|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Fish.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | |  |
| [**PREV CLASS**](http://docs.google.com/fishPond/Controller.html)   [**NEXT CLASS**](http://docs.google.com/fishPond/IllegalFishDirectionException.html) | [**FRAMES**](http://docs.google.com/index.html?fishPond/Fish.html)    [**NO FRAMES**](http://docs.google.com/Fish.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#1t3h5sf) | [CONSTR](#35nkun2) | [METHOD](#2jxsxqh) |

## **fishPond**

Class Fish

java.lang.Object  
  
 **fishPond.Fish**

public class **Fish**extends java.lang.Object

The state of a fish consists of its position in the pond (row and column), it's size, and the direction in which it is moving (UP, DOWN, LEFT, or RIGHT).

A fish moves, eats other fish, eats plants, and shrinks over time.

STUDENTS MAY NOT ADD ANY FIELDS OR PUBLIC METHODS! (You may add private methods of your own, if you wish.)

**Author:** Put Your Name Here

|  |  |
| --- | --- |
| **Field Summary** | |
| static int | [**DOWN**](http://docs.google.com/fishPond/Fish.html#DOWN)            Code for "Down" fish direction |
| static int | [**FISH\_STARTING\_SIZE**](http://docs.google.com/fishPond/Fish.html#FISH_STARTING_SIZE)            Initial size of each fish when simulation begins |
| static int | [**LEFT**](http://docs.google.com/fishPond/Fish.html#LEFT)            Code for "Left" fish direction |
| static int | [**MAX\_FISH\_SIZE**](http://docs.google.com/fishPond/Fish.html#MAX_FISH_SIZE)            Maximum size for a fish |
| static int | [**RIGHT**](http://docs.google.com/fishPond/Fish.html#RIGHT)            Code for "Right" fish direction |
| static int | [**UP**](http://docs.google.com/fishPond/Fish.html#UP)            Code for "Up" fish direction |

|  |  |
| --- | --- |
| **Constructor Summary** | |
| [**Fish**](http://docs.google.com/fishPond/Fish.html#Fish(fishPond.Fish))([Fish](http://docs.google.com/fishPond/Fish.html) other)            Standard copy constructor -- just copies the fields |
| [**Fish**](http://docs.google.com/fishPond/Fish.html#Fish(int,%20int,%20int,%20int))(int rowIn, int colIn, int size, int direction)            Simply initializes the state of the fish with these parameters |

|  |  |
| --- | --- |
| **Method Summary** | |
| void | [**eat**](http://docs.google.com/fishPond/Fish.html#eat(int))(int nutritionalValue)            Fish size increased by nutritionalValue. |
| void | [**fight**](http://docs.google.com/fishPond/Fish.html#fight(fishPond.Fish))([Fish](http://docs.google.com/fishPond/Fish.html) other)            The current object battles the parameter (other). |
| int | [**getCol**](http://docs.google.com/fishPond/Fish.html#getCol())()            Returns column |
| int | [**getDirection**](http://docs.google.com/fishPond/Fish.html#getDirection())()            Returns direction (UP, DOWN, LEFT, or RIGHT) |
| int | [**getRow**](http://docs.google.com/fishPond/Fish.html#getRow())()            Returns row |
| int | [**getSize**](http://docs.google.com/fishPond/Fish.html#getSize())()            Returns size |
| boolean | [**isAlive**](http://docs.google.com/fishPond/Fish.html#isAlive())()            Returns true if size is greater than zero, false otherwise |
| void | [**move**](http://docs.google.com/fishPond/Fish.html#move())()            The fish's location (row or col) is adjusted by ONE unit, depending on the fish's current direction. |
| void | [**setRandomDirection**](http://docs.google.com/fishPond/Fish.html#setRandomDirection())()            The fish's direction is randomly determined (UP, DOWN, LEFT or RIGHT). |
| void | [**shrink**](http://docs.google.com/fishPond/Fish.html#shrink())()            Size is decreased by TWO. |
| java.lang.String | [**toString**](http://docs.google.com/fishPond/Fish.html#toString())()            Implement this however you want -- it's for your own purposes while debugging |

|  |
| --- |
| **Methods inherited from class java.lang.Object** |
| equals, getClass, hashCode, notify, notifyAll, wait, wait, wait |

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

### FISH\_STARTING\_SIZE

public static final int **FISH\_STARTING\_SIZE**

Initial size of each fish when simulation begins

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#fishPond.Fish.FISH_STARTING_SIZE)

### MAX\_FISH\_SIZE

public static final int **MAX\_FISH\_SIZE**

Maximum size for a fish

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#fishPond.Fish.MAX_FISH_SIZE)

### LEFT

public static final int **LEFT**

Code for "Left" fish direction

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#fishPond.Fish.LEFT)

### RIGHT

public static final int **RIGHT**

Code for "Right" fish direction

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#fishPond.Fish.RIGHT)

### UP

public static final int **UP**

Code for "Up" fish direction

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#fishPond.Fish.UP)

### DOWN

public static final int **DOWN**

Code for "Down" fish direction

**See Also:**[Constant Field Values](http://docs.google.com/constant-values.html#fishPond.Fish.DOWN)

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

### Fish

public **Fish**(int rowIn,  
  
 int colIn,  
  
 int size,  
  
 int direction)

Simply initializes the state of the fish with these parameters

### Fish

public **Fish**([Fish](http://docs.google.com/fishPond/Fish.html) other)

Standard copy constructor -- just copies the fields

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

### eat

public void **eat**(int nutritionalValue)

Fish size increased by nutritionalValue.

### isAlive

public boolean **isAlive**()

Returns true if size is greater than zero, false otherwise

### shrink

public void **shrink**()

Size is decreased by TWO.

### toString

public java.lang.String **toString**()

Implement this however you want -- it's for your own purposes while debugging

**Overrides:**toString in class java.lang.Object

### fight

public void **fight**([Fish](http://docs.google.com/fishPond/Fish.html) other)

The current object battles the parameter (other). Whichever one is larger eats the other by calling the private "eat" method. In cases where the sizes of the two fish are exactly the same, have the current object win!

### move

public void **move**()

The fish's location (row or col) is adjusted by ONE unit, depending on the fish's current direction. For example, if the current direction is "UP", then the fish's row should be decremented.

If the fish's current direction is not equal to one of the static constants UP, DOWN, LEFT, or RIGHT, then this method will throw an IllegalFishDirectionException, passing the fish's direction to the constructor. STUDENTS: You are required to use a switch statement when implementing this method.

### setRandomDirection

public void **setRandomDirection**()

The fish's direction is randomly determined (UP, DOWN, LEFT or RIGHT). Sometimes the resulting direction will be the same as the original one.

YOU MUST FOLLOW THE INSTRUCTIONS BELOW OR YOU WILL NOT PASS OUR TESTS!

Call Random131.getRandomInteger(4).

If the value is 0, set the direction to UP. If 1, set to DOWN. If 2, set to LEFT. If 3, set to RIGHT. IMPORTANT: DO NOT SET THE DIRECTION TO THE VALUES 0, 1, 2, OR 3 -- directions must be set using the static constants (UP, DOWN, LEFT, RIGHT).

### getSize

public int **getSize**()

Returns size

### getRow

public int **getRow**()

Returns row

### getCol

public int **getCol**()

Returns column

### getDirection

public int **getDirection**()

Returns direction (UP, DOWN, LEFT, or RIGHT)

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | [**Overview**](http://docs.google.com/overview-summary.html) | [**Package**](http://docs.google.com/package-summary.html) | **Class** | [**Use**](http://docs.google.com/class-use/Fish.html) | [**Tree**](http://docs.google.com/package-tree.html) | [**Deprecated**](http://docs.google.com/deprecated-list.html) | [**Index**](http://docs.google.com/index-files/index-1.html) | [**Help**](http://docs.google.com/help-doc.html) | | |  |
| [**PREV CLASS**](http://docs.google.com/fishPond/Controller.html)   [**NEXT CLASS**](http://docs.google.com/fishPond/IllegalFishDirectionException.html) | [**FRAMES**](http://docs.google.com/index.html?fishPond/Fish.html)    [**NO FRAMES**](http://docs.google.com/Fish.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | [FIELD](#3znysh7) | [CONSTR](#2et92p0) | [METHOD](#tyjcwt) | DETAIL: [FIELD](#1t3h5sf) | [CONSTR](#35nkun2) | [METHOD](#2jxsxqh) |

[Web Accessibility](https://www.umd.edu/web-accessibility)