AD1: Programming with Graphical Interfaces

1.0.0

Jefferson Peralva Machiqueira

Generated by Doxygen 1.8.16

Chapter 1

Hierarchical Index

1.1 Class Hierarchy

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

Chapter 2

Class Index

2.1 Class List

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

Chapter 3

File Index

3.1 File List

Here is a list of all files with brief descriptions:

Chapter 4

Namespace Documentation

4.1 Actor Namespace Reference

Classes

- class Actor

  Actor class, which is the base class for Disease objects.

4.1.1 Detailed Description

Author

Jefferson Peralva Machiqueira

Date

31/08/2020

4.2 ActorTest Namespace Reference

Classes

- class ActorTest

  Class for testing Actor.Actor class.

Variables

- verbosity

4.2.1 Variable Documentation

4.2.1.1 verbosity

```
ActorTest.verbosity
```

4.3 Constants Namespace Reference

Variables

- int MAXIMUM_WIDTH = 1000

  Constant with the maximum grid width.
- int MAXIMUM_HEIGHT = 1000

  Constant with the maximum grid height.

4.3.1 Variable Documentation

4.3.1.1 MAXIMUM_HEIGHT

```
int Constants.MAXIMUM_HEIGHT = 1000
```

Constant with the maximum grid height.

4.3.1.2 MAXIMUM_WIDTH

```
int Constants.MAXIMUM_WIDTH = 1000
```

Constant with the maximum grid width.

4.4 Disease Namespace Reference

Classes

- class Disease

  This Disease class is a sub-class of the Actor class.

4.4.1 Detailed Description

Author

Jefferson Peralva Machiqueira

Date

31/08/2020

4.5 DiseaseTest Namespace Reference

Classes

- class ActorTest

  Class for testing Disease.Disease class.

Variables

- verbosity

4.5.1 Variable Documentation

4.5.1.1 verbosity

```
DiseaseTest.verbosity
```

4.6 IDisease Namespace Reference

Classes

- class IDisease

  Interface IDisease allows setting the strength and growth condition of a disease.

Variables

- ABC = object

4.6.1 Detailed Description

Author

Jefferson Peralva Machiqueira

Date

31/08/2020

4.6.2 Variable Documentation

4.6.2.1 ABC

```
IDisease.ABC = object
```

4.7 IWorld Namespace Reference

Classes

- class IWorld

  Interface IWorld allows initializing and setting diseases for a world.

Variables

- ABC = object

4.7.1 Detailed Description

Author

Jefferson Peralva Machiqueira

Date

31/08/2020

4.7.2 Variable Documentation

4.7.2.1 ABC

```
IWorld.ABC = object
```

4.8 MyWorld Namespace Reference

Classes

- class MyWorld

  SubClass of World and IWorld classes.

Variables

- ArrayDisease = List[Disease]

  Type definition for use in Python Type Hinting for ArrayDisease/list-of-disease-instances.
- valor = MyWorld(720, 640)
- objetos = valor.getObjects()

4.8.1 Detailed Description

Author

Jefferson Peralva Machiqueira

Date

31/08/2020

4.8.2 Variable Documentation

4.8.2.1 ArrayDisease

```
MyWorld.ArrayDisease = List[Disease]
```

Type definition for use in Python Type Hinting for ArrayDisease/list-of-disease-instances.

4.8.2.2 objetos

```
MyWorld.objetos = valor.getObjects()
```

4.8.2.3 valor

```
MyWorld.valor = MyWorld(720, 640)
```

4.9 simulator Namespace Reference

Functions

- def main (args=None)

  This is the main method that sets up a virtual world and simulates the gowth of the diseases in the world if the number of iterations is given in the comand line argument, run the simulation for that number of iterations Otherwise, use the deafault number of iterations: 5.

4.9.1 Detailed Description

Author

Jefferson Peralva Machiqueira

Date

31/08/2020

4.9.2 Function Documentation

4.9.2.1 main()

```
def simulator.main (
              args = None )
```

This is the main method that sets up a virtual world and simulates the gowth of the diseases in the world if the number of iterations is given in the comand line argument, run the simulation for that number of iterations Otherwise, use the deafault number of iterations: 5.

Author

Jefferson Peralva Machiqueira

Date

31/08/2020

4.10 World Namespace Reference

Classes

- class World

  Class for holding Actor objects in cells of a grid in the world.

Functions

- def main ()

Variables

- Grid = List[List[List[int or None or Actor]]]

  Type definition for use in Python Type Hinting for Grid/3D-list.
- ArrayActor = List[Actor]

  Type definition for use in Python Type Hinting for ArrayActor/list-of-actor-instances.

4.10.1 Detailed Description

Author

Jefferson Peralva Machiqueira

Date

31/08/2020

4.10.2 Function Documentation

4.10.2.1 main()

```
def World.main ( )
```

4.10.3 Variable Documentation

4.10.3.1 ArrayActor

```
World.ArrayActor = List[Actor]
```

Type definition for use in Python Type Hinting for ArrayActor/list-of-actor-instances.

4.10.3.2 Grid

```
World.Grid = List[List[List[int or None or Actor]]]
```

Type definition for use in Python Type Hinting for Grid/3D-list.

4.11 WorldTest Namespace Reference

Classes

- class WorldTest

  Class for testing World.World class.

Variables

- verbosity

4.11.1 Detailed Description

Author

Jefferson Peralva Machiqueira

Date

31/08/2020

4.11.2 Variable Documentation

4.11.2.1 verbosity

`WorldTest.verbosity`

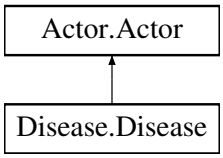
Chapter 5

Class Documentation

5.1 Actor.Actor Class Reference

Actor class, which is the base class for Disease objects.

Inheritance diagram for Actor.Actor:



Public Member Functions

- def __init__ (self)

  Construct a new Actor object.
- int getID (self)

  Used for testing.
- int Iteration (self)

  Used for testing.
- def act (self)

  Prints on screen in the format "Iteration <ID>: Actor <Actor ID>".
- def setLocation (self, int x, int y)

  Sets the cell coordinates of this object.
- def addedToWorld (self, object world)

  Sets the world this actor is into.
- object getWorld (self)

  Gets the world this object in into.
- int getX (self)

  Gets the X coordinate of the cell this actor object is into.
- int getY (self)

  Gets the Y coordinate of the cell this actor object is into.
- def nextIteration (self)

  Jumps for next iteration.
- str __str__ (self)

  Return a string with this actor ID and position.

Private Attributes

- __locX
  X coordinate of this actor.
- __locY
  Y coordinate of this actor.
- __world
  World this actor belongs to.
- __actorID
  Unique identifier for this actor.
- __itCounter
  Iteration counter.
- __worldWidth
  World width.
- __worldHeight
  World height.

Static Private Attributes

- int __ID = 0
  Holds the value of the next "free" id.

5.1.1 Detailed Description

Actor class, which is the base class for Disease objects.

Author

Jefferson Peralva Machiqueira

5.1.2 Constructor & Destructor Documentation

5.1.2.1 __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.

5.1.3 Member Function Documentation

5.1.3.1 __str__()

```
str Actor.Actor.__str__ (
            self )
```

Return a string with this actor ID and position.

Reimplemented in Disease.Disease.

5.1.3.2 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.

5.1.3.3 addedToWorld()

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

Sets the world this actor is into.

Parameters

world	Reference to the World object this Actor object is added.

Exceptions

RuntimeError	when world is null.

5.1.3.4 getID()

```
int Actor.Actor.getID (
              self )
```

Used for testing.

Returns

ActorID

5.1.3.5 getWorld()

```
object Actor.Actor.getWorld (
              self )
```

Gets the world this object in into.

Returns

the world this object belongs to

5.1.3.6 getX()

```
int Actor.Actor.getX (
              self )
```

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

Returns

the x coordinate of this Actor object.

5.1.3.7 getY()

```
int Actor.Actor.getY (
              self )
```

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

Returns

the y coordinate of this Actor object.

5.1.3.8 Iteration()

```
int Actor.Actor.Iteration (
              self )
```

Used for testing.

Returns

number of iterations

5.1.3.9 nextIteration()

```
def Actor.Actor.nextIteration (
              self )
```

Jumps for next iteration.

5.1.3.10 setLocation()

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

Sets the cell coordinates of this object.

Parameters

x	the column.
y	the row.

Exceptions

ValueError	when x $<$ 0 or x $>$= world width,
ValueError	when y $<$ 0 or y $>$= world height,
RuntimeError	when the world is null.

5.1.4 Member Data Documentation

5.1.4.1 __actorID

`Actor.Actor.__actorID [private]`

Unique identifier for this actor.

5.1.4.2 __ID

`int Actor.Actor.__ID = 0 [static], [private]`

Holds the value of the next "free" id.

5.1.4.3 __itCounter

`Actor.Actor.__itCounter [private]`

Iteration counter.

5.1.4.4 __locX

`Actor.Actor.__locX [private]`

X coordinate of this actor.

5.1.4.5 __locY

`Actor.Actor.__locY [private]`

Y coordinate of this actor.

5.1.4.6 __world

`Actor.Actor.__world [private]`

World this actor belongs to.

5.1.4.7 __worldHeight

`Actor.Actor.__worldHeight [private]`

World height.

5.1.4.8 __worldWidth

`Actor.Actor.__worldWidth [private]`

World width.

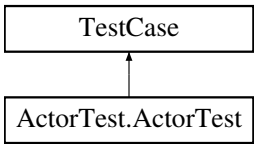
The documentation for this class was generated from the following file:

- C:/Users/teejp/Documents/Python/AD1-PIG/Actor.py

5.2 ActorTest.ActorTest Class Reference

Class for testing Actor.Actor class.

Inheritance diagram for ActorTest.ActorTest:



Public Member Functions

- def setUpClass (cls)

  Generate Worlds for testing purposes This method is used before execution of any other method We are using setUpClass(cls) instead of setUp(self) because in this case, because of Actor ID, we want to setUp only once not everytime a method is called by unittest.main()
- def test_constructor (self)
- def test_setLocation (self)
- def test_getWorld (self)
- def test_addedtoWorld (self)

Public Attributes

- world_one
- world_two
- actor_one
- actor_two
- actor_three

5.2.1 Detailed Description

Class for testing Actor.Actor class.

Author

Jefferson Peralva Machiqueira

5.2.2 Member Function Documentation

5.2.2.1 setUpClass()

```
def ActorTest.ActorTest.setUpClass (
              cls )
```

Generate Worlds for testing purposes This method is used before execution of any other method We are using setUpClass(cls) instead of setUp(self) because in this case, because of Actor ID, we want to setUp only once not everytime a method is called by unittest.main()

5.2.2.2 test_addedtoWorld()

```
def ActorTest.ActorTest.test_addedtoWorld (
              self )
```

5.2.2.3 test_constructor()

```
def ActorTest.ActorTest.test_constructor (
              self )
```

5.2.2.4 test_getWorld()

```
def ActorTest.ActorTest.test_getWorld (
              self )
```

5.2.2.5 test_setLocation()

```
def ActorTest.ActorTest.test_setLocation (
              self )
```

5.2.3 Member Data Documentation

5.2.3.1 actor_one

```
ActorTest.ActorTest.actor_one
```

5.2.3.2 actor_three

```
ActorTest.ActorTest.actor_three
```

5.2.3.3 actor_two

```
ActorTest.ActorTest.actor_two
```

5.2.3.4 world_one

```
ActorTest.ActorTest.world_one
```

5.2.3.5 world_two

```
ActorTest.ActorTest.world_two
```

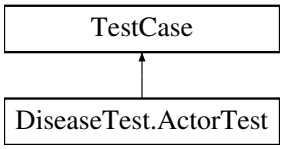
The documentation for this class was generated from the following file:

- C:/Users/teejp/Documents/Python/AD1-PIG/ActorTest.py

5.3 DiseaseTest.ActorTest Class Reference

Class for testing Disease.Disease class.

Inheritance diagram for DiseaseTest.ActorTest:



Public Member Functions

- def setUp (self)

  Generate World and disease for testing purposes.
- def test_constructor (self)
- def test_getStrenght (self)
- def test_getQuadrant (self)
- def test_setStrength (self)

Public Attributes

- world_one
- disease_one

5.3.1 Detailed Description

Class for testing Disease.Disease class.

Author

Jefferson Peralva Machiqueira

5.3.2 Member Function Documentation

5.3.2.1 setUp()

```
def DiseaseTest.ActorTest.setUp (
            self )
```

Generate World and disease for testing purposes.

5.3.2.2 test_constructor()

```
def DiseaseTest.ActorTest.test_constructor (
              self )
```

5.3.2.3 test_getQuadrant()

```
def DiseaseTest.ActorTest.test_getQuadrant (
              self )
```

5.3.2.4 test_getStrenght()

```
def DiseaseTest.ActorTest.test_getStrenght (
              self )
```

5.3.2.5 test_setStrength()

```
def DiseaseTest.ActorTest.test_setStrength (
              self )
```

5.3.3 Member Data Documentation

5.3.3.1 disease_one

```
DiseaseTest.ActorTest.disease_one
```

5.3.3.2 world_one

```
DiseaseTest.ActorTest.world_one
```

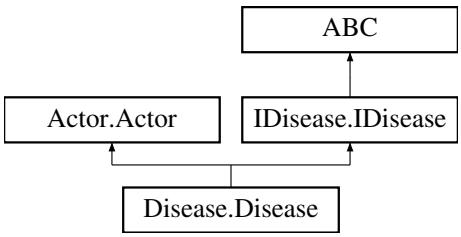
The documentation for this class was generated from the following file:

- C:/Users/teejp/Documents/Python/AD1-PIG/DiseaseTest.py

5.4 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, float lTemp, float hTemp, float gRate)

  Sets the disease growth rate, lower temperature and higher temperature.
- tuple getGrowthCondition (self)

  Returns the disease growth rate, lower temperature and higher temperature.
- int getQuadrant (self)

  Returns the quadrant of this disease.
- def act (self)

  Print on screen in the format "Iteration <ID>: Actor <Actor ID>." The $< ID >$ is replaced by the current iteration number.
- def getStrength (self)

  Return the disease strength of this object.
- def setStrength (self, strength)

  Set Strength.
- def __str__ (self)

  Return a string with the strength, growth and quadrant of this disease.

Private Attributes

- __growthRate

  Rate at which the disease grows when subjected to the appropriate temperature range.
- __lowerTemp

  Minimum temperature for the disease development.
- __higherTemp

  Maximum temperature for the disease development.
- __dStrength

  Disease strength.

5.4.1 Detailed Description

This Disease class is a sub-class of the Actor class.

Author

Jefferson Peralva Machiqueira

Date

31/08/2020

5.4.2 Constructor & Destructor Documentation

5.4.2.1 __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.

5.4.3 Member Function Documentation

5.4.3.1 __str__()

```
def Disease.Disease.__str__ (
              self )
```

Return a string with the strength, growth and quadrant of this disease.

Reimplemented from Actor.Actor.

5.4.3.2 act()

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

Print 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() method.

Reimplemented from Actor.Actor.

5.4.3.3 getGrowthCondition()

```
tuple Disease.Disease.getGrowthCondition (
              self )
```

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

Returns

growth rate, lower temp and higher temp

5.4.3.4 getQuadrant()

```
int Disease.Disease.getQuadrant (
              self )
```

Returns the quadrant of this disease.

Returns

0, 1, 2 or 3.

5.4.3.5 getStrength()

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

Return the disease strength of this object.

Returns

disease strength of the object.

Reimplemented from IDisease.IDisease.

5.4.3.6 setGrowthCondition()

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

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

Parameters

float	lTemp Lower bound temperature for the disease to grow at this gRate.
float	hTemp Upper bound temperature for the disease to grow at this gRate.
float	gRate The growth rate.

5.4.3.7 setStrength()

```
def Disease.Disease.setStrength (
              self,
              strength )
```

Set Strength.

Parameters

float	strength

5.4.4 Member Data Documentation

5.4.4.1 __dStrength

```
Disease.Disease.__dStrength  [private]
```

Disease strength.

5.4.4.2 __growthRate

```
Disease.Disease.__growthRate  [private]
```

Rate at which the disease grows when subjected to the appropriate temperature range.

5.4.4.3 __higherTemp

```
Disease.Disease.__higherTemp  [private]
```

Maximum temperature for the disease development.

5.4.4.4 __lowerTemp

```
Disease.Disease.__lowerTemp  [private]
```

Minimum temperature for the disease development.

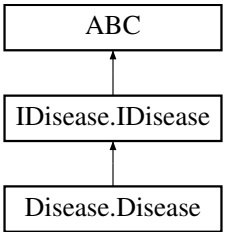
The documentation for this class was generated from the following file:

- C:/Users/teejp/Documents/Python/AD1-PIG/Disease.py

5.5 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.

Static Private Attributes

- __metaclass__ = ABCMeta

5.5.1 Detailed Description

Interface IDisease allows setting the strength and growth condition of a disease.

Author

Jefferson Peralva Machiqueira

5.5.2 Member Function Documentation

5.5.2.1 getStrength()

```
def IDisease.IDisease.getStrength (
              self )
```

Return the disease strength of the object implements this interface.

Reimplemented in Disease.Disease.

5.5.2.2 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.

5.5.3 Member Data Documentation

5.5.3.1 __metaclass__

```
IDisease.IDisease.__metaclass__ = ABCMeta  [static], [private]
```

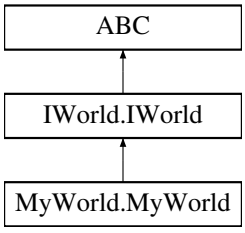
The documentation for this class was generated from the following file:

- C:/Users/teejp/Documents/Python/AD1-PIG/IDisease.py

5.6 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)
- def setTemp (self, quad, temp)
- def getTemp (self, quad)
- def getObjects (self)
- def getSumStrength (self)
- def initDiseases (self, numDisStr)
- def initLocations (self, locationsStr, diseaseArr)
- def initGrowthConditions (self, growthStr, diseaseArr)
- def initTemps (self, tempStr)

Static Private Attributes

- __metaclass__ = ABCMeta

5.6.1 Detailed Description

Interface IWorld allows initializing and setting diseases for a world.

Author

Jefferson Peralva Machiqueira

Date

31/08/2020

5.6.2 Member Function Documentation

5.6.2.1 getObjects()

```
def IWorld.IWorld.getObjects (
              self )
```

5.6.2.2 getSumStrength()

```
def IWorld.IWorld.getSumStrength (
              self )
```

Reimplemented in MyWorld.MyWorld.

5.6.2.3 getTemp()

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

5.6.2.4 initDiseases()

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

5.6.2.5 initGrowthConditions()

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

5.6.2.6 initLocations()

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

5.6.2.7 initTemps()

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

5.6.2.8 prepare()

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

Reimplemented in MyWorld.MyWorld.

5.6.2.9 setTemp()

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

5.6.3 Member Data Documentation

5.6.3.1 __metaclass__

```
IWorld.IWorld.__metaclass__ = ABCMeta  [static], [private]
```

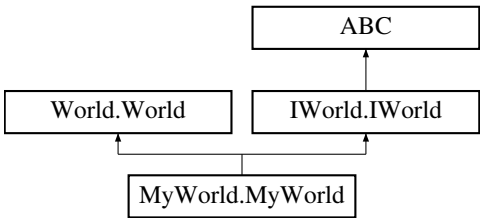
The documentation for this class was generated from the following file:

- C:/Users/teejp/Documents/Python/AD1-PIG/IWorld.py

5.7 MyWorld.MyWorld Class Reference

SubClass of World and IWorld classes.

Inheritance diagram for MyWorld.MyWorld:



Public Member Functions

- def __init__ (self, int w=720, int h=640)

  Call the constructor of the World with the width and height.
- def prepare (self)

  Prepare the world.
- ArrayDisease initDiseases (self, int numDisStr)

  Create Disease objects.
- int initGrowthConditions (self, str growthStr, ArrayDisease diseaseArr)

  Create gowth conditions.
- int initTemps (self, str tempStr)

  Create Temperatures.
- int initLocations (self, str locationsStr, ArrayDisease diseaseArr)

  setup all diseases in MyWorld

- float getSumStrength (self)

  Return the total disease strenght of all diseases.
- float getTemp (self, int quadID)

  Return the temperature of the region with the ID of quadID.
- def setTemp (self, tuple quadID, float temp)

  Set temperature of quadID quadrant.
- def act (self)

  Overrides the method act in the world class.

Private Attributes

- __temperature
- __itCounter

Static Private Attributes

- tuple __quadID = (0, 1, 2, 3)

5.7.1 Detailed Description

SubClass of World and IWorld classes.

Author

Jefferson Peralva Machiqueira

5.7.2 Constructor & Destructor Documentation

5.7.2.1 __init__()

```
def MyWorld.MyWorld.__init__ (
              self,
             int  w = 720,
             int  h = 640 )
```

Call the constructor of the World with the width and height.

Reimplemented from World.World.

5.7.3 Member Function Documentation

5.7.3.1 act()

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

Overrides the method act in the world class.

Reimplemented from World.World.

5.7.3.2 getSumStrength()

```
float MyWorld.MyWorld.getSumStrength (
              self )
```

Return the total disease strenght of all diseases.

Returns

float

Reimplemented from IWorld.IWorld.

5.7.3.3 getTemp()

```
float MyWorld.MyWorld.getTemp (
              self,
             int quadID )
```

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

Parameters

int	quadID

Returns

float with the temperature of ID

5.7.3.4 initDiseases()

```
ArrayDisease MyWorld.MyWorld.initDiseases (
              self,
             int numDisStr )
```

Create Disease objects.

Parameters

int	numDisStr

Returns

array of diseases

5.7.3.5 initGrowthConditions()

```
int MyWorld.MyWorld.initGrowthConditions (
              self,
             str growthStr,
             ArrayDisease diseaseArr )
```

Create gowth conditions.

Parameters

str	growthStr
ArrayDiseases	diseaseArr

Returns

int

5.7.3.6 initLocations()

```
int MyWorld.MyWorld.initLocations (
              self,
             str locationsStr,
             ArrayDisease diseaseArr )
```

setup all diseases in MyWorld

Parameters

str	locationsStr
diseaseArr	

5.7.3.7 initTemps()

```
int MyWorld.MyWorld.initTemps (
              self,
             str tempStr )
```

Create Temperatures.

Parameters

str	tempStr

Returns

int

5.7.3.8 prepare()

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

Prepare the world.

Open a text file named "simulation.config" in the current path Parse the configuration for the number of disease objects, the cell locations of these objects, the gowth rates, and the temperature ranges associates with individual gowth rates

Reimplemented from IWorld.IWorld.

5.7.3.9 setTemp()

```
def MyWorld.MyWorld.setTemp (
              self,
             tuple quadID,
             float temp )
```

Set temperature of quadID quadrant.

Parameters

int	quadID
dict	temp

5.7.4 Member Data Documentation

5.7.4.1 __itCounter

```
MyWorld.MyWorld.__itCounter  [private]
```

5.7.4.2 __quadID

```
tuple MyWorld.MyWorld.__quadID = (0, 1, 2, 3)  [static], [private]
```

5.7.4.3 __temperature

```
MyWorld.MyWorld.__temperature  [private]
```

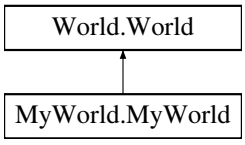
The documentation for this class was generated from the following file:

- C:/Users/teejp/Documents/Python/AD1-PIG/MyWorld.py

5.8 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, int worldWidth, int worldHeight)

  Constructor.
- Grid createGrid (self, int h, int w, int d)

  Initializes each object of the array as None.
- str __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.
- int addObject (self, Actor obj, int x, int y)

  Adds a new actor to this world at a given position.
- int getHeight (self)

  Height getter.
- int getWidth (self)

  Width getter.
- int getDepth (self)

  Depth getter.
- Grid getGrid (self)

  Grid getter.
- int numberOfObjects (self)

  Returns the total number of objects in this world.
- ArrayActor getObjects (self)

  Returns an array with all Actor objects in this world.
- def setGrid (self, aGrid, numObjs)

  todo implement set_grid()

Private Attributes

- __width

  Private instance attribute width.
- __height

  Private instance attribute height.
- __depth

  Private instance attribute depth.
- __grid

  Private instance attribute grid.
- __objCounter

  Private instance attribute objCounter.

5.8.1 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

Jefferson Peralva Machiqueira

5.8.2 Constructor & Destructor Documentation

5.8.2.1 __init__()

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

Constructor.

Creates a world with the given width and height.

Parameters

int	world_width Width in number of cells
int	world_height Height in number of cells

Reimplemented in MyWorld.MyWorld.

5.8.3 Member Function Documentation

5.8.3.1 __repr__()

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

Return a string representation of the grid.

List by depth. Each slice is height x width.

Returns

str string with the grid.

5.8.3.2 __str__()

```
str World.World.__str__ (
              self )
```

Return a string representation of the grid.

List by width. Each slice is height x depth.

Returns

str string with the grid.

5.8.3.3 act()

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

Blank method body.

Overriden in subclasses as appropriate

Reimplemented in MyWorld.MyWorld.

5.8.3.4 addObject()

```
int World.World.addObject (
              self,
             Actor obj,
             int x,
             int y )
```

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

Parameters

Actor	obj
int	x width
int	y height

Returns

int Number of objects in that cell

Exceptions

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

5.8.3.5 createGrid()

```
Grid World.World.createGrid (
              self,
             int h,
             int w,
             int d )
```

Initializes each object of the array as None.

Parameters

h	grid height.
w	grid width.
d	grid depth.

Returns

grid. PS.: This method could be static, but I was oriented to keep the professor proposed skeleton

5.8.3.6 getDepth()

```
int World.World.getDepth (
              self )
```

Depth getter.

Returns

int Returns the world depth.

5.8.3.7 getGrid()

```
Grid World.World.getGrid (
              self )
```

Grid getter.

Returns

Grid Returns the grid.

5.8.3.8 getHeight()

```
int World.World.getHeight (
              self )
```

Height getter.

Returns

int Returns the world height.

5.8.3.9 getObjects()

```
ArrayActor World.World.getObjects (
              self )
```

Returns an array with all Actor objects in this world.

Returns

ArrayActor List[Actor]

5.8.3.10 getWidth()

```
int World.World.getWidth (
              self )
```

Width getter.

Returns

int Returns the world width.

5.8.3.11 numberOfObjects()

```
int World.World.numberOfObjects (
              self )
```

Returns the total number of objects in this world.

Returns

int Total number of objects in this world.

5.8.3.12 setGrid()

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

todo implement set_grid()

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

aGrid	reference to a 3D array of Actor objects.
numObjs	the number of Actor objects in aGrid.

Exceptions

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

5.8.4 Member Data Documentation

5.8.4.1 __depth

`World.World.__depth [private]`

Private instance attribute depth.

5.8.4.2 __grid

`World.World.__grid [private]`

Private instance attribute grid.

5.8.4.3 __height

`World.World.__height [private]`

Private instance attribute height.

5.8.4.4 __objCounter

`World.World.__objCounter [private]`

Private instance attribute objCounter.

5.8.4.5 __width

```
World.World.__width  [private]
```

Private instance attribute width.

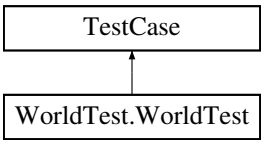
The documentation for this class was generated from the following file:

- C:/Users/teejp/Documents/Python/AD1-PIG/World.py

5.9 WorldTest.WorldTest Class Reference

Class for testing World.World class.

Inheritance diagram for WorldTest.WorldTest:



Public Member Functions

- def setUp (self)

  Generate Worlds for testing purposes This method is used before execution of any other method.
- def test_getWidthandHeight (self)

  Test initial height and width.
- def test_addObj (self)

  Test to see if added object to correct cell.
- def test_nullBeginning (self)

  Tests to see if the grid is completely initialized as null.
- def test_exceptions (self)

  Tests the thrown exceptions of addObject()
- def test_setGrid (self)

  todo implement
- def test_largeWorld (self)

  Sets the world to an illegal size.

Public Attributes

- world_one
- world_two

5.9.1 Detailed Description

Class for testing World.World class.

Author

Jefferson Peralva Machiqueira

5.9.2 Member Function Documentation

5.9.2.1 setUp()

```
def WorldTest.WorldTest.setUp (
              self )
```

Generate Worlds for testing purposes This method is used before execution of any other method.

5.9.2.2 test_addObj()

```
def WorldTest.WorldTest.test_addObj (
              self )
```

Test to see if added object to correct cell.

5.9.2.3 test_exceptions()

```
def WorldTest.WorldTest.test_exceptions (
              self )
```

Tests the thrown exceptions of addObject()

5.9.2.4 test_getWidthandHeight()

```
def WorldTest.WorldTest.test_getWidthandHeight (
              self )
```

Test initial height and width.

5.9.2.5 test_largeWorld()

```
def WorldTest.WorldTest.test_largeWorld (
              self )
```

Sets the world to an illegal size.

5.9.2.6 test_nullBeginning()

```
def WorldTest.WorldTest.test_nullBeginning (
              self )
```

Tests to see if the grid is completely initialized as null.

5.9.2.7 test_setGrid()

```
def WorldTest.WorldTest.test_setGrid (
              self )
```

todo implement

5.9.3 Member Data Documentation

5.9.3.1 world_one

```
WorldTest.WorldTest.world_one
```

5.9.3.2 world_two

```
WorldTest.WorldTest.world_two
```

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

- C:/Users/teejp/Documents/Python/AD1-PIG/WorldTest.py

Chapter 6

File Documentation

6.1 C:/Users/teejp/Documents/Python/AD1-PIG/Actor.py File Reference

Classes

- class Actor.Actor

  Actor class, which is the base class for Disease objects.

Namespaces

- Actor

6.2 C:/Users/teejp/Documents/Python/AD1-PIG/ActorTest.py File Reference

Classes

- class ActorTest.ActorTest

  Class for testing Actor.Actor class.

Namespaces

- ActorTest
- WorldTest

Variables

- ActorTest.verbosity

6.3 C:/Users/teejp/Documents/Python/AD1-PIG/Constants.py File Reference

Namespaces

- Constants
- World

Variables

- int Constants.MAXIMUM_WIDTH = 1000

  Constant with the maximum grid width.
- int Constants.MAXIMUM_HEIGHT = 1000

  Constant with the maximum grid height.

6.4 C:/Users/teejp/Documents/Python/AD1-PIG/Disease.py File Reference

Classes

- class Disease.Disease

  This Disease class is a sub-class of the Actor class.

Namespaces

- Disease

6.5 C:/Users/teejp/Documents/Python/AD1-PIG/DiseaseTest.py File Reference

Classes

- class DiseaseTest.ActorTest

  Class for testing Disease.Disease class.

Namespaces

- DiseaseTest
- WorldTest

Variables

- DiseaseTest.verbosity

6.6 C:/Users/teejp/Documents/Python/AD1-PIG/IDisease.py File Reference

Classes

- class IDisease.IDisease

  Interface IDisease allows setting the strength and growth condition of a disease.

Namespaces

- IDisease

Variables

- IDisease.ABC = object

6.7 C:/Users/teejp/Documents/Python/AD1-PIG/IWorld.py File Reference

Classes

- class IWorld.IWorld

  Interface IWorld allows initializing and setting diseases for a world.

Namespaces

- IWorld

Variables

- IWorld.ABC = object

6.8 C:/Users/teejp/Documents/Python/AD1-PIG/MyWorld.py File Reference

Classes

- class MyWorld.MyWorld

  SubClass of World and IWorld classes.

Namespaces

- MyWorld

Variables

- MyWorld.ArrayDisease = List[Disease]

  Type definition for use in Python Type Hinting for ArrayDisease/list-of-disease-instances.
- MyWorld.valor = MyWorld(720, 640)
- MyWorld.objetos = valor.getObjects()

6.9 C:/Users/teejp/Documents/Python/AD1-PIG/simulator.py File Reference

Namespaces

- simulator

Functions

- def simulator.main (args=None)

  This is the main method that sets up a virtual world and simulates the gowth of the diseases in the world if the number of iterations is given in the comand line argument, run the simulation for that number of iterations Otherwise, use the deafault number of iterations: 5.

6.10 C:/Users/teejp/Documents/Python/AD1-PIG/World.py File Reference

Classes

- class World.World

  Class for holding Actor objects in cells of a grid in the world.

Namespaces

- World

Functions

- def World.main ()

Variables

- World.Grid = List[List[List[int or None or Actor]]]

  Type definition for use in Python Type Hinting for Grid/3D-list.
- World.ArrayActor = List[Actor]

  Type definition for use in Python Type Hinting for ArrayActor/list-of-actor-instances.

6.11 C:/Users/teejp/Documents/Python/AD1-PIG/WorldTest.py File Reference

Classes

- class WorldTest.WorldTest

  Class for testing World.World class.

Namespaces

- WorldTest

Variables

- WorldTest.verbosity

Index