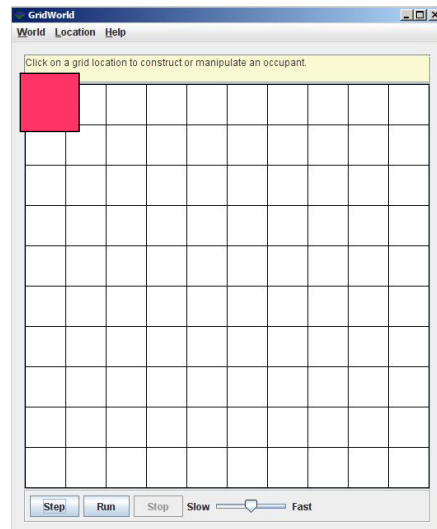




What is GridWorld?



Row = 0

Column = 0

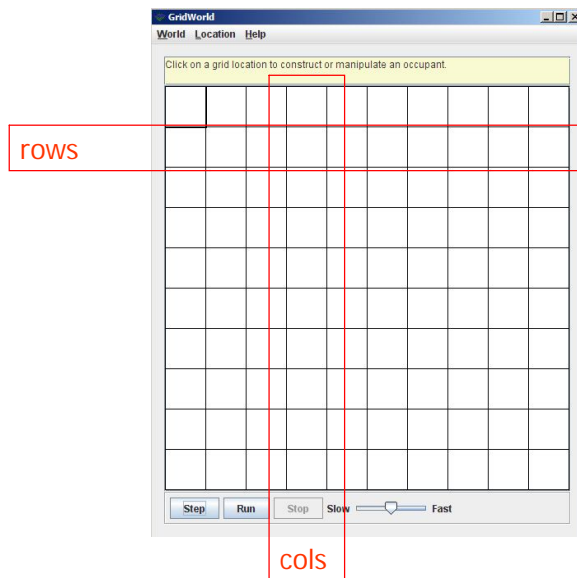
© A+ Computer Science - www.apluscompsci.com

A grid is a structure that has rows and columns.

A spreadsheet is a grid.

A checker board is a grid.

What is GridWorld?



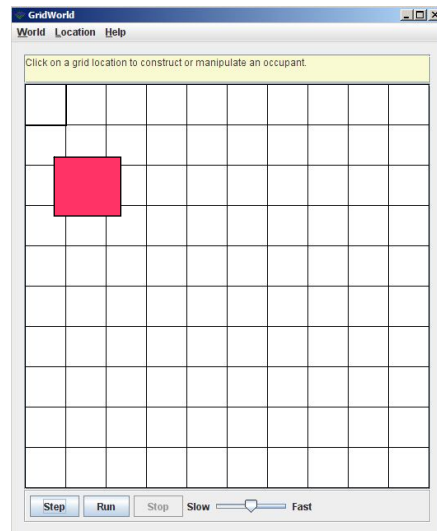
A grid is a structure that has rows and columns.

A grid is a structure that has rows and columns.

A spreadsheet is a grid.

A checker board is a grid.

What is GridWorld?



Row = 2

Column = 1

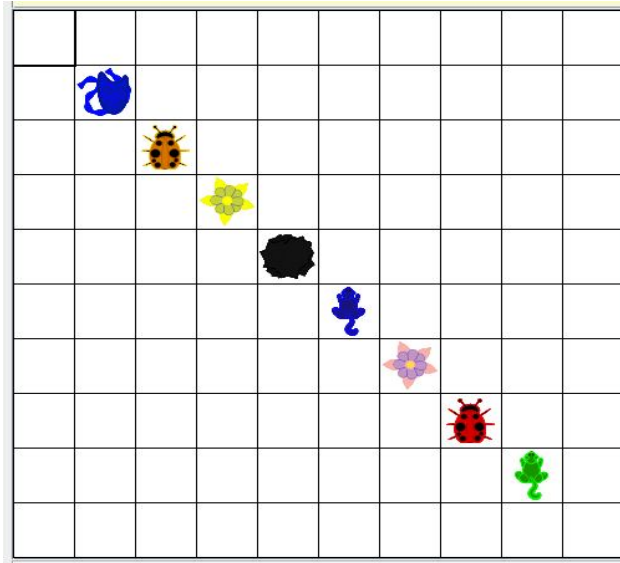
© A+ Computer Science - www.apluscompsci.com

A grid is a structure that has rows and columns.

A spreadsheet is a grid.

A checker board is a grid.

What is GridWorld?



© A+ Computer Science - www.apluscompsci.com

GridWorld

Classes

© A+ Computer Science - www.apluscompsci.com

ActorWorld

© A+ Computer Science - www.apluscompsci.com

ActorWorld

frequently used methods

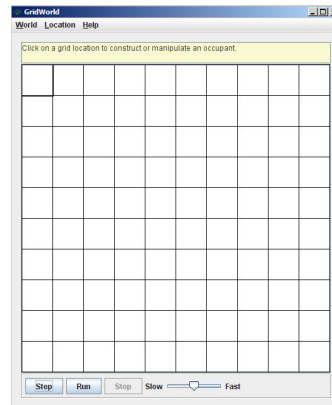
Name	Use
ActorWorld()	creates a new world using 10X10 grid
ActorWorld(grid)	creates a new world using grid
add(loc, thang)	add thang at spot loc
add(thang)	add thang at a random empty loc
show()	makes the world visible

```
import info.gridworld.actor.World;
```

© A+ Computer Science - www.apluscompsci.com

ActorWorld

```
ActorWorld earth = new ActorWorld();  
earth.show();
```



© A+ Computer Science - www.apluscompsci.com

ActorWorld is used to graphically display a grid of Actors.

ActorWorld

A reference variable stores the memory address of an object.

```
ActorWorld earth = new ActorWorld();
```



© A+ Computer Science - www.apluscompsci.com

A reference variable is used to store the location of an Object. In most situations, a reference stores the actual memory address of an Object.

`earth` stores the location / memory address of a new `ActorWorld` object.

ActorWorld

```
ActorWorld earth = new ActorWorld();
```

earth
0xF5



0xF5

ActorWorld Object

earth stores the address of an ActorWorld

© A+ Computer Science - www.apluscompsci.com

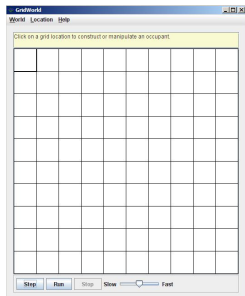

A reference variable is used to store the location of an Object. In most situations, a reference stores the actual memory address of an Object.

earth stores the location / memory address of a new ActorWorld.

ActorWorld

reference command / method

earth.show();



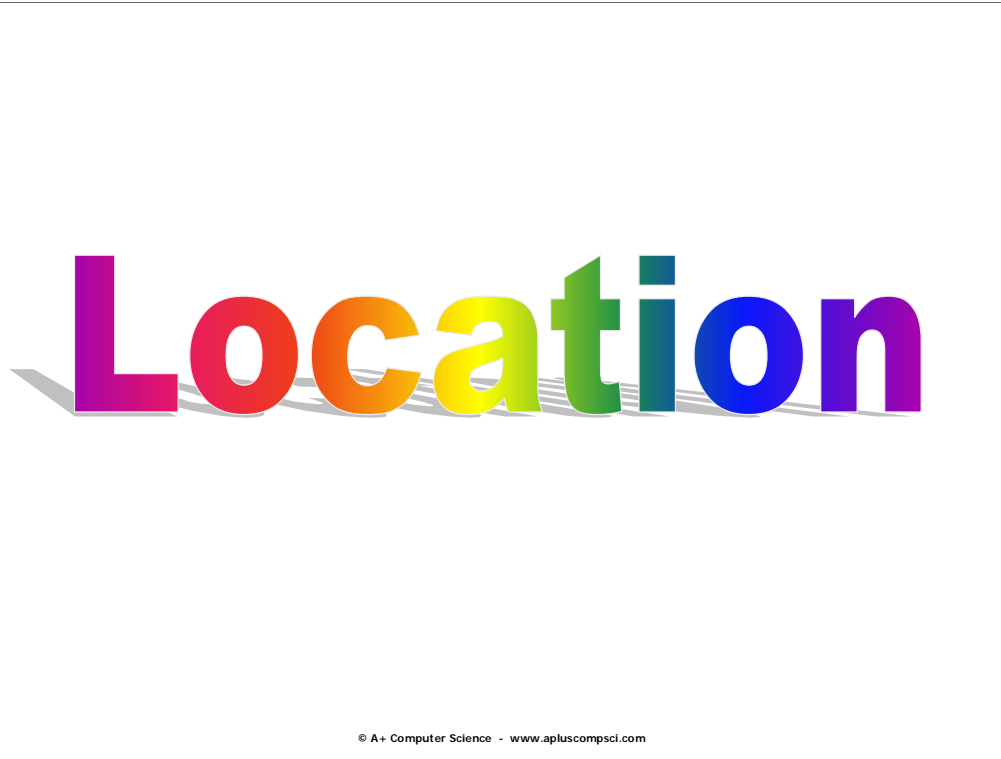
© A+ Computer Science - www.apluscompsci.com

earth is a reference to an ActorWorld.

earth.show() calls the ActorWorld show method to show the grid of actors graphically.

**open
gridworldone.java**

© A+ Computer Science - www.apluscompsci.com



Location

frequently used methods

Name	Use
<code>Location(row, col)</code>	creates a new row,col Location
<code>getCol()</code>	gets the column value for this location
<code>getRow()</code>	gets the row value for this location

```
import info.gridworld.grid.Location;
```

© A+ Computer Science - www.apluscompsci.com

Location is a class that stores row and column information.

```
Location spot = new Location(4,5);
```

spot has a row value of 4 and a column value of 5.

Location

```
Location locTwo = new Location(3,5);  
System.out.println(locTwo);
```

```
Location locThree = new Location(2,9);  
System.out.println(locThree);
```

OUTPUT

(3, 5)

(2, 9)

The Location class stores row and column information.

© A+ Computer Science - www.apluscompsci.com

Location is a class that stores row and column information.

```
Location spot = new Location(4,5);
```

spot has a row value of 4 and a column value of 5.

Location

reference variable

```
Location theLoc = new Location(3,4);
```

object instantiation

© A+ Computer Science - www.apluscompsci.com

`Location` is a class which must be instantiated before it can be used. In other words, you must make a new `Location` if you want to use a `Location`. A reference must be used to store the location in memory of the `Location` object created.

A row value and a column value must be passed as parameters to the `Location` constructor.

`theLoc` is a reference that will store the location/memory address of newly created `Location` object.

**open
locationnone.java**

© A+ Computer Science - www.apluscompsci.com

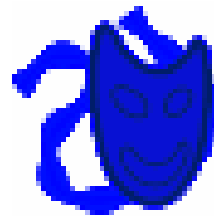
Actor

© A+ Computer Science - www.apluscompsci.com

Actor

Actor is the basic object from which all other GridWorld actors will be built.

Each of the new actors created will extend the original actor class.

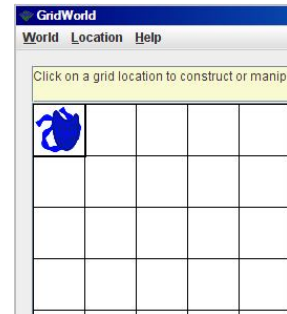


© A+ Computer Science - www.apluscompsci.com

Actor

```
ActorWorld world = new ActorWorld();  
Actor dude = new Actor();  
Location loc = new Location(0,0);  
world.add(loc, dude);  
world.show();
```

What happens if you click
on the actor?



© A+ Computer Science - www.apluscompsci.com

The add method of class World receives a Location parameter and an Actor reference.

The Actor reference is stored in the grid at the specified Location.

The default color of an Actor is BLUE.

```
ActorWorld world = new ActorWorld();  
Actor dude = new Actor();  
Location loc = new Location(0,0);  
world.add(loc, dude);  
world.show();
```

Methods

reference

command / method

world.add(loc, dude);



Methods are used to tell
an object what to do.

© A+ Computer Science - www.apluscompsci.com

World is the reference to an world.

Add is a method that adds a reference at a particular location.

The add method is called to tell world to add dude at location loc.

open actorone.java

© A+ Computer Science - www.apluscompsci.com





Bug differs from actor in that a bug actually moves from cell to cell.

A bug moves to the cell immediately in front if possible. If a move is not possible, the bug turns in 45 degree increments until it finds a spot to which it can move.

© A+ Computer Science - www.apluscompsci.com

Bug is a suped up Actor.

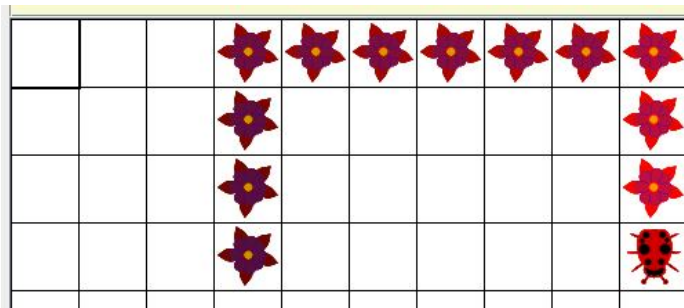
Bug extends Actor.

Bug has two constructors, one of which takes a Color parameter.

Bug will move when its act method is called.



```
ActorWorld world = new ActorWorld();  
Bug dude = new Bug();  
world.add(new Location(3,3), dude);  
world.show();
```



© A+ Computer Science - www.apluscompsci.com

In this example, a default Bug is created.

A default bug is red and facing NORTH.

The bug moves NORTH until it reaches the top of the grid.
The bug then turns in 45 degree increments until it finds an empty location to which to move.

**open
bugone.java**

© A+ Computer Science - www.apluscompsci.com

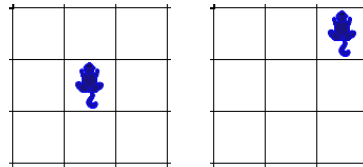
The word "critter" is rendered in a bold, 3D sans-serif font. Each letter is a different color from the rainbow spectrum: 'c' is pink, 'r' is orange, 'i' is yellow, 't' is green, 't' is blue, 'e' is purple, and 'r' is dark blue. The letters have a slight 3D effect with a soft shadow cast to the right and slightly forward.

© A+ Computer Science - www.apluscompsci.com

Critter

Critter differs from actor in that a critter moves around the grid and eats specific types of other actors.

Critter randomly picks one of its valid adjacent empty locations and moves to that location.



© A+ Computer Science - www.apluscompsci.com

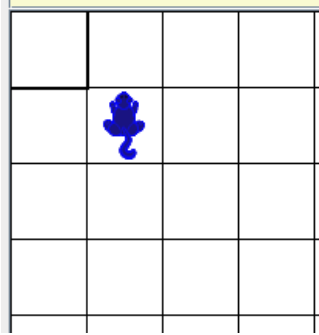
Critter is a suped up actor that moves around the grid and eats other actors.

Critter will eat all neighbors that are not critters and not rocks.

Critter randomly selects one of its valid empty adjacent locations. Critter moves to the new location.

Critter

```
ActorWorld world = new ActorWorld();  
Critter thang = new Critter();  
Location loc = new Location(1,1);  
world.add(loc, thang);  
world.show();
```



© A+ Computer Science - www.apluscompsci.com

//shortcut version of the code above

```
ActorWorld world = new ActorWorld();  
world.add(new Location(1,1), new Critter());  
world.show();
```

open
critterone.java

© A+ Computer Science - www.apluscompsci.com



imports

Imports are used to tell Java
where to find a class.

```
import info.gridworld.grid.Location;
```

```
import info.gridworld.actor.Rock;
```

```
import info.gridworld.actor.Flower;
```

```
import java.awt.Color;
```

© A+ Computer Science - www.apluscompsci.com

**open
gridworldtwo.java**

© A+ Computer Science - www.apluscompsci.com

Start work on ActorBox

© A+ Computer Science - www.apluscompsci.com