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`**(**`IDose`** dose)	Create an IDose Object
`void`	**`createMedicine`**(java.lang.String nameMedicine, java.time.LocalTime timeMaxMedicine, java.time.LocalTime timeHalfLifeMedicine)	Creates a medicine object.
`java.lang.Double`	**`getConcentrationsAtTime`**(java.time.LocalDateTime dateTime)	Calculate the total concentration amount of the medicine (sum of all doses amount)at a specific dateTime.
`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

Method Detail

getNameMedicine

java.lang.String getNameMedicine()

Name of medicine.

Returns:

Return name of the medicine.

getTimeHalfLifeMedicine

java.time.LocalTime getTimeHalfLifeMedicine()

Half life time of the medicine.

Returns:

Return the half life time of the medicine.

getTimeMaxMedicine

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

Tmax of medicine

Returns:

Return TMax of 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 object.

Parameters:

`nameMedicine` - Name of medicine.

`timeMaxMedicine` - Time when medicine is at its peak concentration.

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

getConcentrationsAtTime

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

Calculate the total concentration amount of the medicine (sum of all doses amount)at a specific dateTime.

Parameters:

`dateTime` - specified dateTame

Returns:

Concentration amount of dose at a specified time.

addDose

void addDose(IDose dose)

Create an IDose Object

Parameters:

`dose` - IDose Object

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

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 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

Class Medicine

java.lang.Object
    Medicine

All Implemented Interfaces:

`IMedicine`, `java.io.Serializable`

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

Field Summary

Fields

Modifier and Type	Field	Description
`private java.util.ArrayList<IDose>`	**`doses`**	
`private java.lang.String`	**`nameMedicine`**	Medicine name
`private java.time.LocalTime`	**`timeHalfLifeMedicine`**	Medicine half life.
`private java.time.LocalTime`	**`timeMaxMedicine`**	Medicine tMax.

Constructor Summary

Constructors

Constructor	Description
`Medicine`()	
`Medicine`(java.lang.String name, java.time.LocalTime tMax, java.time.LocalTime timeHalfLifeMedicine)	

Method Summary

All Methods | Instance Methods | Concrete Methods

Modifier and Type	Method	Description
`void`	**`addDose`**(**`IDose`** dose)	Create an IDose Object
`void`	**`createMedicine`**(java.lang.String nameMedicine, java.time.LocalTime timeMaxMedicine, java.time.LocalTime timeHalfLifeMedicine)	Creates a medicine object.
`java.lang.Double`	**`getConcentrationsAtTime`**(java.time.LocalDateTime dateTime)	Calculate the total concentration amount of the medicine (sum of all doses amount)at a specific dateTime.
`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`**()	

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

Constructor Detail

Medicine

```
public Medicine()
```

Medicine

```
public Medicine(java.lang.String name,
                java.time.LocalTime tMax,
                java.time.LocalTime timeHalfLifeMedicine)
```

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 object.

Specified by:

`createMedicine` in interface `IMedicine`

Parameters:

`nameMedicine` - Name of medicine.

`timeMaxMedicine` - Time when medicine is 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:

Return 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:

Return the 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:

Return TMax of medicine.

addDose

```
public void addDose(IDose dose)
```

Description copied from interface: `IMedicine`

Create an IDose Object

Specified by:

`addDose` in interface `IMedicine`

Parameters:

`dose` - IDose Object

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

getConcentrationsAtTime

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

Description copied from interface: `IMedicine`

Calculate the total concentration amount of the medicine (sum of all doses amount)at a specific dateTime.

Specified by:

`getConcentrationsAtTime` in interface `IMedicine`

Parameters:

`dateTime` - specified dateTame

Returns:

Concentration amount of dose at a specified time.

toString

public java.lang.String toString()

Overrides:

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

removeTestDoses

public void removeTestDoses()

Description copied from interface: `IMedicine`

Remove all test doses from the doses array

Specified by:

`removeTestDoses` in interface `IMedicine`

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 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

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`**(**`IDose`** dose)	Create an IDose Object
`void`	**`createMedicine`**(java.lang.String nameMedicine, java.time.LocalTime timeMaxMedicine, java.time.LocalTime timeHalfLifeMedicine)	Creates a medicine object.
`java.lang.Double`	**`getConcentrationsAtTime`**(java.time.LocalDateTime dateTime)	Calculate the total concentration amount of the medicine (sum of all doses amount)at a specific dateTime.
`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

Method Detail

getNameMedicine

java.lang.String getNameMedicine()

Name of medicine.

Returns:

Return name of the medicine.

getTimeHalfLifeMedicine

java.time.LocalTime getTimeHalfLifeMedicine()

Half life time of the medicine.

Returns:

Return the half life time of the medicine.

getTimeMaxMedicine

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

Tmax of medicine

Returns:

Return TMax of 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 object.

Parameters:

`nameMedicine` - Name of medicine.

`timeMaxMedicine` - Time when medicine is at its peak concentration.

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

getConcentrationsAtTime

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

Calculate the total concentration amount of the medicine (sum of all doses amount)at a specific dateTime.

Parameters:

`dateTime` - specified dateTame

Returns:

Concentration amount of dose at a specified time.

addDose

void addDose(IDose dose)

Create an IDose Object

Parameters:

`dose` - IDose Object

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

PACKAGE CLASS TREE DEPRECATED INDEX HELP

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

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

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)	Creates a Dose object.
`java.lang.Double`	**`getAmountDose`**()	Concentration amount of the dose taken by patient.
`java.lang.Double`	**`getConcentrationAtTime`**(java.time.LocalDateTime timeIn, java.time.LocalTime timeMaxIn, java.time.LocalTime timeHalfLifeIn)	Calculate dose concentration amount at a specified time.
`java.time.LocalDateTime`	**`getDateTimeTakeDose`**()	Time when the dose is taken by patient.
`boolean`	**`getIsTestDose`**()	Identifies dose type.
`void`	**`setTestDose`**()	Sets dose to test dose.

Method Detail

createDose

```
void createDose(java.time.LocalDateTime dateTimeTakeDose,
                double amount)
```

Creates a Dose object.

Parameters:

`dateTimeTakeDose` - Time when the dose is taken by patient.

`amount` - Concentration amount of the dose taken by patient.

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.

getConcentrationAtTime

```
java.lang.Double getConcentrationAtTime(java.time.LocalDateTime timeIn,
                                        java.time.LocalTime timeMaxIn,
                                        java.time.LocalTime timeHalfLifeIn)
```

Calculate dose concentration amount at a specified time.

Parameters:

`timeIn` - Time when dose is taken.

`timeMaxIn` - tMax of medicine.

`timeHalfLifeIn` - half life of medicine.

Returns:

Concentration amount of dose amount at a specified time.

getIsTestDose

boolean getIsTestDose()

Identifies dose type.

Returns:

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

setTestDose

void setTestDose()

Sets dose to test dose. Used for the what-if feature of the

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 <Unnamed>

Interface Summary

Interface	Description
IAction	
IDose	
IMedicine	

Class Summary

Class	Description
Action	
Controller	
Dose	
Medicine	

Hierarchy For Package <Unnamed>

Class Hierarchy

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

Interface Hierarchy

- **IAction**
- **IDose**
- **IMedicine**

Deprecated API

Contents

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

A

action - Static variable in class Controller

Action - Class in <Unnamed>

Action() - Constructor for class Action

Action(IMedicine) - Constructor for class Action

addDose(IDose) - Method in class Action

addDose(IDose) - Method in interface IAction

Adds a new dose.

addDose(IDose) - Method in interface IMedicine

Create an IDose Object

addDose(IDose) - Method in class Medicine

amountDose - Variable in class Dose

Amount of dose.

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

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.

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.

PACKAGE CLASS TREE DEPRECATED INDEX HELP

PREV NEXT FRAMES NO FRAMES ALL CLASSES

All Classes

Action

Controller

Dose

IAction

IDose

IMedicine

Medicine

Class Dose

java.lang.Object
    Dose

All Implemented Interfaces:

`IDose`, `java.io.Serializable`

```
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.
`private java.time.ZoneOffset`	**`timezone`**	Defines time zone.

Constructor Summary

Constructors

Constructor	Description
`Dose``()`	

Dose(java.time.LocalDateTime timeTake, double amountDose)

Dose(java.time.LocalDateTime timeTake, double amountDose, boolean isTestDose)

Method Summary

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

Modifier and Type	Method	Description
`void`	**createDose**(java.time.LocalDateTime dateTimeTakeDose, double amount)	Creates a Dose object.
`java.lang.Double`	**getAmountDose**()	Concentration amount of the dose taken by patient.
`java.lang.Double`	**getConcentrationAtTime**(java.time.LocalDateTime dateTimeAtIn, java.time.LocalTime timeMaxIn, java.time.LocalTime timeHalfLifeIn)	Calculate dose concentration amount at a specified time.
`java.time.LocalDateTime`	**getDateTimeTakeDose**()	Time when the dose is taken by patient.
`boolean`	**getIsTestDose**()	Identifies dose type.
`void`	**setTestDose**()	Sets dose to test dose.
`java.lang.String`	**toString**()	

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.

timezone

private java.time.ZoneOffset timezone

Defines time zone. Default: -05:00

Constructor Detail

Dose

public Dose()

Dose

public Dose(java.time.LocalDateTime timeTake,
double amountDose)

Dose

```
public Dose(java.time.LocalDateTime timeTake,
            double amountDose,
            boolean isTestDose)
```

Method Detail

createDose

```
public void createDose(java.time.LocalDateTime dateTimeTakeDose,
                       double amount)
```

Description copied from interface: `IDose`

Creates a Dose object.

Specified by:

`createDose` in interface `IDose`

Parameters:

`dateTimeTakeDose` - Time when the dose is taken by patient.

`amount` - Concentration amount of the dose taken by patient.

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.

getConcentrationAtTime

```
public java.lang.Double getConcentrationAtTime(java.time.LocalDateTime dateTimeAtIn,
                                               java.time.LocalTime timeMaxIn,
                                               java.time.LocalTime timeHalfLifeIn)
```

Description copied from interface: `IDose`

Calculate dose concentration amount at a specified time.

Specified by:

`getConcentrationAtTime` in interface `IDose`

Parameters:

`dateTimeAtIn` - Time when dose is taken.

`timeMaxIn` - tMax of medicine.

`timeHalfLifeIn` - half life of medicine.

Returns:

Concentration amount of dose amount at a specified time.

toString

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

Overrides:

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

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);

setTestDose

public void setTestDose()

Description copied from interface: `IDose`

Sets dose to test dose. Used for the what-if feature of the

Specified by:

`setTestDose` in interface `IDose`

Interface IAction

All Known Implementing Classes:

`Action`

```
public interface IAction
```

Method Summary

All Methods | Instance Methods | Abstract Methods

Modifier and Type	Method	Description
`void`	`addDose(IDose dose)`	Adds a new dose.
`java.lang.Double`	`getConcentrationAtTime(java.time.LocalDateTime localDateTime)`	Calculate the total concentration amount of the medicine (sum of all doses amount)at a specific dateTime.
`IMedicine`	`getMedicine()`	Retrieves the Medicine instance.
`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`	`printDoses()`	Prints all doses from Medicine's Dose array.
`void`	`printMedicine()`	Print the name,

		tmax, and halfLife of the medicine.
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:

IMedicine.Medicine instance

printDoses

void printDoses()

Prints all doses from Medicine's Dose array.

addDose

void addDose(IDose dose)

Adds a new dose. Added doses can be type Dose (actual dose) or TestDose (what-if dose)

Parameters:

`dose` - Dose object.

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.

getConcentrationAtTime

```
java.lang.Double getConcentrationAtTime(java.time.LocalDateTime localDateTime)
```

Calculate the total concentration amount of the medicine (sum of all doses amount)at a specific dateTime.

Parameters:

localDateTime - specified dateTime.

Returns:

Concentration amount at specified time.

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

printMedicine

```
void printMedicine()
```

Print the name, tmax, and halfLife of the medicine.

removeTestDoses

```
void removeTestDoses()
```

Removes all test doses.

PACKAGE CLASS TREE DEPRECATED INDEX HELP

PREV CLASS NEXT CLASS FRAMES NO FRAMES ALL CLASSES

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

Class Action

java.lang.Object
    Action

All Implemented Interfaces:

`IAction`

```
public class Action
extends java.lang.Object
implements IAction
```

Field Summary

Fields

Modifier and Type	Field	Description
`private IMedicine`	`medicine`	Medicine file

Constructor Summary

Constructors

Constructor	Description
`Action()`	
`Action(IMedicine medicine)`	

Method Summary

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

Modifier and Type	Method	Description

`void`	**`addDose`**(**`IDose`** dose)	Adds a new dose.
`java.lang.Double`	**`getConcentrationAtTime`**(java.time.LocalDateTime localDateTime)	Calculate the total concentration amount of the medicine (sum of all doses amount)at a specific dateTime.
`IMedicine`	**`getMedicine`**()	Retrieves the Medicine instance.
`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.
`void`	**`printConcentrationAtTime`**(java.time.LocalDateTime dateTime)	
`void`	**`printDetailedConcentrationAtTime`**(java.time.LocalDateTime dateTime)	
`void`	**`printDoses`**()	Prints all doses from Medicine's Dose array.
`void`	**`printMedicine`**()	Print the name, tmax, and halfLife of the medicine.
`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

medicine

private IMedicine medicine

Medicine file

Constructor Detail

Action

public Action()

Action

public Action(IMedicine medicine)

Method Detail

getMedicine

public IMedicine getMedicine()

Description copied from interface: `IAction`

Retrieves the Medicine instance.

Specified by:

`getMedicine` in interface `IAction`

Returns:

IMedicine.Medicine instance

printMedicine

public void printMedicine()

Description copied from interface: `IAction`

Print the name, tmax, and halfLife of the medicine.

Specified by:

`printMedicine` in interface `IAction`

printDoses

public void printDoses()

Description copied from interface: `IAction`

Prints all doses from Medicine's Dose array.

Specified by:

`printDoses` in interface `IAction`

addDose

public void addDose(IDose dose)

Description copied from interface: `IAction`

Adds a new dose. Added doses can be type Dose (actual dose) or TestDose (what-if dose)

Specified by:

`addDose` in interface `IAction`

Parameters:

`dose` - Dose object.

removeDose

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

Description copied from interface: `IAction`

Remove a dose by its index.

Specified by:

`removeDose` in interface `IAction`

Parameters:

`index` - index of dose

removeAllDoses

```
public void removeAllDoses()
```

Description copied from interface: `IAction`

Delete all doses from the dosages array.

Specified by:

`removeAllDoses` in interface `IAction`

getConcentrationAtTime

```
public java.lang.Double getConcentrationAtTime(java.time.LocalDateTime localDateTime)
```

Description copied from interface: `IAction`

Calculate the total concentration amount of the medicine (sum of all doses amount)at a specific dateTime.

Specified by:

`getConcentrationAtTime` in interface `IAction`

Parameters:

`localDateTime` - specified dateTime.

Returns:

Concentration amount at specified time.

printConcentrationAtTime

```
public void printConcentrationAtTime(java.time.LocalDateTime dateTime)
```

printDetailedConcentrationAtTime

```
public void printDetailedConcentrationAtTime(java.time.LocalDateTime dateTime)
```

saveFile

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

Description copied from interface: `IAction`

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 `IAction`

Parameters:

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

loadFile

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

Description copied from interface: `IAction`

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 `IAction`

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: `IAction`

Create a new Medicine instance.

Specified by:

`newFile` in interface `IAction`

Parameters:

`nameMedicine` - Name of medicine

`timeMaxMedicine` - TMax of medicine.

`timeHalfLifeMedicine` - Half life of medicine

removeTestDoses

public void removeTestDoses()

Description copied from interface: `IAction`

Removes all test doses.

Specified by:

`removeTestDoses` in interface `IAction`

Class Controller

java.lang.Object
    Controller

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

Field Summary

Fields

Modifier and Type	Field	Description
`private static Action`	`action`	
`private static java.util.Scanner`	`userInput`	

Constructor Summary

Constructors

Constructor	Description
`Controller()`	

Method Summary

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

Modifier and Type	Method	Description
`private static void`	`checkIfInteger()`	Validates if user input is an integer only.
`private static void`	`checkIfValidNumber()`	Validates if user input is a

		number, either Integer or Double.
`private static void`	**clear**()	Clears the console screen.
`private static` **IDose**	**createDose**()	Creates Dose from user input.
`private static java.time.LocalDateTime`	**createLocalDateTime**()	
`private static java.time.LocalTime`	**createLocalTime**()	Generates LocalTime instance from user input.
`private static void`	**createMedicine**()	Creates a Medicine instance from user input.
`static void`	**main**(java.lang.String[] args)	
`private static void`	**pause**()	Pause scrolling fot the console screen until user hits the Enter key.
`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**()	Starts the 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

private static java.util.Scanner userInput

action

private static Action action

Constructor Detail

Controller

public Controller()

Method Detail

createLocalTime

private static java.time.LocalTime createLocalTime()

Generates LocalTime instance from user input. Validates the user input to minimize application crash due to unexpected errors.

Returns:

LocalDateTime instance.

createLocalDateTime

private static java.time.LocalDateTime createLocalDateTime()

createMedicine

private static void createMedicine()

Creates a Medicine instance from user input.

createDose

private static IDose createDose()

Creates Dose from user input.

Returns:

A Dose instance

removeDose

private static void removeDose()

Removes dose using an index selected by the user.

checkIfInteger

private static void checkIfInteger()

Validates if user input is an integer only. If input is not an integer, the program stops.

checkIfValidNumber

private static void checkIfValidNumber()

Validates if user input is a number, either Integer or Double. If input is not an number, the program stops.

clear

private static void clear()

Clears the console screen.

pause

private static void pause()

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

start

private static void start()

Starts the 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 action.

main

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

PACKAGE CLASS TREE DEPRECATED INDEX HELP

PREV CLASS NEXT CLASS FRAMES NO FRAMES ALL CLASSES

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

Hierarchy For All Packages

Class Hierarchy

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

Interface Hierarchy

- **IAction**
- **IDose**
- **IMedicine**

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

C

checkIfInteger() - Static method in class Controller

Validates if user input is an integer only.

checkIfValidNumber() - Static method in class Controller

Validates if user input is a number, either Integer or Double.

clear() - Static method in class Controller

Clears the console screen.

Controller - Class in <Unnamed>

Controller() - Constructor for class Controller

createDose() - Static method in class Controller

Creates Dose from user input.

createDose(LocalDateTime, double) - Method in class Dose

createDose(LocalDateTime, double) - Method in interface IDose

Creates a Dose object.

createLocalDateTime() - Static method in class Controller

createLocalTime() - Static method in class Controller

Generates LocalTime instance from user input.

createMedicine() - Static method in class Controller

Creates a Medicine instance from user input.

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

Creates a medicine object.

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

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

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

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

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

getTimeMaxMedicine() - Method in class Medicine

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

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

I

IAction - Interface in <Unnamed>

IDose - Interface in <Unnamed>

IMedicine - Interface in <Unnamed>

isTestDose - Variable in class Dose

Defines whether dose is a test dose Default: false.

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

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

L

loadFile(String) - Method in class Action

loadFile(String) - Method in interface IAction

Load a saved Medicine file.

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

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

M

main(String[]) - Static method in class Controller

medicine - Variable in class Action

Medicine file

Medicine - Class in <Unnamed>

Medicine() - Constructor for class Medicine

Medicine(String, LocalTime, LocalTime) - Constructor for class Medicine

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

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

P

pause() - Static method in class Controller

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

printConcentrationAtTime(LocalDateTime) - Method in class Action

printDetailedConcentrationAtTime(LocalDateTime) - Method in class Action

printDoses() - Method in class Action

printDoses() - Method in interface IAction

Prints all doses from Medicine's Dose array.

printMedicine() - Method in class Action

printMedicine() - Method in interface IAction

Print the name, tmax, and halfLife of the medicine.

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

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

R

removeAllDoses() - Method in class Action

removeAllDoses() - Method in interface IAction

Delete all doses from the dosages array.

removeAllDoses() - Method in interface IMedicine

Removes all doses from the doses array in the medicine

removeAllDoses() - Method in class Medicine

removeDose() - Static method in class Controller

Removes dose using an index selected by the user.

removeDose(int) - Method in class Action

removeDose(int) - Method in interface IAction

Remove a dose by its index.

removeDose(int) - Method in interface IMedicine

Remove a dose by its index.

removeDose(int) - Method in class Medicine

removeTestDoses() - Method in class Action

removeTestDoses() - Method in interface IAction

Removes all test doses.

removeTestDoses() - Method in interface IMedicine

Remove all test doses from the doses array

removeTestDoses() - Method in class Medicine

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

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

S

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

PACKAGE CLASS TREE DEPRECATED INDEX HELP

PREV LETTER NEXT LETTER FRAMES NO FRAMES ALL CLASSES

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

T

timeHalfLifeMedicine - Variable in class Medicine

Medicine half life.

timeMaxMedicine - Variable in class Medicine

Medicine tMax.

timezone - Variable in class Dose

Defines time zone.

toString() - Method in class Dose

toString() - Method in class Medicine

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

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

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

Constant Field Values

Contents

Serialized Form

Package <Unnamed>

Class 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.

timezone

```
java.time.ZoneOffset timezone
```

Defines time zone. Default: -05:00