| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-all.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | | |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/info/gridworld/grid/Grid.html)   [**NEXT CLASS**](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html) | [**FRAMES**](http://docs.google.com/index.html?info/gridworld/grid/Location.html)    [**NO FRAMES**](http://docs.google.com/Location.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#1t3h5sf) | [CONSTR](#1ci93xb) | [METHOD](#2bn6wsx) |

## **info.gridworld.grid**

Class Location

[java.lang.Object](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true)  
 **info.gridworld.grid.Location**

**All Implemented Interfaces:** [Comparable](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Comparable.html?is-external=true)

public class **Location**extends [Object](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true)implements [Comparable](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Comparable.html?is-external=true)

A Location object represents the row and column of a location in a two-dimensional grid.

The API of this class is testable on the AP CS A and AB exams.

| **Field Summary** | |
| --- | --- |
| static int | [**AHEAD**](http://docs.google.com/info/gridworld/grid/Location.html#AHEAD)            The turn angle for making no turn. |
| static int | [**EAST**](http://docs.google.com/info/gridworld/grid/Location.html#EAST)            The compass direction for east. |
| static int | [**FULL\_CIRCLE**](http://docs.google.com/info/gridworld/grid/Location.html#FULL_CIRCLE)            The turn angle for turning a full circle. |
| static int | [**HALF\_CIRCLE**](http://docs.google.com/info/gridworld/grid/Location.html#HALF_CIRCLE)            The turn angle for turning a half circle. |
| static int | [**HALF\_LEFT**](http://docs.google.com/info/gridworld/grid/Location.html#HALF_LEFT)            The turn angle for turning 45 degrees to the left. |
| static int | [**HALF\_RIGHT**](http://docs.google.com/info/gridworld/grid/Location.html#HALF_RIGHT)            The turn angle for turning 45 degrees to the right. |
| static int | [**LEFT**](http://docs.google.com/info/gridworld/grid/Location.html#LEFT)            The turn angle for turning 90 degrees to the left. |
| static int | [**NORTH**](http://docs.google.com/info/gridworld/grid/Location.html#NORTH)            The compass direction for north. |
| static int | [**NORTHEAST**](http://docs.google.com/info/gridworld/grid/Location.html#NORTHEAST)            The compass direction for northeast. |
| static int | [**NORTHWEST**](http://docs.google.com/info/gridworld/grid/Location.html#NORTHWEST)            The compass direction for northwest. |
| static int | [**RIGHT**](http://docs.google.com/info/gridworld/grid/Location.html#RIGHT)            The turn angle for turning 90 degrees to the right. |
| static int | [**SOUTH**](http://docs.google.com/info/gridworld/grid/Location.html#SOUTH)            The compass direction for south. |
| static int | [**SOUTHEAST**](http://docs.google.com/info/gridworld/grid/Location.html#SOUTHEAST)            The compass direction for southeast. |
| static int | [**SOUTHWEST**](http://docs.google.com/info/gridworld/grid/Location.html#SOUTHWEST)            The compass direction for southwest. |
| static int | [**WEST**](http://docs.google.com/info/gridworld/grid/Location.html#WEST)            The compass direction for west. |

| **Constructor Summary** | |
| --- | --- |
| [**Location**](http://docs.google.com/info/gridworld/grid/Location.html#Location(int,%20int))(int r, int c)            Constructs a location with given row and column coordinates. |

| **Method Summary** | |
| --- | --- |
| int | [**compareTo**](http://docs.google.com/info/gridworld/grid/Location.html#compareTo(java.lang.Object))([Object](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true) other)            Compares this location to other for ordering. |
| boolean | [**equals**](http://docs.google.com/info/gridworld/grid/Location.html#equals(java.lang.Object))([Object](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true) other)            Indicates whether some other Location object is "equal to" this one. |
| [Location](http://docs.google.com/info/gridworld/grid/Location.html) | [**getAdjacentLocation**](http://docs.google.com/info/gridworld/grid/Location.html#getAdjacentLocation(int))(int direction)            Gets the adjacent location in any one of the eight compass directions. |
| int | [**getCol**](http://docs.google.com/info/gridworld/grid/Location.html#getCol())()            Gets the column coordinate. |
| int | [**getDirectionToward**](http://docs.google.com/info/gridworld/grid/Location.html#getDirectionToward(info.gridworld.grid.Location))([Location](http://docs.google.com/info/gridworld/grid/Location.html) target)            Returns the direction from this location toward another location. |
| int | [**getRow**](http://docs.google.com/info/gridworld/grid/Location.html#getRow())()            Gets the row coordinate. |
| int | [**hashCode**](http://docs.google.com/info/gridworld/grid/Location.html#hashCode())()            Generates a hash code. |
| [String](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/String.html?is-external=true) | [**toString**](http://docs.google.com/info/gridworld/grid/Location.html#toString())()            Creates a string that describes this location. |

| **Methods inherited from class java.lang.**[**Object**](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true) |
| --- |
| [clone](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#clone()), [finalize](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#finalize()), [getClass](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#getClass()), [notify](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#notify()), [notifyAll](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#notifyAll()), [wait](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#wait()), [wait](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#wait(long)), [wait](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#wait(long,%20int)) |

| **Field Detail** |
| --- |

### LEFT

public static final int **LEFT**

The turn angle for turning 90 degrees to the left.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#info.gridworld.grid.Location.LEFT)

### RIGHT

public static final int **RIGHT**

The turn angle for turning 90 degrees to the right.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#info.gridworld.grid.Location.RIGHT)

### HALF\_LEFT

public static final int **HALF\_LEFT**

The turn angle for turning 45 degrees to the left.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#info.gridworld.grid.Location.HALF_LEFT)

### HALF\_RIGHT

public static final int **HALF\_RIGHT**

The turn angle for turning 45 degrees to the right.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#info.gridworld.grid.Location.HALF_RIGHT)

### FULL\_CIRCLE

public static final int **FULL\_CIRCLE**

The turn angle for turning a full circle.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#info.gridworld.grid.Location.FULL_CIRCLE)

### HALF\_CIRCLE

public static final int **HALF\_CIRCLE**

The turn angle for turning a half circle.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#info.gridworld.grid.Location.HALF_CIRCLE)

### AHEAD

public static final int **AHEAD**

The turn angle for making no turn.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#info.gridworld.grid.Location.AHEAD)

### NORTH

public static final int **NORTH**

The compass direction for north.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#info.gridworld.grid.Location.NORTH)

### NORTHEAST

public static final int **NORTHEAST**

The compass direction for northeast.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#info.gridworld.grid.Location.NORTHEAST)

### EAST

public static final int **EAST**

The compass direction for east.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#info.gridworld.grid.Location.EAST)

### SOUTHEAST

public static final int **SOUTHEAST**

The compass direction for southeast.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#info.gridworld.grid.Location.SOUTHEAST)

### SOUTH

public static final int **SOUTH**

The compass direction for south.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#info.gridworld.grid.Location.SOUTH)

### SOUTHWEST

public static final int **SOUTHWEST**

The compass direction for southwest.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#info.gridworld.grid.Location.SOUTHWEST)

### WEST

public static final int **WEST**

The compass direction for west.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#info.gridworld.grid.Location.WEST)

### NORTHWEST

public static final int **NORTHWEST**

The compass direction for northwest.

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#info.gridworld.grid.Location.NORTHWEST)

| **Constructor Detail** |
| --- |

### Location

public **Location**(int r,  
 int c)

Constructs a location with given row and column coordinates.

**Parameters:**r - the rowc - the column

| **Method Detail** |
| --- |

### getRow

public int **getRow**()

Gets the row coordinate.

**Returns:**the row of this location

### getCol

public int **getCol**()

Gets the column coordinate.

**Returns:**the column of this location

### getAdjacentLocation

public [Location](http://docs.google.com/info/gridworld/grid/Location.html) **getAdjacentLocation**(int direction)

Gets the adjacent location in any one of the eight compass directions.

**Parameters:**direction - the direction in which to find a neighbor location **Returns:**the adjacent location in the direction that is closest to direction

### getDirectionToward

public int **getDirectionToward**([Location](http://docs.google.com/info/gridworld/grid/Location.html) target)

Returns the direction from this location toward another location. The direction is rounded to the nearest compass direction.

**Parameters:**target - another location **Returns:**the closest compass direction from this location toward target

### equals

public boolean **equals**([Object](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true) other)

Indicates whether some other Location object is "equal to" this one.

**Overrides:**[equals](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#equals(java.lang.Object)) in class [Object](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true) **Parameters:**other - the other location to test **Returns:**true if other is a Location with the same row and column as this location; false otherwise

### hashCode

public int **hashCode**()

Generates a hash code.

**Overrides:**[hashCode](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#hashCode()) in class [Object](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true) **Returns:**a hash code for this location

### compareTo

public int **compareTo**([Object](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true) other)

Compares this location to other for ordering. Returns a negative integer, zero, or a positive integer as this location is less than, equal to, or greater than other. Locations are ordered in row-major order.

(Precondition: other is a Location object.)

**Specified by:**[compareTo](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Comparable.html?is-external=true#compareTo(T)) in interface [Comparable](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Comparable.html?is-external=true) **Parameters:**other - the other location to test **Returns:**a negative integer if this location is less than other, zero if the two locations are equal, or a positive integer if this location is greater than other

### toString

public [String](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/String.html?is-external=true) **toString**()

Creates a string that describes this location.

**Overrides:**[toString](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true#toString()) in class [Object](http://java.sun.com/j2se/1.5.0/docs/api/java/lang/Object.html?is-external=true) **Returns:**a string with the row and column of this location, in the format (row, col)

| | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-all.html) | [**Help**](http://docs.google.com/help-doc.html) | | --- | --- | --- | --- | --- | --- | --- | | |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| [**PREV CLASS**](http://docs.google.com/info/gridworld/grid/Grid.html)   [**NEXT CLASS**](http://docs.google.com/info/gridworld/grid/UnboundedGrid.html) | [**FRAMES**](http://docs.google.com/index.html?info/gridworld/grid/Location.html)    [**NO FRAMES**](http://docs.google.com/Location.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#1t3h5sf) | [CONSTR](#1ci93xb) | [METHOD](#2bn6wsx) |