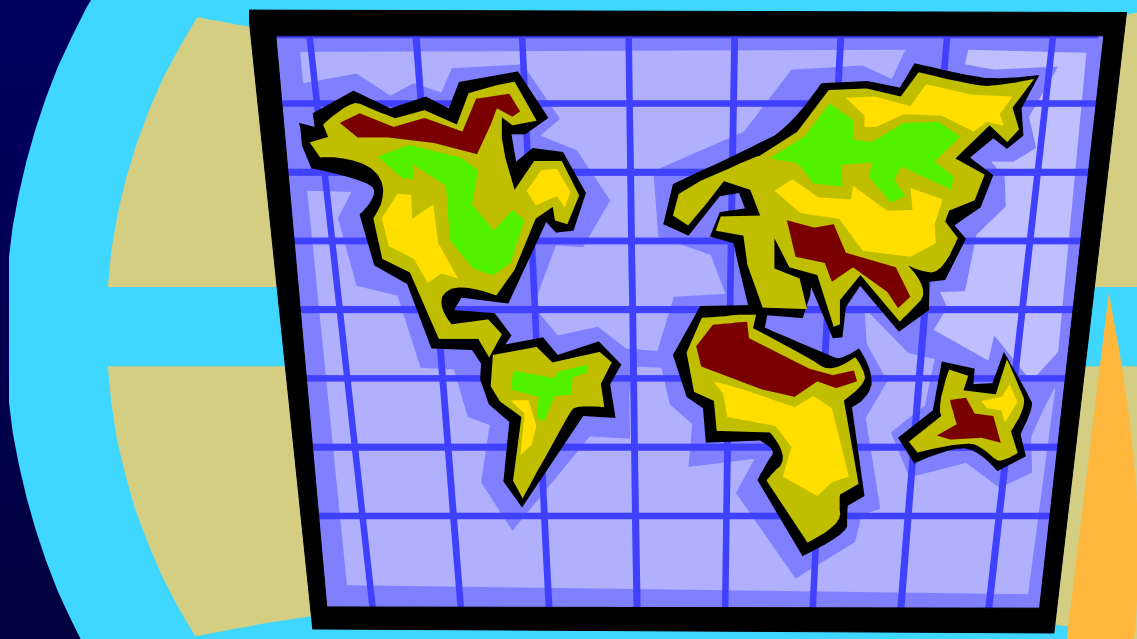
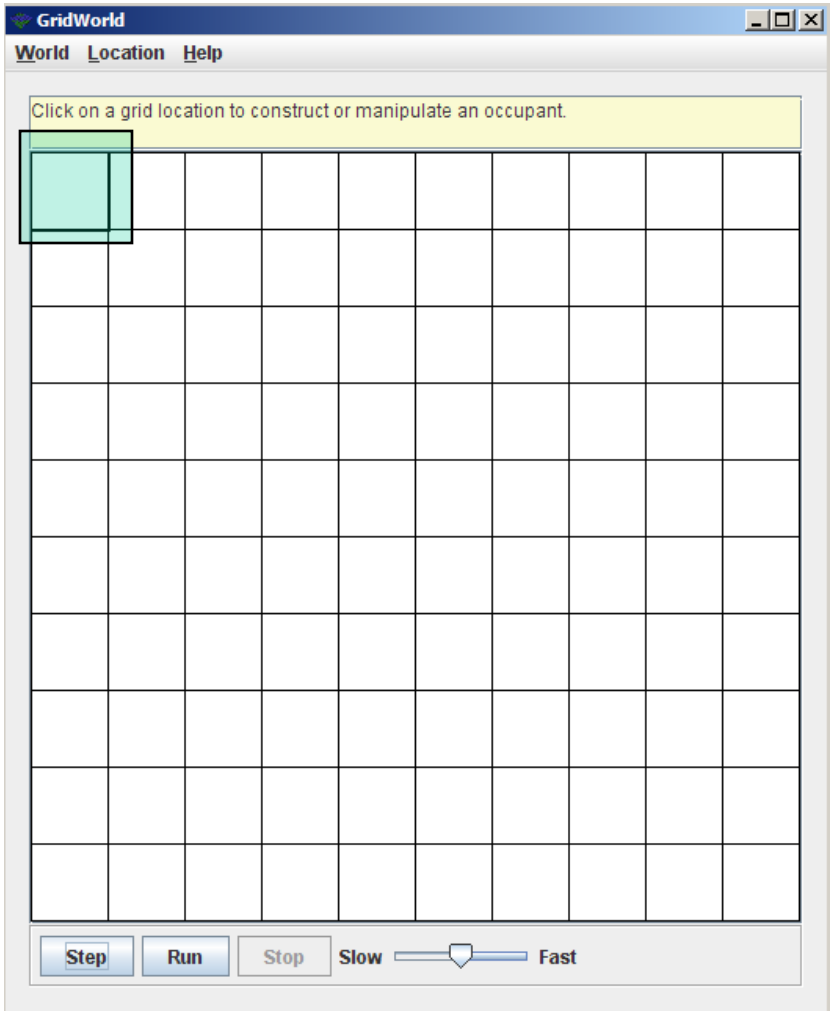
GridWorld



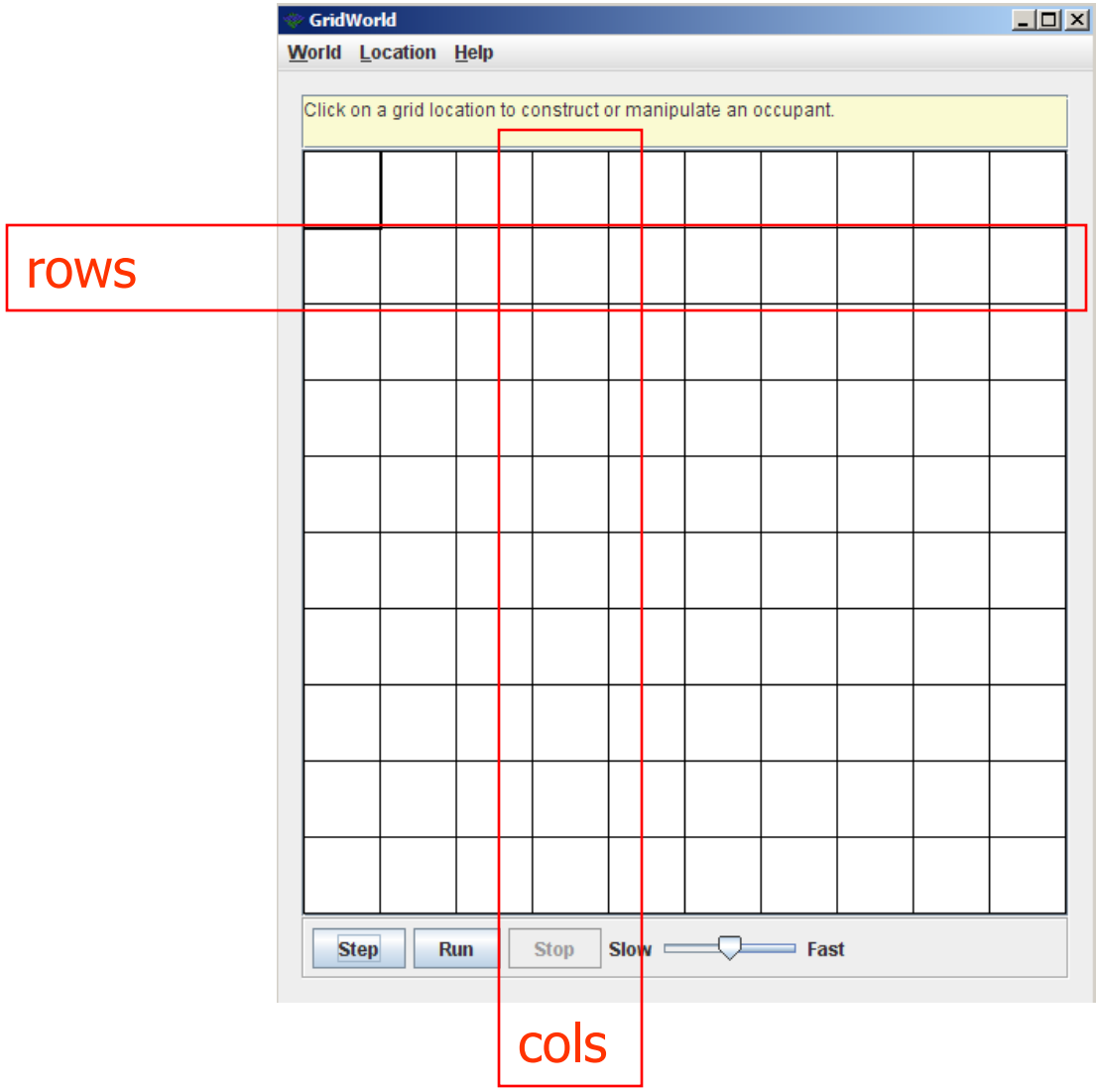
What is GridWorld?



Row = 0

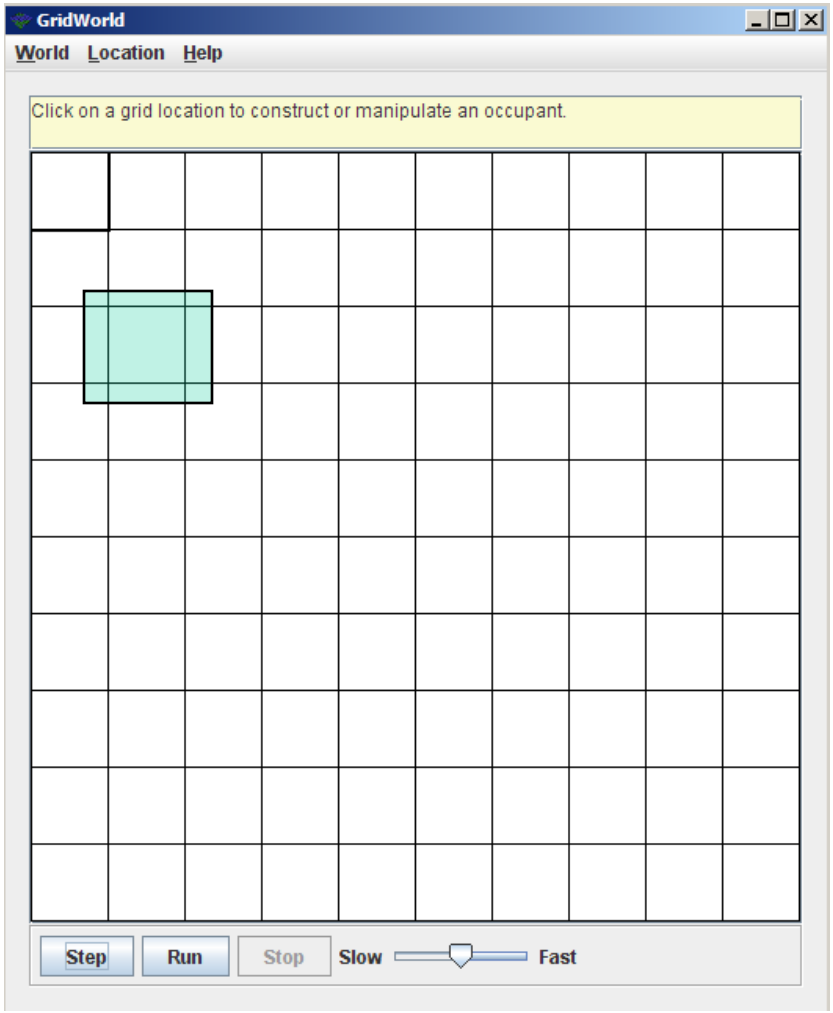
Column = 0

What is GridWorld?



A grid is a structure that has rows and columns.

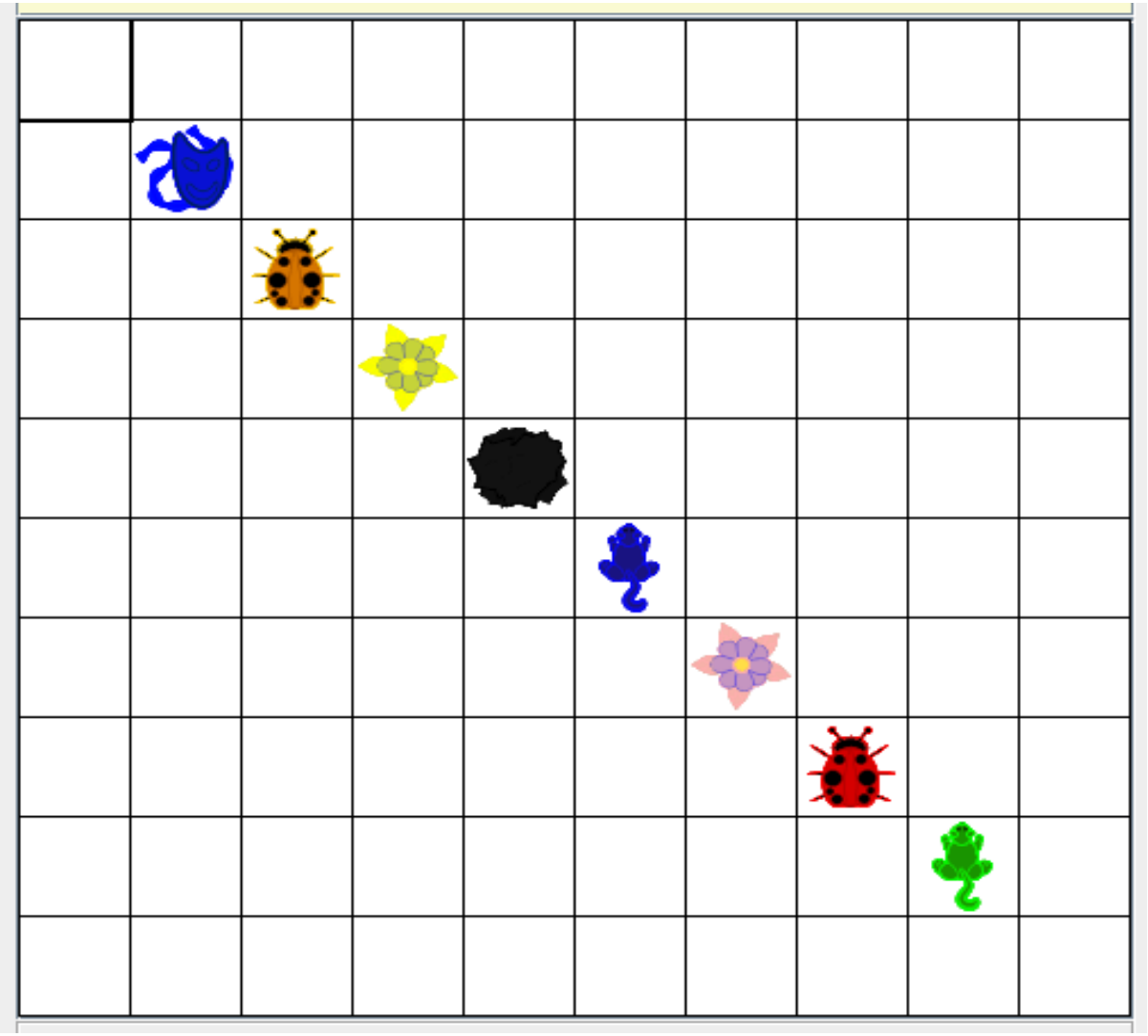
What is GridWorld?



Row = 2

Column = 1

What is GridWorld?



GridWorld Classes

ActorWorld

ActorWorld

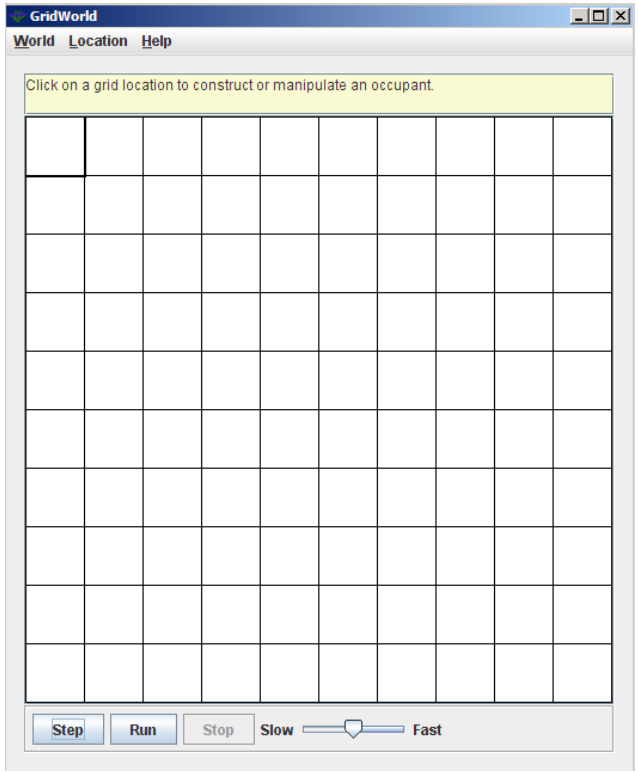
frequently used methods

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;
```

ActorWorld

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



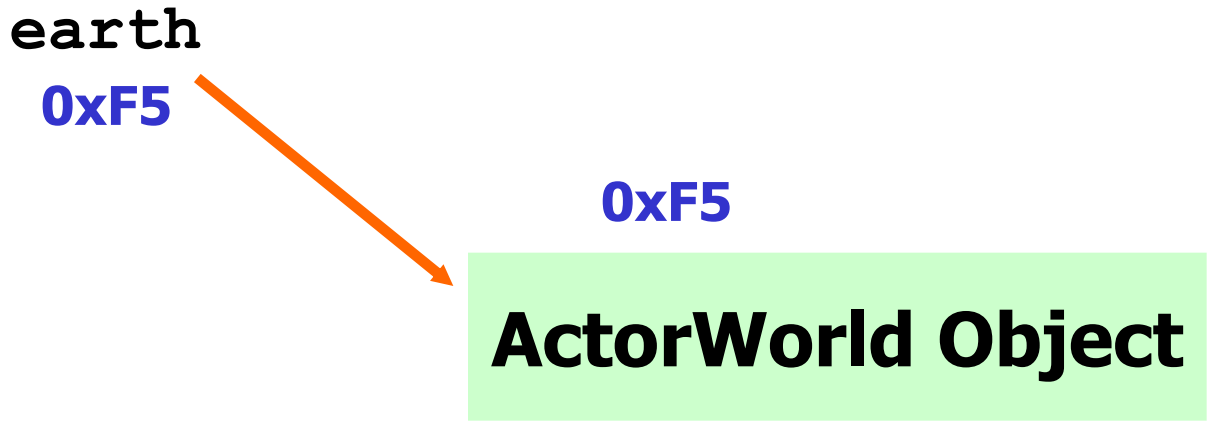
ActorWorld

A reference variable stores the memory address of an object.

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

ActorWorld

ActorWorld earth = new ActorWorld();



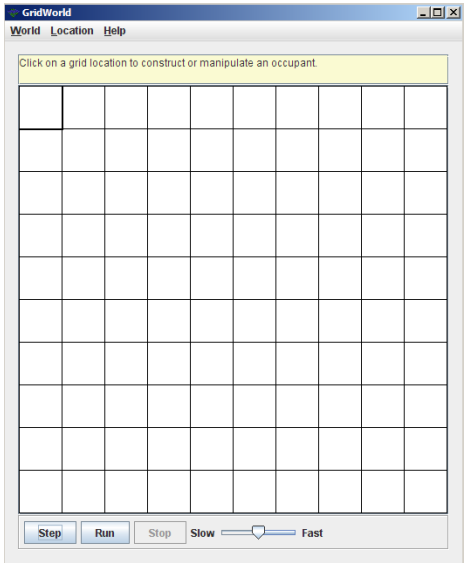
earth stores the address of an ActorWorld

ActorWorld

reference

command / method

```
earth.show();
```



open gridworldone.java

Location

Location

frequently used methods

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

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

Location

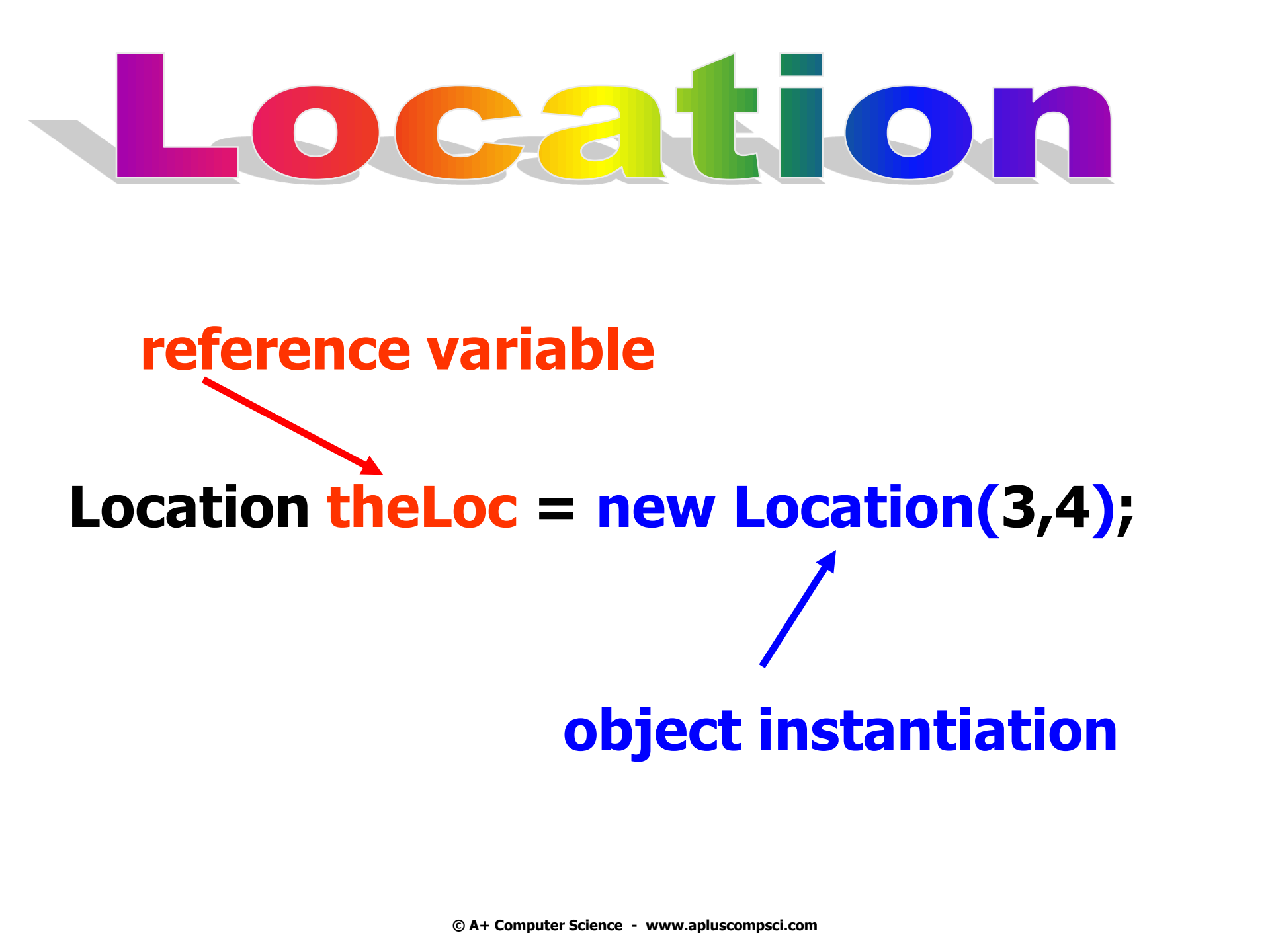
```
Location locTwo = new Location(3,5);
System.out.println(locTwo);

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

OUTPUT

```
(3, 5)
(2, 9)
```

The Location class stores row and column information.

Location

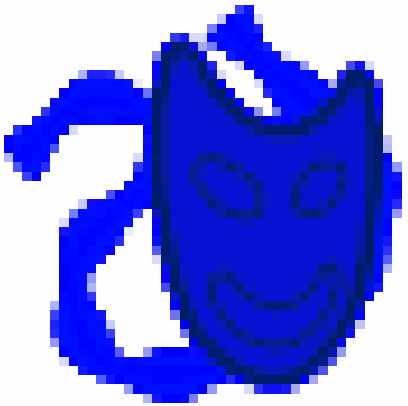


open locationone.java

Actor

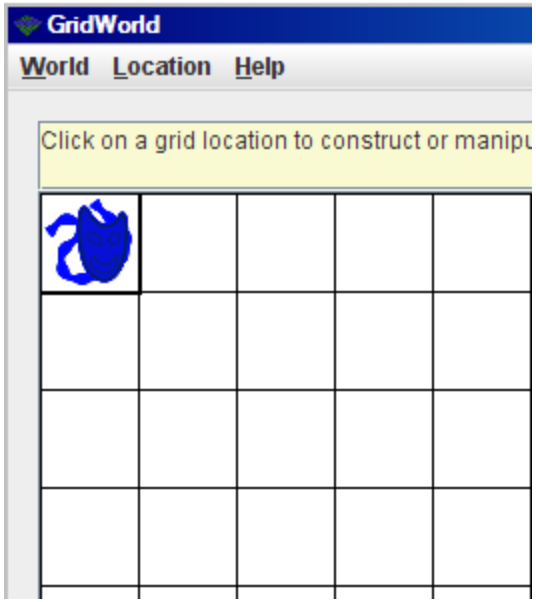
Actor is the basic object from which all other GridWorld actors will be built.

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



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?

Methods

reference

command / method

```
world.add(loc, dude);
```

Methods are used to tell an object what to do.

open actorone.java

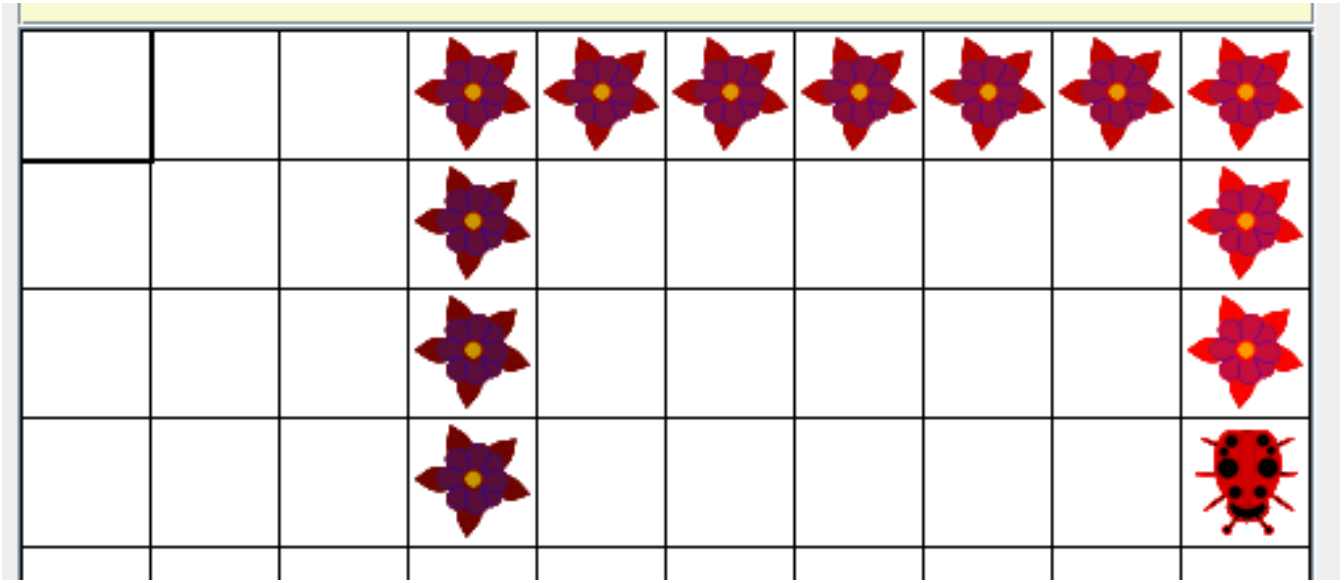
Bug

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.

Bug

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

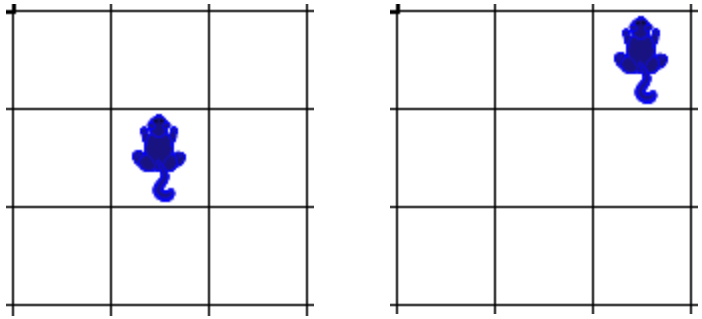


open bugone.java

Critter

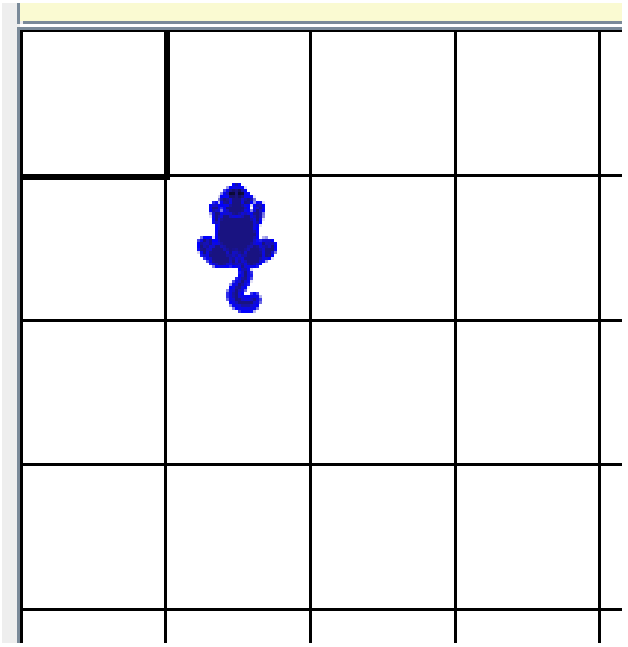
Critter differs from actor in that a critter moves around the grid and eats specific types of other actors.

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



Critter

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



open critterone.java

imports

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;
```

open gridworldtwo.java

Start work on ActorBox