

Disease Simulator

1.0

Generated by Doxygen 1.8.20

1 Namespace Index	1
1.1 Namespace List	1
2 Hierarchical Index	3
2.1 Class Hierarchy	3
3 Class Index	5
3.1 Class List	5
4 Namespace Documentation	7
4.1 Actor Namespace Reference	7
4.1.1 Detailed Description	7
4.2 IDisease Namespace Reference	7
4.2.1 Detailed Description	7
5 Class Documentation	9
5.1 Actor.Actor Class Reference	9
5.1.1 Constructor & Destructor Documentation	10
5.1.1.1 __init__()	10
5.1.2 Member Function Documentation	10
5.1.2.1 act()	10
5.1.2.2 addToWorld()	10
5.1.2.3 getID()	11
5.1.2.4 getWorld()	11
5.1.2.5 getX()	11
5.1.2.6 getY()	12
5.1.2.7 Iteration()	12
5.1.2.8 setLocation()	12
5.2 Disease.Disease Class Reference	13
5.2.1 Detailed Description	13
5.2.2 Constructor & Destructor Documentation	13
5.2.2.1 __init__()	14
5.2.3 Member Function Documentation	14
5.2.3.1 act()	14
5.2.3.2 getGrowthCondition()	14
5.2.3.3 getQuadrant()	15
5.2.3.4 getStrength()	15
5.2.3.5 setGrowthCondition()	15
5.3 IDisease.IDisease Class Reference	16
5.3.1 Detailed Description	16
5.3.2 Member Function Documentation	16
5.3.2.1 setGrowthCondition()	16
5.4 IWorld.IWorld Class Reference	17
5.4.1 Detailed Description	17

5.4.2 Member Function Documentation	18
5.4.2.1 getTemp()	18
5.4.2.2 initDiseases()	18
5.4.2.3 initGrowthConditions()	18
5.4.2.4 initLocations()	19
5.4.2.5 initTemps()	19
5.4.2.6 prepare()	19
5.4.2.7 setTemp()	20
5.5 MyWorld.MyWorld Class Reference	20
5.5.1 Detailed Description	21
5.5.2 Constructor & Destructor Documentation	21
5.5.2.1 __init__()	21
5.5.3 Member Function Documentation	21
5.5.3.1 act()	21
5.5.3.2 getTemp()	22
5.5.3.3 initDiseases()	22
5.5.3.4 initGrowthConditions()	23
5.5.4 initialization of the Disease growth conditions	23
5.5.5 No exceptions are thrown	23
5.5.5.1 initLocations()	23
5.5.5.2 initTemps()	24
5.5.5.3 prepare()	24
5.5.5.4 setTemp()	24
5.6 World.World Class Reference	25
5.6.1 Detailed Description	25
5.6.2 Constructor & Destructor Documentation	26
5.6.2.1 __init__()	26
5.6.3 Member Function Documentation	26
5.6.3.1 __repr__()	26
5.6.3.2 __str__()	27
5.6.3.3 act()	27
5.6.3.4 addObject()	27
5.6.3.5 createGrid()	28
5.6.3.6 getDepth()	28
5.6.3.7 getHeight()	29
5.6.3.8 getObjects()	29
5.6.3.9 getWidth()	29
5.6.3.10 numberOfObjects()	30
5.6.3.11 setGrid()	30
Index	31

Chapter 1

Namespace Index

1.1 Namespace List

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

Actor	Actor class, which is the base class for Disease objects	7
IDisease	Disease Interface	7

Chapter 2

Hierarchical Index

2.1 Class Hierarchy

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

Actor.Actor	9
Disease.Disease	13
World.World	25
MyWorld.MyWorld	20
ABC	
IDisease.IDisease	16
Disease.Disease	13
IWorld.IWorld	17
MyWorld.MyWorld	20

Chapter 3

Class Index

3.1 Class List

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

Actor.Actor	9
Disease.Disease	
This Disease class is a sub-class of the Actor class	13
IDisease.IDisease	
Interface IDisease allows setting the strength and growth condition of a disease	16
IWorld.IWorld	
Interface IWorld allows initializing and setting diseases for a world	17
MyWorld.MyWorld	
Call the constructor of the World class with the width and height of 720 and 640 cells , respectively	20
World.World	
Class for holding Actor objects in cells of a grid in the world	25

Chapter 4

Namespace Documentation

4.1 Actor Namespace Reference

[Actor](#) class, which is the base class for Disease objects.

Classes

- class [Actor](#)

4.1.1 Detailed Description

[Actor](#) class, which is the base class for Disease objects.

Author

Viviane Magalhães Siqueira (based on Professor Paulo Roma's work)

Date

10/08/2020

4.2 IDisease Namespace Reference

Disease Interface.

Classes

- class [IDisease](#)
Interface [IDisease](#) allows setting the strength and growth condition of a disease.

Variables

- **ABC** = object

4.2.1 Detailed Description

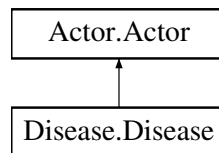
Disease Interface.

Chapter 5

Class Documentation

5.1 Actor.Actor Class Reference

Inheritance diagram for Actor.Actor:



Public Member Functions

- def `__init__` (self)
Construct a new [Actor](#) object.
- def `getID` (self)
Used for testing.
- def `Iteration` (self)
Used for testing.
- def `act` (self)
Prints on screen in the format "Iteration <ID>: Actor <Actor ID>".
- def `setLocation` (self, x, y)
Sets the cell coordinates of this object.
- def `addedToWorld` (self, world)
Sets the world this actor is into.
- def `getWorld` (self)
Gets the world this object is into.
- def `getX` (self)
Gets the X coordinate of the cell this actor object is into.
- def `getY` (self)
Gets the Y coordinate of the cell this actor object is into.
- def `__str__` (self)
Return a string with this actor ID and position.

5.1.1 Constructor & Destructor Documentation

5.1.1.1 `__init__()`

```
def Actor.Actor.__init__ (
    self )
```

Construct a new [Actor](#) object.

- Sets the initial values of its member variables.
- Sets the unique ID for the object and initializes the reference to the World object to which this [Actor](#) object belongs to null.
- The ID of the first [Actor](#) object is 0.
- The ID gets incremented by one each time a new [Actor](#) object is created.
- Sets the iteration counter to zero and initialize the location of the object to cell (0,0).

Reimplemented in [Disease.Disease](#).

5.1.2 Member Function Documentation

5.1.2.1 `act()`

```
def Actor.Actor.act (
    self )
```

Prints on screen in the format "Iteration <ID>: Actor <Actor ID>".

The < *ID* > is replaced by the current iteration number. < *ActorID* > is replaced by the unique ID of the [Actor](#) object that performs the `act(self)` method.

For instance, the actor with ID 1 shows the following result on the output screen after its `act(self)` method has been called twice.

```
Iteration 0: Actor 1
Iteration 1: Actor 1
```

Reimplemented in [Disease.Disease](#).

5.1.2.2 `addedToWorld()`

```
def Actor.Actor.addedToWorld (
    self,
    world )
```

Sets the world this actor is into.

Parameters

<i>world</i>	Reference to the World object this Actor object is added.
--------------	---

Exceptions

<i>RuntimeError</i>	when world is null.
---------------------	---------------------

5.1.2.3 getID()

```
def Actor.Actor.getID (
    self )
```

Used for testing.

Returns

ActorID

5.1.2.4 getWorld()

```
def Actor.Actor.getWorld (
    self )
```

Gets the world this object in into.

Returns

the world this object belongs to

5.1.2.5 getX()

```
def Actor.Actor.getX (
    self )
```

Gets the X coordinate of the cell this actor object is into.

Returns

the x coordinate of this [Actor](#) object.

5.1.2.6 getY()

```
def Actor.Actor.getY (
    self )
```

Gets the Y coordinate of the cell this actor object is into.

Returns

the y coordinate of this [Actor](#) object.

5.1.2.7 Iteration()

```
def Actor.Actor.Iteration (
    self )
```

Used for testing.

Returns

number of iterations

5.1.2.8 setLocation()

```
def Actor.Actor.setLocation (
    self,
    x,
    y )
```

Sets the cell coordinates of this object.

Parameters

<i>x</i>	the column.
<i>y</i>	the row.

Exceptions

<i>ValueError</i>	when $x < 0$ or $x \geq$ world width,
<i>ValueError</i>	when $y < 0$ or $y \geq$ world height,
<i>RuntimeError</i>	when the world is null.

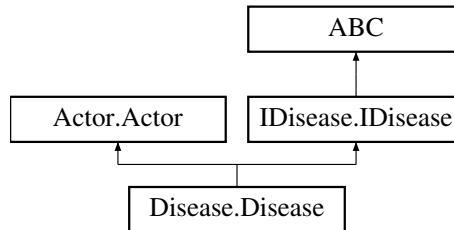
The documentation for this class was generated from the following file:

- C:/TI-Faculdade/PIG/AD1/Actor.py

5.2 Disease.Disease Class Reference

This [Disease](#) class is a sub-class of the [Actor](#) class.

Inheritance diagram for Disease.Disease:



Public Member Functions

- `def __init__ (self)`
Constructor.
- `def setGrowthCondition (self, lTemp, hTemp, gRate)`
Sets the disease growth rate, lower temperature and higher temperature.
- `def getGrowthCondition (self)`
Returns the disease growth rate, lower temperature and higher temperature.
- `def getQuadrant (self)`
Returns the quadrant of this disease.
- `def getStrength (self)`
Return the disease strength of this object.
- `def act (self)`
This method overrides the act () method in the [Actor](#) class .
- `def __str__ (self)`
Return a string with the strength, growth and quadrant of this disease.

5.2.1 Detailed Description

This [Disease](#) class is a sub-class of the [Actor](#) class.

Author

Viviane Magalhães Siqueira (based on Professor Paulo Roma's work)

Date

10/08/2020

5.2.2 Constructor & Destructor Documentation

5.2.2.1 `__init__()`

```
def Disease.Disease.__init__ (
    self )
```

Constructor.

- Call its superclass's default constructor.
- Initialize the lower bound and the upper bound temperatures for the growth rate to 0.
- Set the growth rate to 0.
- Set the disease strength to 1.

Reimplemented from [Actor.Actor](#).

5.2.3 Member Function Documentation

5.2.3.1 `act()`

```
def Disease.Disease.act (
    self )
```

This method overrides the `act ()` method in the [Actor](#) class .

Check whether the object is in the region where the region temperature is within the lower bound and the upper bound temperatures for the object 's growth rate . If it is the case , multiply its strength with the growth rate .

Reimplemented from [Actor.Actor](#).

5.2.3.2 `getGrowthCondition()`

```
def Disease.Disease.getGrowthCondition (
    self )
```

Returns the disease growth rate, lower temperature and higher temperature.

Returns

growth rate, lower temp and higher temp

5.2.3.3 getQuadrant()

```
def Disease.Disease.getQuadrant (
    self )
```

Returns the quadrant of this disease.

Returns

0, 1, 2 or 3.

5.2.3.4 getStrength()

```
def Disease.Disease.getStrength (
    self )
```

Return the disease strength of this object.

Returns

disease strength of the object.

Reimplemented from [IDisease.IDisease](#).

5.2.3.5 setGrowthCondition()

```
def Disease.Disease.setGrowthCondition (
    self,
    lTemp,
    hTemp,
    gRate )
```

Sets the disease growth rate, lower temperature and higher temperature.

Parameters

<i>lTemp</i>	Lower bound temperature for the disease to grow at this gRate.
<i>hTemp</i>	Upper bound temperature for the disease to grow at this gRate.
<i>gRate</i>	The growth rate.

Reimplemented from [IDisease.IDisease](#).

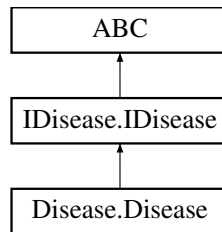
The documentation for this class was generated from the following file:

- C:/TI-Faculdade/PIG/AD1/Disease.py

5.3 IDisease.IDisease Class Reference

Interface [IDisease](#) allows setting the strength and growth condition of a disease.

Inheritance diagram for IDisease.IDisease:



Public Member Functions

- def [setGrowthCondition](#) (self, lTemp, hTemp, gRate)
Set the growth condition of a Disease object to gRate.
- def [getStrength](#) (self)
Return the disease strength of the object implements this interface.

5.3.1 Detailed Description

Interface [IDisease](#) allows setting the strength and growth condition of a disease.

Author

Paulo Cavalcanti

Date

22/02/2020

5.3.2 Member Function Documentation

5.3.2.1 setGrowthCondition()

```
def IDisease.IDisease.setGrowthCondition (
    self,
    lTemp,
    hTemp,
    gRate )
```

Set the growth condition of a Disease object to gRate.

The value of gRate gets multiplied to the current disease strength only when the disease is located in the world region with the average temperature in between the values of lTemp and hTemp.

Reimplemented in [Disease.Disease](#).

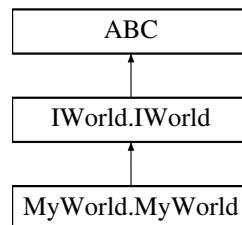
The documentation for this class was generated from the following file:

- C:/TI-Faculdade/PIG/AD1/IDisease.py

5.4 IWorld.IWorld Class Reference

Interface [IWorld](#) allows initializing and setting diseases for a world.

Inheritance diagram for IWorld.IWorld:



Public Member Functions

- def [prepare](#) (self)
Prepare the world .
- def [setTemp](#) (self, quad, temp)
Set the temperature of the region of the world to the value of temp .
- def [getTemp](#) (self, quad)
Return the temperature of the world region with the ID of quadID .
- def [initDiseases](#) (self, numDisStr)
Create Disease objects ; the number of the objects equals to the value passed in numDisStr .
- def [getObjects](#) (self)
Return the list of objects in the class implementing this interface .
- def [initTemps](#) (self, tempStr)
Set the temperature for each quadrant of the MyWorld according to the value of the tempStr .
- def [getSumStrength](#) (self)
Return the total disease strength of all the diseases in the class implementing this interface .
- def [initLocations](#) (self, locationsStr, diseaseArr)
Add each Disease object into the MyWorld object implementing this method according to the information in locationStr .
- def [initGrowthConditions](#) (self, growthStr, diseaseArr)
Set the lower bound and upper bound temperature and the growth rate for each disease according to the input growthStr .

5.4.1 Detailed Description

Interface [IWorld](#) allows initializing and setting diseases for a world.

Author

Viviane Magalhães Siqueira (based on Professor Paulo Roma's work)

Date

10/08/2020

5.4.2 Member Function Documentation

5.4.2.1 getTemp()

```
def IWorld.IWorld.getTemp (
    self,
    quad )
```

Return the temperature of the world region with the ID of quadID .

The valid value is between zero and three inclusive .

Reimplemented in [MyWorld.MyWorld](#).

5.4.2.2 initDiseases()

```
def IWorld.IWorld.initDiseases (
    self,
    numDisStr )
```

Create Disease objects ; the number of the objects equals to the value passed in numDisStr .

Return a list of object references to the created Disease objects .

An example of a valid numDisStr is below.

Ex: "2"

If numDisStr is None or it cannot be converted to a positive integer , print a message on screen " Check the NumDiseases line in simulation . config ." and return None .

No exceptions are thrown.

Reimplemented in [MyWorld.MyWorld](#).

5.4.2.3 initGrowthConditions()

```
def IWorld.IWorld.initGrowthConditions (
    self,
    growthStr,
    diseaseArr )
```

Set the lower bound and upper bound temperature and the growth rate for each disease according to the input growthStr .

An example of a valid string for two Disease objects is:

Ex: "10.0 ,15.0 ,2.0;10.0 ,13.0 ,3.0"

If growthStr is empty or not in the correct format or does not have all the growth for all the Disease objects in the Disease array , print on screen " Check the DiseasesGrowth line in simulation . config ." and return -1.

Return 0 for a successful initialization of the Disease growth conditions . No exceptions are thrown .

Reimplemented in [MyWorld.MyWorld](#).

5.4.2.4 initLocations()

```
def IWorld.IWorld.initLocations (
    self,
    locationsStr,
    diseaseArr )
```

Add each Disease object into the MyWorld object implementing this method according to the information in locationStr .

An example of a locationStr is "200 ,200;400 ,480". This means that the first Disease is planted at cell (200 ,200) and the second Disease is at cell (400 , 480).

If the locationStr is empty or not in the correct format or does not have all the cell coordinates of all the Disease objects , print on screen " Check the Locations line in simulation . config " and return -1.

Return 0 for a successful initialization of the Disease locations . No exceptions are thrown .

Reimplemented in [MyWorld.MyWorld](#).

5.4.2.5 initTemps()

```
def IWorld.IWorld.initTemps (
    self,
    tempStr )
```

Set the temperature for each quadrant of the MyWorld according to the value of the tempStr .

An example of tempStr is below . The region temperatures for regions 0, 1, 2, and 3 are 12, 20, 50, and 100 , respectively .

Return 0 for a successful initialization of the quadrant temperatures . No exceptions are thrown . the quadrant temperatures . No exceptions are thrown .

Ex: "12;20;50;100"

If tempStr is empty or not in the correct format or does not have all the temperatures of all the regions , print on screen " Check the Temperature line in simulation . config ." and return -1.

Reimplemented in [MyWorld.MyWorld](#).

5.4.2.6 prepare()

```
def IWorld.IWorld.prepare (
    self )
```

Prepare the world .

Open a text file named " simulation . config " in the current path (directly under the project directory). Parse the configuration file for the number of Disease objects , the cell locations of these objects , the growth rates , and the temperature ranges associated with individual growth rates . Read Section 4 on the content of the configuration file before reading the rest .

Reimplemented in [MyWorld.MyWorld](#).

5.4.2.7 setTemp()

```
def IWorld.IWorld.setTemp (
    self,
    quad,
    temp )
```

Set the temperature of the region of the world to the value of temp .

The quadID indicates the region . The valid value is between [0, 3]. Any value of float is accepted for temp .

Reimplemented in [MyWorld.MyWorld](#).

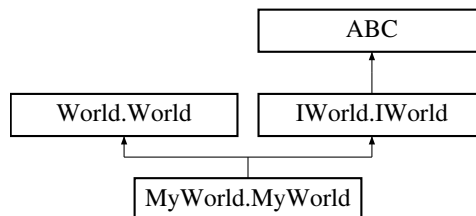
The documentation for this class was generated from the following file:

- C:/TI-Faculdade/PIG/AD1/IWorld.py

5.5 MyWorld.MyWorld Class Reference

Call the constructor of the World class with the width and height of 720 and 640 cells , respectively .

Inheritance diagram for MyWorld.MyWorld:



Public Member Functions

- def [__init__](#) (self, width=720, height=640)
Constructor.
- def [prepare](#) (self)
Prepare the world .
- def [getObjects](#) (self)
Return the list of objects in the class implementing this interface .
- def [initDiseases](#) (self, numDisStr)
Create Disease objects ; the number of the objects equals to the value passed in numDisStr .
- def [initLocations](#) (self, locationsStr, diseaseArr)
Add each Disease object into the [MyWorld](#) object implementing this method according to the information in locationStr .
- def [initGrowthConditions](#) (self, growthStr, diseaseArr)
Set the lower bound and upper bound temperature and the growth rate for each disease according to the input growthStr .
- def [initTemps](#) (self, tempStr)
Set the temperature for each quadrant of the [MyWorld](#) according to the value of the tempStr .
- def [getSumStrength](#) (self)
Return the total disease strength of all the diseases in the class implementing this interface .
- def [getTemp](#) (self, quad)
Return the temperature of the world region with the ID of quadID .
- def [setTemp](#) (self, quad, temp)
Set the temperature of the region of the world to the value of temp .
- def [act](#) (self)
This method overrides the act () method in the World class .

5.5.1 Detailed Description

Call the constructor of the World class with the width and height of 720 and 640 cells , respectively .

Initialize a list to keep the average temperature of each world region (quadrant).

Call the prepare () method .

5.5.2 Constructor & Destructor Documentation

5.5.2.1 __init__()

```
def MyWorld.MyWorld.__init__ (
    self,
    worldWidth = 720,
    worldHeight = 640 )
```

Constructor.

Creates a world with the given width and height.

- The maximum width and height are 1000.
- The maximum number of [Actor](#) objects in a cell is 5.

```
If worldWidth <= 0 or worldWidth > maximum width
    use the maximum width instead.
If worldHeight <=0 or worldHeight > maximum height
    use the maximum height instead.
```

Parameters

<i>worldWidth</i>	Width in number of cells
<i>worldHeight</i>	Height in number of cells

Reimplemented from [World.World](#).

5.5.3 Member Function Documentation

5.5.3.1 act()

```
def MyWorld.MyWorld.act (
    self )
```

This method overrides the `act ()` method in the `World` class .

This method prints :

" Iteration <ITRID >: World disease strength is <WorldDisease >" where <ITRID > is replaced by the current iteration number and <WorldDisease > is replaced by the returned value of `getSumStrength ()` in 2 decimal places . An example is below .

Iteration 0: World disease strength is 2.00 Iteration 1: World disease strength is 3.00

Reimplemented from [World.World](#).

5.5.3.2 `getTemp()`

```
def MyWorld.MyWorld.getTemp (
    self,
    quad )
```

Return the temperature of the world region with the ID of `quadID` .

The valid value is between zero and three inclusive .

Reimplemented from [IWorld.IWorld](#).

5.5.3.3 `initDiseases()`

```
def MyWorld.MyWorld.initDiseases (
    self,
    numDisStr )
```

Create Disease objects ; the number of the objects equals to the value passed in `numDisStr` .

Return a list of object references to the created Disease objects .

An example of a valid `numDisStr` is below.

Ex: "2"

If `numDisStr` is `None` or it cannot be converted to a positive integer , print a message on screen " Check the NumDiseases line in simulation . config ." and return `None` .

No exceptions are thrown.

Reimplemented from [IWorld.IWorld](#).

5.5.3.4 initGrowthConditions()

```
def MyWorld.MyWorld.initGrowthConditions (
    self,
    growthStr,
    diseaseArr )
```

Set the lower bound and upper bound temperature and the growth rate for each disease according to the input growthStr .

An example of a valid string for two Disease objects is:

Ex: "10.0 ,15.0 ,2.0;10.0 ,13.0 ,3.0"

If growthStr is empty or not in the correct format or does not have all the growth for all the Disease objects in the Disease array , print on screen "Check the DiseasesGrowth line in simulation.config." and return -1.

Return 0 for a successful

5.5.4 initialization of the Disease growth conditions .

5.5.5 No exceptions are thrown .

Reimplemented from [IWorld.IWorld](#).

5.5.5.1 initLocations()

```
def MyWorld.MyWorld.initLocations (
    self,
    locationsStr,
    diseaseArr )
```

Add each Disease object into the [MyWorld](#) object implementing this method according to the information in locationStr .

An example of a locationStr is "200 ,200;400 ,480". This means that the first Disease is planted at cell (200 ,200) and the second Disease is at cell (400 , 480).

If the locationStr is empty or not in the correct format or does not have all the cell coordinates of all the Disease objects , print on screen " Check the Locations line in simulation . config " and return -1.

Return 0 for a successful initialization of the Disease locations . No exceptions are thrown .

Reimplemented from [IWorld.IWorld](#).

5.5.5.2 `initTemps()`

```
def MyWorld.MyWorld.initTemps (
    self,
    tempStr )
```

Set the temperature for each quadrant of the [MyWorld](#) according to the value of the tempStr .

An example of tempStr is below . The region temperatures for regions 0, 1, 2, and 3 are 12, 20, 50, and 100 , respectively .

Reimplemented from [IWorld.IWorld](#).

5.5.5.3 `prepare()`

```
def MyWorld.MyWorld.prepare (
    self )
```

Prepare the world .

Open a text file named " simulation . config " in the current path (directly under the project directory). Parse the configuration file for the number of Disease objects , the cell locations of these objects , the growth rates , and the temperature ranges associated with individual growth rates . Read Section 4 on the content of the configuration file before reading the rest .

Reimplemented from [IWorld.IWorld](#).

5.5.5.4 `setTemp()`

```
def MyWorld.MyWorld.setTemp (
    self,
    quad,
    temp )
```

Set the temperature of the region of the world to the value of temp .

The quadID indicates the region . The valid value is between [0, 3]. Any value of float is accepted for temp .

Reimplemented from [IWorld.IWorld](#).

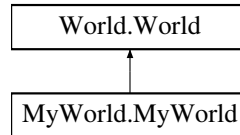
The documentation for this class was generated from the following file:

- C:/TI-Faculdade/PIG/AD1/MyWorld.py

5.6 World.World Class Reference

Class for holding [Actor](#) objects in cells of a grid in the world.

Inheritance diagram for World.World:



Public Member Functions

- `def __init__ (self, worldWidth, worldHeight)`
Constructor.
- `def createGrid (self, h, w, d)`
Initializes each object of the array as None.
- `def __str__ (self)`
Return a string representation of the grid.
- `def __repr__ (self)`
Return a string representation of the grid.
- `def act (self)`
Blank method body.
- `def addObject (self, object, x, y)`
Adds a new actor to this world at a given position.
- `def getHeight (self)`
Returns the world height.
- `def getWidth (self)`
Returns the world width.
- `def getDepth (self)`
Returns the world depth.
- `def numberOfObjects (self)`
Returns the total number of objects in this world.
- `def getObjects (self)`
Returns an array with all [Actor](#) objects in this world.
- `def setGrid (self, aGrid, numObjs)`
It checks if aGrid is a 3D array with the same positive length in each dimension.

5.6.1 Detailed Description

Class for holding [Actor](#) objects in cells of a grid in the world.

The world is represented by a 2 dimensional array of cells, with the specified width and height. One cell can keep at most 5 [Actor](#) objects.

Author

Viviane Magalhães Siqueira (based on Professor Paulo Roma's work)

Date

10/08/2020

5.6.2 Constructor & Destructor Documentation

5.6.2.1 `__init__()`

```
def World.World.__init__ (
    self,
    worldWidth,
    worldHeight )
```

Constructor.

Creates a world with the given width and height.

- The maximum width and height are 1000.
- The maximum number of [Actor](#) objects in a cell is 5.

If worldWidth <= 0 or worldWidth > maximum width
use the maximum width instead.
If worldHeight <=0 or worldHeight > maximum height
use the maximum height instead.

Parameters

<i>worldWidth</i>	Width in number of cells
<i>worldHeight</i>	Height in number of cells

Reimplemented in [MyWorld.MyWorld](#).

5.6.3 Member Function Documentation

5.6.3.1 `__repr__()`

```
def World.World.__repr__ (
    self )
```

Return a string representation of the grid.

List by depth. Each slice is height x width.

Returns

string with the grid.

See also

<https://www.ict.social/python/basics/multidimensional-lists-in-python>

5.6.3.2 `__str__()`

```
def World.World.__str__ (
    self )
```

Return a string representation of the grid.

List by width. Each slice is height x depth. (List by height. Each slice is width x depth)

Returns

string with the grid.

5.6.3.3 `act()`

```
def World.World.act (
    self )
```

Blank method body.

Overriden in subclasses as appropriate

Reimplemented in [MyWorld.MyWorld](#).

5.6.3.4 `addObject()`

```
def World.World.addObject (
    self,
    object,
    x,
    y )
```

Adds a new actor to this world at a given position.

- The new object will be added at the cell (x,y) if there are less than 5 objects in this cell.
- Be sure to make the added object know that it is in this world and it is at this cell.
- Check which methods of the [Actor](#) class to call.

Parameters

<i>object</i>	the object to be added at this cell (x, y)
<i>x</i>	the column
<i>y</i>	the row

Returns

number of objects in cell (x,y).

Exceptions

<i>SyntaxError</i>	when already max number of objects are in that cell
<i>ValueError</i>	if x or y is not in the valid range
<i>NameError</i>	if the object is null

5.6.3.5 createGrid()

```
def World.World.createGrid (
    self,
    h,
    w,
    d )
```

Initializes each object of the array as None.

Parameters

<i>h</i>	grid height.
<i>w</i>	grid width.
<i>d</i>	grid depth.

Returns

grid.

5.6.3.6 getDepth()

```
def World.World.getDepth (
    self )
```

Returns the world depth.

Returns

the world depth.

5.6.3.7 getHeight()

```
def World.World.getHeight (
    self )
```

Returns the world height.

Returns

the world height.

5.6.3.8 getObjects()

```
def World.World.getObjects (
    self )
```

Returns an array with all [Actor](#) objects in this world.

Returns

Array of [Actor](#) objects that are in this world.

Comments:

- Each class in Java is a subclass of the Object class.
- Observe that you use the implicit upcast where you assign an [Actor](#) object (sub-class) in an element of the Object array.

Reimplemented in [MyWorld.MyWorld](#).

5.6.3.9 getWidth()

```
def World.World.getWidth (
    self )
```

Returns the world width.

Returns

the world width.

5.6.3.10 numberOfObjects()

```
def World.World.numberOfObjects (
    self )
```

Returns the total number of objects in this world.

Returns

Total number of objects in this world.

5.6.3.11 setGrid()

```
def World.World.setGrid (
    self,
    aGrid,
    numObjs )
```

It checks if aGrid is a 3D array with the same positive length in each dimension.

If so, it sets the grid to aGrid and the other private fields of class [World](#) to the dimension lengths of aGrid and numObjs.

Note that some checks are omitted. For example, no check is performed to make sure that numObjs is consistent with the number of [Actor](#) objects in aGrid.

Each [Actor](#) object in aGrid has to be set to this [World](#) object.

Parameters

<i>aGrid</i>	reference to a 3D array of Actor objects.
<i>numObjs</i>	the number of Actor objects in aGrid.

Exceptions

<i>ValueError</i>	if the length of each dimension is out of range or 2nd/3rd dimension has different lengths.
-------------------	---

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

- C:/TI-Faculdade/PIG/AD1/World.py

Index

- `__init__`
 - Actor.Actor, 10
 - Disease.Disease, 13
 - MyWorld.MyWorld, 21
 - World.World, 26
 - `__repr__`
 - World.World, 26
 - `__str__`
 - World.World, 26
- act
 - Actor.Actor, 10
 - Disease.Disease, 14
 - MyWorld.MyWorld, 21
 - World.World, 27
- Actor, 7
 - Actor.Actor, 9
 - `__init__`, 10
 - act, 10
 - addedToWorld, 10
 - getID, 11
 - getWorld, 11
 - getX, 11
 - getY, 11
 - Iteration, 12
 - setLocation, 12
- addedToWorld
 - Actor.Actor, 10
- addObject
 - World.World, 27
- createGrid
 - World.World, 28
- Disease.Disease, 13
 - `__init__`, 13
 - act, 14
 - getGrowthCondition, 14
 - getQuadrant, 14
 - getStrength, 15
 - setGrowthCondition, 15
- getDepth
 - World.World, 28
- getGrowthCondition
 - Disease.Disease, 14
- getHeight
 - World.World, 28
- getID
 - Actor.Actor, 11
- getObjects
 - World.World, 29
- getQuadrant
 - Disease.Disease, 14
- getStrength
 - Disease.Disease, 15
- getTemp
 - IWorld.IWorld, 18
 - MyWorld.MyWorld, 22
- getWidth
 - World.World, 29
- getWorld
 - Actor.Actor, 11
- getX
 - Actor.Actor, 11
- getY
 - Actor.Actor, 11
- IDisease, 7
 - IDisease.IDisease, 16
 - setGrowthCondition, 16
- initDiseases
 - IWorld.IWorld, 18
 - MyWorld.MyWorld, 22
- initGrowthConditions
 - IWorld.IWorld, 18
 - MyWorld.MyWorld, 22
- initLocations
 - IWorld.IWorld, 18
 - MyWorld.MyWorld, 23
- initTemps
 - IWorld.IWorld, 19
 - MyWorld.MyWorld, 23
- Iteration
 - Actor.Actor, 12
- IWorld.IWorld, 17
 - getTemp, 18
 - initDiseases, 18
 - initGrowthConditions, 18
 - initLocations, 18
 - initTemps, 19
 - prepare, 19
 - setTemp, 19
- MyWorld.MyWorld, 20
 - `__init__`, 21
 - act, 21
 - getTemp, 22
 - initDiseases, 22
 - initGrowthConditions, 22

- initLocations, [23](#)
- initTemps, [23](#)
- prepare, [24](#)
- setTemp, [24](#)

numberOfObjects

- World.World, [29](#)

prepare

- IWorld.IWorld, [19](#)
- MyWorld.MyWorld, [24](#)

setGrid

- World.World, [30](#)

setGrowthCondition

- Disease.Disease, [15](#)
- IDisease.IDisease, [16](#)

setLocation

- Actor.Actor, [12](#)

setTemp

- IWorld.IWorld, [19](#)
- MyWorld.MyWorld, [24](#)

World.World, [25](#)

- __init__, [26](#)
- __repr__, [26](#)
- __str__, [26](#)
- act, [27](#)
- addObject, [27](#)
- createGrid, [28](#)
- getDepth, [28](#)
- getHeight, [28](#)
- getObjects, [29](#)
- getWidth, [29](#)
- numberOfObjects, [29](#)
- setGrid, [30](#)