKAGE CLASS TREE DEPRECATED INDEX HELP

PREV CLASS **NEXT CLASS** FRAMES NO FRAMES ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Package Model

Interface IDose

All Known Implementing Classes:

`Dose`

```
public interface IDose
```

Method Summary

All Methods | Instance Methods | Abstract Methods

Modifier and Type	Method	Description
`void`	`createDose`(java.time.LocalDateTime dateTimeTakeDose, double amount, java.lang.Boolean isTestDose)	Creates a Dose instance
`java.lang.Double`	`getAmountDose`()	Concentration amount of the dose taken by patient.
`java.time.LocalDateTime`	`getDateTimeTakeDose`()	Time when the dose is taken by patient.
`boolean`	`getIsTestDose`()	Identifies dose type.
`java.lang.String`	`toString`()	String representation of Dose.

Method Detail

createDose

void createDose(java.time.LocalDateTime dateTimeTakeDose,

double amount,
java.lang.Boolean isTestDose)

Creates a Dose instance

Parameters:

`dateTimeTakeDose` - Date and time when dose is taken.

`amount` - Amount of dose taken.

`isTestDose` - Defines whether the dose is a test dose.

getDateTimeTakeDose

java.time.LocalDateTime getDateTimeTakeDose()

Time when the dose is taken by patient.

Returns:

LocalTime Time of Dose.

getAmountDose

java.lang.Double getAmountDose()

Concentration amount of the dose taken by patient.

Returns:

Concentration amount of Dose.

toString

java.lang.String toString()

String representation of Dose.

Overrides:

`toString` in class `java.lang.Object`

Returns:

Dose type, dose amount, and dose taken time.

getIsTestDose

boolean getIsTestDose()

Identifies dose type.

Returns:

True = TestDose (what-if dose); False = Dose (actual dose);

OVERVIEW PACKAGE CLASS TREE DEPRECATED INDEX HELP

PREV CLASS **NEXT CLASS** FRAMES NO FRAMES ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Package Model

Class Medicine

java.lang.Object
    Model.Medicine

All Implemented Interfaces:

`java.io.Serializable`, `IMedicine`

```
public class Medicine
extends java.lang.Object
implements java.io.Serializable, IMedicine
```

See Also:

Serialized Form

Field Summary

Fields

Modifier and Type	Field	Description
`private java.util.ArrayList<IDose>`	**`doses`**	Array of doses
`private java.lang.String`	**`nameMedicine`**	Medicine name
`private java.time.LocalTime`	**`timeHalfLifeMedicine`**	Medicine half life Time it takes for the concentration to be reduced by half of its amount.
`private java.time.LocalTime`	**`timeMaxMedicine`**	Medicine tMax Time when the concentration will be at its peak.

Constructor Summary

Constructors

Constructor	Description

Medicine()

Method Summary

All Methods | Instance Methods | Concrete Methods

Modifier and Type	Method	Description
`void`	**addDose**(java.time.LocalDateTime dateTimeTakeDose, double amount, java.lang.Boolean isTestDose)	Add a Dose to doses array.
`void`	**createMedicine**(java.lang.String nameMedicine, java.time.LocalTime timeMaxMedicine, java.time.LocalTime timeHalfLifeMedicine)	Creates a Medicine instance.
`java.util.ArrayList<IDose>`	**getDoses**()	Corresponds to all existing doses.
`java.lang.String`	**getNameMedicine**()	Name of medicine.
`java.time.LocalTime`	**getTimeHalfLifeMedicine**()	Half life time of the medicine.
`java.time.LocalTime`	**getTimeMaxMedicine**()	Tmax of medicine
`void`	**removeAllDoses**()	Removes all doses from the doses array in the medicine.
`void`	**removeDose**(int index)	Remove a dose by its index.
`void`	**removeTestDoses**()	Remove all test doses from the doses array.
`java.lang.String`	**toString**()	String representation of Medicine.

Methods inherited from class java.lang.Object

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

Field Detail

nameMedicine

private java.lang.String nameMedicine

Medicine name

timeMaxMedicine

private java.time.LocalTime timeMaxMedicine

Medicine tMax Time when the concentration will be at its peak.

timeHalfLifeMedicine

private java.time.LocalTime timeHalfLifeMedicine

Medicine half life Time it takes for the concentration to be reduced by half of its amount.

doses

private java.util.ArrayList<IDose> doses

Array of doses

Constructor Detail

Medicine

public Medicine()

Method Detail

createMedicine

```
public void createMedicine(java.lang.String nameMedicine,
                           java.time.LocalTime timeMaxMedicine,
                           java.time.LocalTime timeHalfLifeMedicine)
```

Description copied from interface: `IMedicine`

Creates a Medicine instance.

Specified by:

`createMedicine` in interface `IMedicine`

Parameters:

`nameMedicine` - Name of medicine.

`timeMaxMedicine` - Time required for medicine to be at its peak concentration.

`timeHalfLifeMedicine` - Time required for medicine to decrease by half.

getNameMedicine

```
public java.lang.String getNameMedicine()
```

Description copied from interface: `IMedicine`

Name of medicine.

Specified by:

`getNameMedicine` in interface `IMedicine`

Returns:

Name of the medicine.

getTimeHalfLifeMedicine

```
public java.time.LocalTime getTimeHalfLifeMedicine()
```

Description copied from interface: `IMedicine`

Half life time of the medicine.

Specified by:

`getTimeHalfLifeMedicine` in interface `IMedicine`

Returns:

Half life time of the medicine.

getDoses

```
public java.util.ArrayList<IDose> getDoses()
```

Description copied from interface: IMedicine

Corresponds to all existing doses. Includes type Dose and TestDose.

Specified by:

`getDoses` in interface `IMedicine`

Returns:

An array containing all doses.

getTimeMaxMedicine

```
public java.time.LocalTime getTimeMaxMedicine()
```

Description copied from interface: IMedicine

Tmax of medicine

Specified by:

`getTimeMaxMedicine` in interface `IMedicine`

Returns:

TMax of the medicine.

addDose

```
public void addDose(java.time.LocalDateTime dateTimeTakeDose,
                    double amount,
                    java.lang.Boolean isTestDose)
```

Description copied from interface: IMedicine

Add a Dose to doses array.

Specified by:

`addDose` in interface `IMedicine`

Parameters:

`dateTimeTakeDose` - Time dose is taken.

`amount` - Amount of dose.

`isTestDose` - defines whether is test dose.

removeAllDoses

public void removeAllDoses()

Description copied from interface: IMedicine

Removes all doses from the doses array in the medicine.

Specified by:

`removeAllDoses` in interface `IMedicine`

removeDose

public void removeDose(int index)

Description copied from interface: IMedicine

Remove a dose by its index.

Specified by:

`removeDose` in interface `IMedicine`

Parameters:

`index` - index of dose.

toString

public java.lang.String toString()

Description copied from interface: IMedicine

String representation of Medicine.

Specified by:

`toString` in interface `IMedicine`

Overrides:

`toString` in class `java.lang.Object`

Returns:

String medicine name, time max, and half life.

removeTestDoses

public void removeTestDoses()

Description copied from interface: IMedicine

Remove all test doses from the doses array.

Specified by:

`removeTestDoses` in interface `IMedicine`

OVERVIEW PACKAGE CLASS TREE DEPRECATED INDEX HELP

PREV CLASS NEXT CLASS FRAMES NO FRAMES ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Package Model

Interface IMedicine

All Known Implementing Classes:

`Medicine`

`public interface` **`IMedicine`**

Method Summary

All Methods | **Instance Methods** | **Abstract Methods**

Modifier and Type	Method	Description
`void`	**`addDose`**(java.time.LocalDateTime dateTimeTakeDose, double amount, java.lang.Boolean isTestDose)	Add a Dose to doses array.
`void`	**`createMedicine`**(java.lang.String nameMedicine, java.time.LocalTime timeMaxMedicine, java.time.LocalTime timeHalfLifeMedicine)	Creates a Medicine instance.
`java.util.ArrayList<`**`IDose`**`>`	**`getDoses`**()	Corresponds to all existing doses.
`java.lang.String`	**`getNameMedicine`**()	Name of medicine.
`java.time.LocalTime`	**`getTimeHalfLifeMedicine`**()	Half life time of the medicine.
`java.time.LocalTime`	**`getTimeMaxMedicine`**()	Tmax of medicine
`void`	**`removeAllDoses`**()	Removes all doses from the doses array in the medicine.
`void`	**`removeDose`**(int index)	Remove a dose by its index.
`void`	**`removeTestDoses`**()	Remove all test doses from the doses

		array.
`java.lang.String`	**toString**()	String representation of Medicine.

Method Detail

getNameMedicine

java.lang.String getNameMedicine()

Name of medicine.

Returns:

Name of the medicine.

getTimeHalfLifeMedicine

java.time.LocalTime getTimeHalfLifeMedicine()

Half life time of the medicine.

Returns:

Half life time of the medicine.

getTimeMaxMedicine

java.time.LocalTime getTimeMaxMedicine()

Tmax of medicine

Returns:

TMax of the medicine.

getDoses

java.util.ArrayList<IDose> getDoses()

Corresponds to all existing doses. Includes type Dose and TestDose.

Returns:

An array containing all doses.

createMedicine

```
void createMedicine(java.lang.String nameMedicine,
                    java.time.LocalTime timeMaxMedicine,
                    java.time.LocalTime timeHalfLifeMedicine)
```

Creates a Medicine instance.

Parameters:

`nameMedicine` - Name of medicine.

`timeMaxMedicine` - Time required for medicine to be at its peak concentration.

`timeHalfLifeMedicine` - Time required for medicine to decrease by half.

addDose

```
void addDose(java.time.LocalDateTime dateTimeTakeDose,
             double amount,
             java.lang.Boolean isTestDose)
```

Add a Dose to doses array.

Parameters:

`dateTimeTakeDose` - Time dose is taken.

`amount` - Amount of dose.

`isTestDose` - defines whether is test dose.

removeAllDoses

```
void removeAllDoses()
```

Removes all doses from the doses array in the medicine.

removeDose

```
void removeDose(int index)
```

Remove a dose by its index.

Parameters:

`index` - index of dose.

removeTestDoses

```
void removeTestDoses()
```

Remove all test doses from the doses array.

toString

```
java.lang.String toString()
```

String representation of Medicine.

Overrides:

`toString` in class `java.lang.Object`

Returns:

String medicine name, time max, and half life.

OVERVIEW PACKAGE CLASS TREE DEPRECATED INDEX HELP

PREV CLASS **NEXT CLASS** FRAMES NO FRAMES ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

OVERVIEW PACKAGE CLASS TREE DEPRECATED INDEX HELP

PREV CLASS NEXT CLASS FRAMES NO FRAMES ALL CLASSES

SEARCH: Search

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Package View

Class View

java.lang.Object
    View.View

```
public class View
extends java.lang.Object
```

Field Summary

Fields

Modifier and Type	Field	Description
`private static Controller`	`controller`	
`static java.util.Scanner`	`userInput`	Read from user input

Constructor Summary

Constructors

Constructor	Description
`View()`	

Method Summary

All Methods **Static Methods** **Concrete Methods**

Modifier and Type	Method	Description
`private static void`	`addDose(boolean isTest)`	Creates Dose from user input.
`private static void`	`clear()`	Clears the console screen.

`private static java.time.LocalDate`	**`createLocalDate`**()	Creates LocalDate from user input.
`private static java.time.LocalTime`	**`createLocalTime`**()	Creates LocalTime instance from user input.
`private static void`	**`createNewFile`**()	Creates a Medicine instance from user input.
`static void`	**`main`**(java.lang.String[] args)	
`static void`	**`pause`**()	Pause scrolling fot the console screen until user hits the Enter key.
`private static void`	**`printConcentrationAtTime`**(java.time.LocalDateTime dateTime)	Prints concentration amount at specific time.
`private static void`	**`printDoses`**()	Prints Dose index, time taken, and quantity.
`private static void`	**`printMedicine`**()	Prints Medicine name, time max, and half life.
`private static void`	**`printPeakLevelAt`**(java.lang.Boolean includeTestDoses)	Prints peak level of concentration.
`private static void`	**`printWhenToDose`**(double amountDose)	Prints time when to dose to obtain specified dose amount.
`private static void`	**`removeDose`**()	Removes dose using an index selected by the user.
`private static void`	**`selectAction`**()	Display the list of actions

		that can be performed by the application.
`private static void`	**start**()	Display a welcome screen which allows the user to select whether to create a file or open an existing one.

Methods inherited from class java.lang.Object

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

Field Detail

userInput

public static java.util.Scanner userInput

Read from user input

controller

private static Controller controller

Constructor Detail

View

public View()

Method Detail

printDoses

private static void printDoses()

Prints Dose index, time taken, and quantity.

printMedicine

private static void printMedicine()

Prints Medicine name, time max, and half life.

printConcentrationAtTime

private static void printConcentrationAtTime(java.time.LocalDateTime dateTime)

Prints concentration amount at specific time.

Parameters:

`dateTime` - Specific time to calculate concentration amount.

createLocalDate

private static java.time.LocalDate createLocalDate()

Creates LocalDate from user input.

Returns:

LocalDate instance.

createLocalTime

private static java.time.LocalTime createLocalTime()

Creates LocalTime instance from user input.

Returns:

Localtime instance.

createNewFile

private static void createNewFile()

Creates a Medicine instance from user input.

addDose

private static void addDose(boolean isTest)

Creates Dose from user input.

Parameters:

`isTest` - Defines whether is a test dose.

removeDose

private static void removeDose()

Removes dose using an index selected by the user.

printPeakLevelAt

private static void printPeakLevelAt(java.lang.Boolean includeTestDoses)

Prints peak level of concentration. The output can be test Doses and actual Doses, or actual Doses only.

Parameters:

`includeTestDoses` - Define whether to consider test doses.

printWhenToDose

private static void printWhenToDose(double amountDose)

Prints time when to dose to obtain specified dose amount.

Parameters:

`amountDose` - Dose amount.

clear

private static void clear()

Clears the console screen.

pause

public static void pause()

Pause scrolling fot the console screen until user hits the Enter key.

start

private static void start()

Display a welcome screen which allows the user to select whether to create a file or open an existing one.

selectAction

private static void selectAction()

Display the list of actions that can be performed by the application. The user must enter the corresponding number to execute the controller.

main

public static void main(java.lang.String[] args)

OVERVIEW PACKAGE CLASS TREE DEPRECATED INDEX HELP

PREV CLASS NEXT CLASS FRAMES NO FRAMES ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

A C D G I L M N P R S T U V

C

clear() - Static method in class View.View

Clears the console screen.

controller - Static variable in class View.View

Controller - package Controller

Controller - Class in Controller

Controller() - Constructor for class Controller.Controller

createDose(LocalDateTime, double, Boolean) - Method in class Model.Dose

createDose(LocalDateTime, double, Boolean) - Method in interface Model.IDose

Creates a Dose instance

createLocalDate() - Static method in class View.View

Creates LocalDate from user input.

createLocalDate(String) - Method in class Controller.Controller

createLocalDate(String) - Method in interface Controller.IController

Creates a LocalDate instance by parsing a String.

createLocalTime() - Static method in class View.View

Creates LocalTime instance from user input.

createLocalTime(String) - Method in class Controller.Controller

createLocalTime(String) - Method in interface Controller.IController

Creates a LocalTime instance by parsing a String.

createMedicine(String, LocalTime, LocalTime) - Method in interface Model.IMedicine

Creates a Medicine instance.

createMedicine(String, LocalTime, LocalTime) - Method in class Model.Medicine

createNewFile() - Static method in class View.View

Creates a Medicine instance from user input.

A C D G I L M N P R S T U V

A C D G I L M N P R S T U V

D

dateTimeTakeDose - Variable in class Model.Dose

Keep time when dose is taken.

Dose - Class in Model

Dose() - Constructor for class Model.Dose

doses - Variable in class Model.Medicine

Array of doses

A C D G I L M N P R S T U V

A C D G I L M N P R S T U V

G

getAmountDose() - Method in class Model.Dose

getAmountDose() - Method in interface Model.IDose

Concentration amount of the dose taken by patient.

getConcentrationAtTime(IDose, LocalDateTime) - Method in class Controller.Controller

getConcentrationAtTime(IDose, LocalDateTime) - Method in interface Controller.IController

Calculate concentration amount at specified time.

getDateTimeTakeDose() - Method in class Model.Dose

getDateTimeTakeDose() - Method in interface Model.IDose

Time when the dose is taken by patient.

getDoses() - Method in interface Model.IMedicine

Corresponds to all existing doses.

getDoses() - Method in class Model.Medicine

getIsTestDose() - Method in class Model.Dose

getIsTestDose() - Method in interface Model.IDose

Identifies dose type.

getMedicine() - Method in class Controller.Controller

getMedicine() - Method in interface Controller.IController

Retrieves the Medicine instance.

getNameMedicine() - Method in interface Model.IMedicine

Name of medicine.

getNameMedicine() - Method in class Model.Medicine

getPeakLevel(IDose) - Method in class Controller.Controller

getPeakLevel(IDose) - Method in interface Controller.IController

Calculates when Dose is going to reach its peak level of concentration.

getSumConcentrationsAtTime(LocalDateTime) - Method in class Controller.Controller

getSumConcentrationsAtTime(LocalDateTime) - Method in interface Controller.IController

Sums all dose amount at a time specified by the user

getTimeHalfLifeMedicine() - Method in interface Model.IMedicine

Half life time of the medicine.

getTimeHalfLifeMedicine() - Method in class Model.Medicine

getTimeMaxMedicine() - Method in interface Model.IMedicine

Tmax of medicine

A C D G I L M N P R S T U V

OVERVIEW PACKAGE CLASS TREE DEPRECATED INDEX HELP

PREV LETTER NEXT LETTER FRAMES NO FRAMES ALL CLASSES

A C D G I L M N P R S T U V

I

IController - Interface in Controller

IDose - Interface in Model

IMedicine - Interface in Model

isTestDose - Variable in class Model.Dose

Defines whether dose is a test dose Default: false.

A C D G I L M N P R S T U V

A C D G I L M N P R S T U V

L

loadFile(String) - Method in class Controller.Controller

loadFile(String) - Method in interface Controller.IController

Load a saved Medicine file.

A C D G I L M N P R S T U V

A C D G I L M N P R S T U V

M

main(String[]) - Static method in class View.View

medicine - Variable in class Controller.Controller

Medicine file

Medicine - Class in Model

Medicine() - Constructor for class Model.Medicine

Model - package Model

A C D G I L M N P R S T U V

OVERVIEW PACKAGE CLASS TREE DEPRECATED INDEX HELP

PREV LETTER NEXT LETTER FRAMES NO FRAMES ALL CLASSES

SEARCH: Search

A C D G I L M N P R S T U V

A C D G I L M N P R S T U V

OVERVIEW PACKAGE CLASS TREE DEPRECATED INDEX HELP

PREV LETTER NEXT LETTER FRAMES NO FRAMES ALL CLASSES

A C D G I L M N P R S T U V

P

parseValidAmount() - Static method in class Controller.Controller

Verifies if parsed input is a valid number amount.

parseValidInt() - Static method in class Controller.Controller

Verifies if parsed input is a valid integer.

pause() - Static method in class View.View

Pause scrolling fot the console screen until user hits the Enter key.

printConcentrationAtTime(LocalDateTime) - Static method in class View.View

Prints concentration amount at specific time.

printDoses() - Static method in class View.View

Prints Dose index, time taken, and quantity.

printMedicine() - Static method in class View.View

Prints Medicine name, time max, and half life.

printPeakLevelAt(Boolean) - Static method in class View.View

Prints peak level of concentration.

printWhenToDose(double) - Static method in class View.View

Prints time when to dose to obtain specified dose amount.

A C D G I L M N P R S T U V

A C D G I L M N P R S T U V

R

removeAllDoses() - Method in class Controller.Controller

removeAllDoses() - Method in interface Controller.IController

Delete all doses from the dosages array.

removeAllDoses() - Method in interface Model.IMedicine

Removes all doses from the doses array in the medicine.

removeAllDoses() - Method in class Model.Medicine

removeDose() - Static method in class View.View

Removes dose using an index selected by the user.

removeDose(int) - Method in class Controller.Controller

removeDose(int) - Method in interface Controller.IController

Remove a dose by its index.

removeDose(int) - Method in interface Model.IMedicine

Remove a dose by its index.

removeDose(int) - Method in class Model.Medicine

removeTestDoses() - Method in class Controller.Controller

removeTestDoses() - Method in interface Controller.IController

Removes all test doses.

removeTestDoses() - Method in interface Model.IMedicine

Remove all test doses from the doses array.

removeTestDoses() - Method in class Model.Medicine

A C D G I L M N P R S T U V

A C D G I L M N P R S T U V

S

saveFile(String) - Method in class Controller.Controller

saveFile(String) - Method in interface Controller.IController

Save Medicine to a file.

selectAction() - Static method in class View.View

Display the list of actions that can be performed by the application.

start() - Static method in class View.View

Display a welcome screen which allows the user to select whether to create a file or open an existing one.

A C D G I L M N P R S T U V

A C D G I L M N P R S T U V

T

timeHalfLifeMedicine - Variable in class Model.Medicine

Medicine half life Time it takes for the concentration to be reduced by half of its amount.

timeMaxMedicine - Variable in class Model.Medicine

Medicine tMax Time when the concentration will be at its peak.

timezone - Variable in class Controller.Controller

Defines time zone.

toString() - Method in class Model.Dose

toString() - Method in interface Model.IDose

String representation of Dose.

toString() - Method in interface Model.IMedicine

String representation of Medicine.

toString() - Method in class Model.Medicine

A C D G I L M N P R S T U V

A C D G I L M N P R S T U V

U

userInput - Static variable in class View.View

Read from user input

A C D G I L M N P R S T U V

A C D G I L M N P R S T U V

A C D G I L M N P R S T U V

Constant Field Values

Contents

Serialized Form

Package Model

Class Model.Dose extends java.lang.Object implements Serializable

Serialized Fields

amountDose

```
java.lang.Double amountDose
```

Amount of dose. Default: 1.

dateTimeTakeDose

```
java.time.LocalDateTime dateTimeTakeDose
```

Keep time when dose is taken. Default: current time.

isTestDose

```
java.lang.Boolean isTestDose
```

Defines whether dose is a test dose Default: false.

Class Model.Medicine extends java.lang.Object implements Serializable

Serialized Fields

doses

```
java.util.ArrayList<E extends java.lang.Object> doses
```

Array of doses

nameMedicine

```
java.lang.String nameMedicine
```

Medicine name

timeHalfLifeMedicine

```
java.time.LocalTime timeHalfLifeMedicine
```

Medicine half life Time it takes for the concentration to be reduced by half of its amount.

timeMaxMedicine

```
java.time.LocalTime timeMaxMedicine
```

Medicine tMax Time when the concentration will be at its peak.

OVERVIEW PACKAGE CLASS TREE DEPRECATED INDEX HELP

PREV NEXT FRAMES NO FRAMES ALL CLASSES