| | [**Overview**](http://docs.google.com/overview-summary.html) | Package | Class | [**Tree**](http://docs.google.com/overview-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | **Index** | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | | |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PREV   NEXT | [**FRAMES**](http://docs.google.com/index.html?index-all.html)    [**NO FRAMES**](http://docs.google.com/index-all.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |

[A](#3znysh7) [B](#2et92p0) [C](#tyjcwt) [E](#3dy6vkm) [F](#1t3h5sf) [G](#4d34og8) [H](#2s8eyo1) [I](#17dp8vu) [K](#3rdcrjn) [L](#26in1rg) [M](#lnxbz9) [N](#35nkun2) [P](#1ksv4uv) [R](#44sinio) [S](#2jxsxqh) [T](#z337ya) [U](#3j2qqm3) [W](#1y810tw)

## **A**

[**AbstractGrid**](http://docs.google.com/info/gridworld/grid/AbstractGrid.html)<[E](http://docs.google.com/info/gridworld/grid/AbstractGrid.html)> - Class in [info.gridworld.grid](http://docs.google.com/info/gridworld/grid/package-summary.html)AbstractGrid contains the methods that are common to grid implementations.[**AbstractGrid()**](http://docs.google.com/info/gridworld/grid/AbstractGrid.html#AbstractGrid()) - Constructor for class info.gridworld.grid.[AbstractGrid](http://docs.google.com/info/gridworld/grid/AbstractGrid.html)   [**act()**](http://docs.google.com/info/gridworld/actor/Actor.html#act()) - Method in class info.gridworld.actor.[Actor](http://docs.google.com/info/gridworld/actor/Actor.html) Reverses the direction of this actor. [**act()**](http://docs.google.com/info/gridworld/actor/Bug.html#act()) - Method in class info.gridworld.actor.[Bug](http://docs.google.com/info/gridworld/actor/Bug.html) Moves if it can move, turns otherwise. [**act()**](http://docs.google.com/info/gridworld/actor/Critter.html#act()) - Method in class info.gridworld.actor.[Critter](http://docs.google.com/info/gridworld/actor/Critter.html) A critter acts by getting a list of its neighbors, processing them, getting locations to move to, selecting one of them, and moving to the selected location. [**act()**](http://docs.google.com/info/gridworld/actor/Flower.html#act()) - Method in class info.gridworld.actor.[Flower](http://docs.google.com/info/gridworld/actor/Flower.html) Causes the color of this flower to darken. [**act()**](http://docs.google.com/info/gridworld/actor/Rock.html#act()) - Method in class info.gridworld.actor.[Rock](http://docs.google.com/info/gridworld/actor/Rock.html) Overrides the act method in the Actor class to do nothing. [**Actor**](http://docs.google.com/info/gridworld/actor/Actor.html) - Class in [info.gridworld.actor](http://docs.google.com/info/gridworld/actor/package-summary.html)An Actor is an entity with a color and direction that can act.[**Actor()**](http://docs.google.com/info/gridworld/actor/Actor.html#Actor()) - Constructor for class info.gridworld.actor.[Actor](http://docs.google.com/info/gridworld/actor/Actor.html) Constructs a blue actor that is facing north. [**ActorWorld**](http://docs.google.com/info/gridworld/actor/ActorWorld.html) - Class in [info.gridworld.actor](http://docs.google.com/info/gridworld/actor/package-summary.html)An ActorWorld is occupied by actors.[**ActorWorld()**](http://docs.google.com/info/gridworld/actor/ActorWorld.html#ActorWorld()) - Constructor for class info.gridworld.actor.[ActorWorld](http://docs.google.com/info/gridworld/actor/ActorWorld.html) Constructs an actor world with a default grid. [**ActorWorld(Grid<Actor>)**](http://docs.google.com/info/gridworld/actor/ActorWorld.html#ActorWorld(info.gridworld.grid.Grid)) - Constructor for class info.gridworld.actor.[ActorWorld](http://docs.google.com/info/gridworld/actor/ActorWorld.html) Constructs an actor world with a given grid. [**add(Location, Actor)**](http://docs.google.com/info/gridworld/actor/ActorWorld.html#add(info.gridworld.grid.Location,%20info.gridworld.actor.Actor)) - Method in class info.gridworld.actor.[ActorWorld](http://docs.google.com/info/gridworld/actor/ActorWorld.html) Adds an actor to this world at a given location. [**add(Actor)**](http://docs.google.com/info/gridworld/actor/ActorWorld.html#add(info.gridworld.actor.Actor)) - Method in class info.gridworld.actor.[ActorWorld](http://docs.google.com/info/gridworld/actor/ActorWorld.html) Adds an occupant at a random location. [**add(Location, T)**](http://docs.google.com/info/gridworld/world/World.html#add(info.gridworld.grid.Location,%20T)) - Method in class info.gridworld.world.[World](http://docs.google.com/info/gridworld/world/World.html) Adds an occupant at a given location. [**addGridClass(String)**](http://docs.google.com/info/gridworld/world/World.html#addGridClass(java.lang.String)) - Method in class info.gridworld.world.[World](http://docs.google.com/info/gridworld/world/World.html) Adds a class to be shown in the "Set grid" menu. [**addOccupantClass(String)**](http://docs.google.com/info/gridworld/world/World.html#addOccupantClass(java.lang.String)) - Method in class info.gridworld.world.[World](http://docs.google.com/info/gridworld/world/World.html) Adds a class to be shown when clicking on an empty location. [**AHEAD**](http://docs.google.com/info/gridworld/grid/Location.html#AHEAD) - Static variable in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) The turn angle for making no turn.

## **B**

[**BoundedGrid**](http://docs.google.com/info/gridworld/grid/BoundedGrid.html)<[E](http://docs.google.com/info/gridworld/grid/BoundedGrid.html)> - Class in [info.gridworld.grid](http://docs.google.com/info/gridworld/grid/package-summary.html)A BoundedGrid is a rectangular grid with a finite number of rows and columns.[**BoundedGrid(int, int)**](http://docs.google.com/info/gridworld/grid/BoundedGrid.html#BoundedGrid(int,%20int)) - Constructor for class info.gridworld.grid.[BoundedGrid](http://docs.google.com/info/gridworld/grid/BoundedGrid.html) Constructs an empty bounded grid with the given dimensions. [**Bug**](http://docs.google.com/info/gridworld/actor/Bug.html) - Class in [info.gridworld.actor](http://docs.google.com/info/gridworld/actor/package-summary.html)A Bug is an actor that can move and turn.[**Bug()**](http://docs.google.com/info/gridworld/actor/Bug.html#Bug()) - Constructor for class info.gridworld.actor.[Bug](http://docs.google.com/info/gridworld/actor/Bug.html) Constructs a red bug. [**Bug(Color)**](http://docs.google.com/info/gridworld/actor/Bug.html#Bug(java.awt.Color)) - Constructor for class info.gridworld.actor.[Bug](http://docs.google.com/info/gridworld/actor/Bug.html) Constructs a bug of a given color.

## **C**

[**canMove()**](http://docs.google.com/info/gridworld/actor/Bug.html#canMove()) - Method in class info.gridworld.actor.[Bug](http://docs.google.com/info/gridworld/actor/Bug.html) Tests whether this bug can move forward into a location that is empty or contains a flower. [**compareTo(Object)**](http://docs.google.com/info/gridworld/grid/Location.html#compareTo(java.lang.Object)) - Method in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) Compares this location to other for ordering. [**Critter**](http://docs.google.com/info/gridworld/actor/Critter.html) - Class in [info.gridworld.actor](http://docs.google.com/info/gridworld/actor/package-summary.html)A Critter is an actor that moves through its world, processing other actors in some way and then picking a new location.[**Critter()**](http://docs.google.com/info/gridworld/actor/Critter.html#Critter()) - Constructor for class info.gridworld.actor.[Critter](http://docs.google.com/info/gridworld/actor/Critter.html)  

## **E**

[**EAST**](http://docs.google.com/info/gridworld/grid/Location.html#EAST) - Static variable in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) The compass direction for east. [**equals(Object)**](http://docs.google.com/info/gridworld/grid/Location.html#equals(java.lang.Object)) - Method in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) Indicates whether some other Location object is "equal to" this one.

## **F**

[**Flower**](http://docs.google.com/info/gridworld/actor/Flower.html) - Class in [info.gridworld.actor](http://docs.google.com/info/gridworld/actor/package-summary.html)A Flower is an actor that darkens over time.[**Flower()**](http://docs.google.com/info/gridworld/actor/Flower.html#Flower()) - Constructor for class info.gridworld.actor.[Flower](http://docs.google.com/info/gridworld/actor/Flower.html) Constructs a pink flower. [**Flower(Color)**](http://docs.google.com/info/gridworld/actor/Flower.html#Flower(java.awt.Color)) - Constructor for class info.gridworld.actor.[Flower](http://docs.google.com/info/gridworld/actor/Flower.html) Constructs a flower of a given color. [**FULL\_CIRCLE**](http://docs.google.com/info/gridworld/grid/Location.html#FULL_CIRCLE) - Static variable in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) The turn angle for turning a full circle.

## **G**

[**get(Location)**](http://docs.google.com/info/gridworld/grid/BoundedGrid.html#get(info.gridworld.grid.Location)) - Method in class info.gridworld.grid.[BoundedGrid](http://docs.google.com/info/gridworld/grid/BoundedGrid.html)   [**get(Location)**](http://docs.google.com/info/gridworld/grid/Grid.html#get(info.gridworld.grid.Location)) - Method in interface info.gridworld.grid.[Grid](http://docs.google.com/info/gridworld/grid/Grid.html) Returns the object at a given location in this grid. [**get(Location)**](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html#get(info.gridworld.grid.Location)) - Method in class info.gridworld.grid.[UnboundedGrid](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html)   [**getActors()**](http://docs.google.com/info/gridworld/actor/Critter.html#getActors()) - Method in class info.gridworld.actor.[Critter](http://docs.google.com/info/gridworld/actor/Critter.html) Gets the actors for processing. [**getAdjacentLocation(int)**](http://docs.google.com/info/gridworld/grid/Location.html#getAdjacentLocation(int)) - Method in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) Gets the adjacent location in any one of the eight compass directions. [**getCol()**](http://docs.google.com/info/gridworld/grid/Location.html#getCol()) - Method in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) Gets the column coordinate. [**getColor()**](http://docs.google.com/info/gridworld/actor/Actor.html#getColor()) - Method in class info.gridworld.actor.[Actor](http://docs.google.com/info/gridworld/actor/Actor.html) Gets the color of this actor. [**getDirection()**](http://docs.google.com/info/gridworld/actor/Actor.html#getDirection()) - Method in class info.gridworld.actor.[Actor](http://docs.google.com/info/gridworld/actor/Actor.html) Gets the current direction of this actor. [**getDirectionToward(Location)**](http://docs.google.com/info/gridworld/grid/Location.html#getDirectionToward(info.gridworld.grid.Location)) - Method in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) Returns the direction from this location toward another location. [**getEmptyAdjacentLocations(Location)**](http://docs.google.com/info/gridworld/grid/AbstractGrid.html#getEmptyAdjacentLocations(info.gridworld.grid.Location)) - Method in class info.gridworld.grid.[AbstractGrid](http://docs.google.com/info/gridworld/grid/AbstractGrid.html)   [**getEmptyAdjacentLocations(Location)**](http://docs.google.com/info/gridworld/grid/Grid.html#getEmptyAdjacentLocations(info.gridworld.grid.Location)) - Method in interface info.gridworld.grid.[Grid](http://docs.google.com/info/gridworld/grid/Grid.html) Gets the valid empty locations adjacent to a given location in all eight compass directions (north, northeast, east, southeast, south, southwest, west, and northwest). [**getGrid()**](http://docs.google.com/info/gridworld/actor/Actor.html#getGrid()) - Method in class info.gridworld.actor.[Actor](http://docs.google.com/info/gridworld/actor/Actor.html) Gets the grid in which this actor is located. [**getGrid()**](http://docs.google.com/info/gridworld/world/World.html#getGrid()) - Method in class info.gridworld.world.[World](http://docs.google.com/info/gridworld/world/World.html) Gets the grid managed by this world. [**getGridClasses()**](http://docs.google.com/info/gridworld/world/World.html#getGridClasses()) - Method in class info.gridworld.world.[World](http://docs.google.com/info/gridworld/world/World.html) Gets a set of grid classes that should be used by the world frame for this world. [**getLocation()**](http://docs.google.com/info/gridworld/actor/Actor.html#getLocation()) - Method in class info.gridworld.actor.[Actor](http://docs.google.com/info/gridworld/actor/Actor.html) Gets the location of this actor. [**getMessage()**](http://docs.google.com/info/gridworld/world/World.html#getMessage()) - Method in class info.gridworld.world.[World](http://docs.google.com/info/gridworld/world/World.html) Gets the message to be displayed in the world frame above the grid. [**getMoveLocations()**](http://docs.google.com/info/gridworld/actor/Critter.html#getMoveLocations()) - Method in class info.gridworld.actor.[Critter](http://docs.google.com/info/gridworld/actor/Critter.html) Gets the possible locations for the next move. [**getNeighbors(Location)**](http://docs.google.com/info/gridworld/grid/AbstractGrid.html#getNeighbors(info.gridworld.grid.Location)) - Method in class info.gridworld.grid.[AbstractGrid](http://docs.google.com/info/gridworld/grid/AbstractGrid.html)   [**getNeighbors(Location)**](http://docs.google.com/info/gridworld/grid/Grid.html#getNeighbors(info.gridworld.grid.Location)) - Method in interface info.gridworld.grid.[Grid](http://docs.google.com/info/gridworld/grid/Grid.html) Gets the neighboring occupants in all eight compass directions (north, northeast, east, southeast, south, southwest, west, and northwest). [**getNumCols()**](http://docs.google.com/info/gridworld/grid/BoundedGrid.html#getNumCols()) - Method in class info.gridworld.grid.[BoundedGrid](http://docs.google.com/info/gridworld/grid/BoundedGrid.html)   [**getNumCols()**](http://docs.google.com/info/gridworld/grid/Grid.html#getNumCols()) - Method in interface info.gridworld.grid.[Grid](http://docs.google.com/info/gridworld/grid/Grid.html) Returns the number of columns in this grid. [**getNumCols()**](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html#getNumCols()) - Method in class info.gridworld.grid.[UnboundedGrid](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html)   [**getNumRows()**](http://docs.google.com/info/gridworld/grid/BoundedGrid.html#getNumRows()) - Method in class info.gridworld.grid.[BoundedGrid](http://docs.google.com/info/gridworld/grid/BoundedGrid.html)   [**getNumRows()**](http://docs.google.com/info/gridworld/grid/Grid.html#getNumRows()) - Method in interface info.gridworld.grid.[Grid](http://docs.google.com/info/gridworld/grid/Grid.html) Returns the number of rows in this grid. [**getNumRows()**](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html#getNumRows()) - Method in class info.gridworld.grid.[UnboundedGrid](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html)   [**getOccupantClasses()**](http://docs.google.com/info/gridworld/world/World.html#getOccupantClasses()) - Method in class info.gridworld.world.[World](http://docs.google.com/info/gridworld/world/World.html) Gets a set of occupant classes that should be used by the world frame for this world. [**getOccupiedAdjacentLocations(Location)**](http://docs.google.com/info/gridworld/grid/AbstractGrid.html#getOccupiedAdjacentLocations(info.gridworld.grid.Location)) - Method in class info.gridworld.grid.[AbstractGrid](http://docs.google.com/info/gridworld/grid/AbstractGrid.html)   [**getOccupiedAdjacentLocations(Location)**](http://docs.google.com/info/gridworld/grid/Grid.html#getOccupiedAdjacentLocations(info.gridworld.grid.Location)) - Method in interface info.gridworld.grid.[Grid](http://docs.google.com/info/gridworld/grid/Grid.html) Gets the valid occupied locations adjacent to a given location in all eight compass directions (north, northeast, east, southeast, south, southwest, west, and northwest). [**getOccupiedLocations()**](http://docs.google.com/info/gridworld/grid/BoundedGrid.html#getOccupiedLocations()) - Method in class info.gridworld.grid.[BoundedGrid](http://docs.google.com/info/gridworld/grid/BoundedGrid.html)   [**getOccupiedLocations()**](http://docs.google.com/info/gridworld/grid/Grid.html#getOccupiedLocations()) - Method in interface info.gridworld.grid.[Grid](http://docs.google.com/info/gridworld/grid/Grid.html) Gets the locations in this grid that contain objects. [**getOccupiedLocations()**](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html#getOccupiedLocations()) - Method in class info.gridworld.grid.[UnboundedGrid](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html)   [**getRandomEmptyLocation()**](http://docs.google.com/info/gridworld/world/World.html#getRandomEmptyLocation()) - Method in class info.gridworld.world.[World](http://docs.google.com/info/gridworld/world/World.html) Gets a random empty location in this world. [**getRow()**](http://docs.google.com/info/gridworld/grid/Location.html#getRow()) - Method in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) Gets the row coordinate. [**getValidAdjacentLocations(Location)**](http://docs.google.com/info/gridworld/grid/AbstractGrid.html#getValidAdjacentLocations(info.gridworld.grid.Location)) - Method in class info.gridworld.grid.[AbstractGrid](http://docs.google.com/info/gridworld/grid/AbstractGrid.html)   [**getValidAdjacentLocations(Location)**](http://docs.google.com/info/gridworld/grid/Grid.html#getValidAdjacentLocations(info.gridworld.grid.Location)) - Method in interface info.gridworld.grid.[Grid](http://docs.google.com/info/gridworld/grid/Grid.html) Gets the valid locations adjacent to a given location in all eight compass directions (north, northeast, east, southeast, south, southwest, west, and northwest). [**Grid**](http://docs.google.com/info/gridworld/grid/Grid.html)<[E](http://docs.google.com/info/gridworld/grid/Grid.html)> - Interface in [info.gridworld.grid](http://docs.google.com/info/gridworld/grid/package-summary.html)Grid provides an interface for a two-dimensional, grid-like environment containing arbitrary objects.

## **H**

[**HALF\_CIRCLE**](http://docs.google.com/info/gridworld/grid/Location.html#HALF_CIRCLE) - Static variable in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) The turn angle for turning a half circle. [**HALF\_LEFT**](http://docs.google.com/info/gridworld/grid/Location.html#HALF_LEFT) - Static variable in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) The turn angle for turning 45 degrees to the left. [**HALF\_RIGHT**](http://docs.google.com/info/gridworld/grid/Location.html#HALF_RIGHT) - Static variable in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) The turn angle for turning 45 degrees to the right. [**hashCode()**](http://docs.google.com/info/gridworld/grid/Location.html#hashCode()) - Method in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) Generates a hash code.

## **I**

[**info.gridworld.actor**](http://docs.google.com/info/gridworld/actor/package-summary.html) - package info.gridworld.actor [**info.gridworld.grid**](http://docs.google.com/info/gridworld/grid/package-summary.html) - package info.gridworld.grid [**info.gridworld.world**](http://docs.google.com/info/gridworld/world/package-summary.html) - package info.gridworld.world [**isValid(Location)**](http://docs.google.com/info/gridworld/grid/BoundedGrid.html#isValid(info.gridworld.grid.Location)) - Method in class info.gridworld.grid.[BoundedGrid](http://docs.google.com/info/gridworld/grid/BoundedGrid.html)   [**isValid(Location)**](http://docs.google.com/info/gridworld/grid/Grid.html#isValid(info.gridworld.grid.Location)) - Method in interface info.gridworld.grid.[Grid](http://docs.google.com/info/gridworld/grid/Grid.html) Checks whether a location is valid in this grid. [**isValid(Location)**](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html#isValid(info.gridworld.grid.Location)) - Method in class info.gridworld.grid.[UnboundedGrid](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html)  

## **K**

[**keyPressed(String, Location)**](http://docs.google.com/info/gridworld/world/World.html#keyPressed(java.lang.String,%20info.gridworld.grid.Location)) - Method in class info.gridworld.world.[World](http://docs.google.com/info/gridworld/world/World.html) This method is called when a key was pressed.

## **L**

[**LEFT**](http://docs.google.com/info/gridworld/grid/Location.html#LEFT) - Static variable in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) The turn angle for turning 90 degrees to the left. [**Location**](http://docs.google.com/info/gridworld/grid/Location.html) - Class in [info.gridworld.grid](http://docs.google.com/info/gridworld/grid/package-summary.html)A Location object represents the row and column of a location in a two-dimensional grid.[**Location(int, int)**](http://docs.google.com/info/gridworld/grid/Location.html#Location(int,%20int)) - Constructor for class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) Constructs a location with given row and column coordinates. [**locationClicked(Location)**](http://docs.google.com/info/gridworld/world/World.html#locationClicked(info.gridworld.grid.Location)) - Method in class info.gridworld.world.[World](http://docs.google.com/info/gridworld/world/World.html) This method is called when the user clicks on a location in the WorldFrame.

## **M**

[**makeMove(Location)**](http://docs.google.com/info/gridworld/actor/Critter.html#makeMove(info.gridworld.grid.Location)) - Method in class info.gridworld.actor.[Critter](http://docs.google.com/info/gridworld/actor/Critter.html) Moves this critter to the given location. [**move()**](http://docs.google.com/info/gridworld/actor/Bug.html#move()) - Method in class info.gridworld.actor.[Bug](http://docs.google.com/info/gridworld/actor/Bug.html) Moves the bug forward, putting a flower into the location it previously occupied. [**moveTo(Location)**](http://docs.google.com/info/gridworld/actor/Actor.html#moveTo(info.gridworld.grid.Location)) - Method in class info.gridworld.actor.[Actor](http://docs.google.com/info/gridworld/actor/Actor.html) Moves this actor to a new location.

## **N**

[**NORTH**](http://docs.google.com/info/gridworld/grid/Location.html#NORTH) - Static variable in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) The compass direction for north. [**NORTHEAST**](http://docs.google.com/info/gridworld/grid/Location.html#NORTHEAST) - Static variable in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) The compass direction for northeast. [**NORTHWEST**](http://docs.google.com/info/gridworld/grid/Location.html#NORTHWEST) - Static variable in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) The compass direction for northwest.

## **P**

[**processActors(ArrayList<Actor>)**](http://docs.google.com/info/gridworld/actor/Critter.html#processActors(java.util.ArrayList)) - Method in class info.gridworld.actor.[Critter](http://docs.google.com/info/gridworld/actor/Critter.html) Processes the actors. [**put(Location, E)**](http://docs.google.com/info/gridworld/grid/BoundedGrid.html#put(info.gridworld.grid.Location,%20E)) - Method in class info.gridworld.grid.[BoundedGrid](http://docs.google.com/info/gridworld/grid/BoundedGrid.html)   [**put(Location, E)**](http://docs.google.com/info/gridworld/grid/Grid.html#put(info.gridworld.grid.Location,%20E)) - Method in interface info.gridworld.grid.[Grid](http://docs.google.com/info/gridworld/grid/Grid.html) Puts an object at a given location in this grid. [**put(Location, E)**](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html#put(info.gridworld.grid.Location,%20E)) - Method in class info.gridworld.grid.[UnboundedGrid](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html)   [**putSelfInGrid(Grid<Actor>, Location)**](http://docs.google.com/info/gridworld/actor/Actor.html#putSelfInGrid(info.gridworld.grid.Grid,%20info.gridworld.grid.Location)) - Method in class info.gridworld.actor.[Actor](http://docs.google.com/info/gridworld/actor/Actor.html) Puts this actor into a grid.

## **R**

[**remove(Location)**](http://docs.google.com/info/gridworld/actor/ActorWorld.html#remove(info.gridworld.grid.Location)) - Method in class info.gridworld.actor.[ActorWorld](http://docs.google.com/info/gridworld/actor/ActorWorld.html) Removes an actor from this world. [**remove(Location)**](http://docs.google.com/info/gridworld/grid/BoundedGrid.html#remove(info.gridworld.grid.Location)) - Method in class info.gridworld.grid.[BoundedGrid](http://docs.google.com/info/gridworld/grid/BoundedGrid.html)   [**remove(Location)**](http://docs.google.com/info/gridworld/grid/Grid.html#remove(info.gridworld.grid.Location)) - Method in interface info.gridworld.grid.[Grid](http://docs.google.com/info/gridworld/grid/Grid.html) Removes the object at a given location from this grid. [**remove(Location)**](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html#remove(info.gridworld.grid.Location)) - Method in class info.gridworld.grid.[UnboundedGrid](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html)   [**remove(Location)**](http://docs.google.com/info/gridworld/world/World.html#remove(info.gridworld.grid.Location)) - Method in class info.gridworld.world.[World](http://docs.google.com/info/gridworld/world/World.html) Removes an occupant from a given location. [**removeSelfFromGrid()**](http://docs.google.com/info/gridworld/actor/Actor.html#removeSelfFromGrid()) - Method in class info.gridworld.actor.[Actor](http://docs.google.com/info/gridworld/actor/Actor.html) Removes this actor from its grid. [**RIGHT**](http://docs.google.com/info/gridworld/grid/Location.html#RIGHT) - Static variable in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) The turn angle for turning 90 degrees to the right. [**Rock**](http://docs.google.com/info/gridworld/actor/Rock.html) - Class in [info.gridworld.actor](http://docs.google.com/info/gridworld/actor/package-summary.html)A Rock is an actor that does nothing.[**Rock()**](http://docs.google.com/info/gridworld/actor/Rock.html#Rock()) - Constructor for class info.gridworld.actor.[Rock](http://docs.google.com/info/gridworld/actor/Rock.html) Constructs a black rock. [**Rock(Color)**](http://docs.google.com/info/gridworld/actor/Rock.html#Rock(java.awt.Color)) - Constructor for class info.gridworld.actor.[Rock](http://docs.google.com/info/gridworld/actor/Rock.html) Constructs a rock of a given color.

## **S**

[**selectMoveLocation(ArrayList<Location>)**](http://docs.google.com/info/gridworld/actor/Critter.html#selectMoveLocation(java.util.ArrayList)) - Method in class info.gridworld.actor.[Critter](http://docs.google.com/info/gridworld/actor/Critter.html) Selects the location for the next move. [**setColor(Color)**](http://docs.google.com/info/gridworld/actor/Actor.html#setColor(java.awt.Color)) - Method in class info.gridworld.actor.[Actor](http://docs.google.com/info/gridworld/actor/Actor.html) Sets the color of this actor. [**setDirection(int)**](http://docs.google.com/info/gridworld/actor/Actor.html#setDirection(int)) - Method in class info.gridworld.actor.[Actor](http://docs.google.com/info/gridworld/actor/Actor.html) Sets the current direction of this actor. [**setGrid(Grid<T>)**](http://docs.google.com/info/gridworld/world/World.html#setGrid(info.gridworld.grid.Grid)) - Method in class info.gridworld.world.[World](http://docs.google.com/info/gridworld/world/World.html) Sets the grid managed by this world. [**setMessage(String)**](http://docs.google.com/info/gridworld/world/World.html#setMessage(java.lang.String)) - Method in class info.gridworld.world.[World](http://docs.google.com/info/gridworld/world/World.html) Sets the message to be displayed in the world frame above the grid. [**show()**](http://docs.google.com/info/gridworld/actor/ActorWorld.html#show()) - Method in class info.gridworld.actor.[ActorWorld](http://docs.google.com/info/gridworld/actor/ActorWorld.html)   [**show()**](http://docs.google.com/info/gridworld/world/World.html#show()) - Method in class info.gridworld.world.[World](http://docs.google.com/info/gridworld/world/World.html) Constructs and shows a frame for this world. [**SOUTH**](http://docs.google.com/info/gridworld/grid/Location.html#SOUTH) - Static variable in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) The compass direction for south. [**SOUTHEAST**](http://docs.google.com/info/gridworld/grid/Location.html#SOUTHEAST) - Static variable in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) The compass direction for southeast. [**SOUTHWEST**](http://docs.google.com/info/gridworld/grid/Location.html#SOUTHWEST) - Static variable in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) The compass direction for southwest. [**step()**](http://docs.google.com/info/gridworld/actor/ActorWorld.html#step()) - Method in class info.gridworld.actor.[ActorWorld](http://docs.google.com/info/gridworld/actor/ActorWorld.html)   [**step()**](http://docs.google.com/info/gridworld/world/World.html#step()) - Method in class info.gridworld.world.[World](http://docs.google.com/info/gridworld/world/World.html) This method is called when the user clicks on the step button, or when run mode has been activated by clicking the run button.

## **T**

[**toString()**](http://docs.google.com/info/gridworld/actor/Actor.html#toString()) - Method in class info.gridworld.actor.[Actor](http://docs.google.com/info/gridworld/actor/Actor.html) Creates a string that describes this actor. [**toString()**](http://docs.google.com/info/gridworld/grid/AbstractGrid.html#toString()) - Method in class info.gridworld.grid.[AbstractGrid](http://docs.google.com/info/gridworld/grid/AbstractGrid.html) Creates a string that describes this grid. [**toString()**](http://docs.google.com/info/gridworld/grid/Location.html#toString()) - Method in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) Creates a string that describes this location. [**toString()**](http://docs.google.com/info/gridworld/world/World.html#toString()) - Method in class info.gridworld.world.[World](http://docs.google.com/info/gridworld/world/World.html) Returns a string that shows the positions of the grid occupants. [**turn()**](http://docs.google.com/info/gridworld/actor/Bug.html#turn()) - Method in class info.gridworld.actor.[Bug](http://docs.google.com/info/gridworld/actor/Bug.html) Turns the bug 45 degrees to the right without changing its location.

## **U**

[**UnboundedGrid**](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html)<[E](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html)> - Class in [info.gridworld.grid](http://docs.google.com/info/gridworld/grid/package-summary.html)An UnboundedGrid is a rectangular grid with an unbounded number of rows and columns.[**UnboundedGrid()**](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html#UnboundedGrid()) - Constructor for class info.gridworld.grid.[UnboundedGrid](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html) Constructs an empty unbounded grid.

## **W**

[**WEST**](http://docs.google.com/info/gridworld/grid/Location.html#WEST) - Static variable in class info.gridworld.grid.[Location](http://docs.google.com/info/gridworld/grid/Location.html) The compass direction for west. [**World**](http://docs.google.com/info/gridworld/world/World.html)<[T](http://docs.google.com/info/gridworld/world/World.html)> - Class in [info.gridworld.world](http://docs.google.com/info/gridworld/world/package-summary.html)A World is the mediator between a grid and the GridWorld GUI.[**World()**](http://docs.google.com/info/gridworld/world/World.html#World()) - Constructor for class info.gridworld.world.[World](http://docs.google.com/info/gridworld/world/World.html)   [**World(Grid<T>)**](http://docs.google.com/info/gridworld/world/World.html#World(info.gridworld.grid.Grid)) - Constructor for class info.gridworld.world.[World](http://docs.google.com/info/gridworld/world/World.html)  [A](#3znysh7) [B](#2et92p0) [C](#tyjcwt) [E](#3dy6vkm) [F](#1t3h5sf) [G](#4d34og8) [H](#2s8eyo1) [I](#17dp8vu) [K](#3rdcrjn) [L](#26in1rg) [M](#lnxbz9) [N](#35nkun2) [P](#1ksv4uv) [R](#44sinio) [S](#2jxsxqh) [T](#z337ya) [U](#3j2qqm3) [W](#1y810tw)

| | [**Overview**](http://docs.google.com/overview-summary.html) | Package | Class | [**Tree**](http://docs.google.com/overview-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | **Index** | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | | |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| PREV   NEXT | [**FRAMES**](http://docs.google.com/index.html?index-all.html)    [**NO FRAMES**](http://docs.google.com/index-all.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |