| | [**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/DatabaseManager.html)   [**NEXT CLASS**](http://docs.google.com/DigitalPicture.html) | [**FRAMES**](http://docs.google.com/index.html?Deer.html)    [**NO FRAMES**](http://docs.google.com/Deer.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#1t3h5sf) | [METHOD](#17dp8vu) |

## Class Deer

[java.lang.Object](http://java.sun.com/javase/6/docs/api/java/lang/Object.html?is-external=true)  
 [SimpleTurtle](http://docs.google.com/SimpleTurtle.html)  
 [Turtle](http://docs.google.com/Turtle.html)  
 **Deer**

public class **Deer**extends [Turtle](http://docs.google.com/Turtle.html)

Class that represents a deer. The deer class tracks all living deer with a linked list.

**Author:** Barb Ericson ericson@cc.gatech.edu

| **Constructor Summary** | |
| --- | --- |
| [**Deer**](http://docs.google.com/Deer.html#Deer(int,%20int,%20ModelDisplay))(int x, int y, [ModelDisplay](http://docs.google.com/ModelDisplay.html) modelDisplayer)            Constructor that takes the x and y and a model display to draw it on |
| [**Deer**](http://docs.google.com/Deer.html#Deer(ModelDisplay))([ModelDisplay](http://docs.google.com/ModelDisplay.html) modelDisplayer)            Constructor that takes the model display (the original position will be randomally assigned |

| **Method Summary** | |
| --- | --- |
| void | [**act**](http://docs.google.com/Deer.html#act())()            Method to act during a time step pick a random direction and move some random amount up to top speed |
| void | [**die**](http://docs.google.com/Deer.html#die())()            Method that handles when a deer dies |
| static [Deer](http://docs.google.com/Deer.html) | [**getHead**](http://docs.google.com/Deer.html#getHead())()            Method to get the head of the linked list of deer |
| [Deer](http://docs.google.com/Deer.html) | [**getNext**](http://docs.google.com/Deer.html#getNext())()            Method to get the next Deer in the linked list |
| void | [**init**](http://docs.google.com/Deer.html#init())()            Method to initialize the new deer object |
| static void | [**main**](http://docs.google.com/Deer.html#main(java.lang.String%5B%5D))([String](http://java.sun.com/javase/6/docs/api/java/lang/String.html?is-external=true)[] args)            main method for testing |

| **Methods inherited from class** [**SimpleTurtle**](http://docs.google.com/SimpleTurtle.html) |
| --- |
| [backward](http://docs.google.com/SimpleTurtle.html#backward()), [backward](http://docs.google.com/SimpleTurtle.html#backward(int)), [clearPath](http://docs.google.com/SimpleTurtle.html#clearPath()), [drawInfoString](http://docs.google.com/SimpleTurtle.html#drawInfoString(java.awt.Graphics)), [drop](http://docs.google.com/SimpleTurtle.html#drop(Picture)), [forward](http://docs.google.com/SimpleTurtle.html#forward()), [forward](http://docs.google.com/SimpleTurtle.html#forward(int)), [getBodyColor](http://docs.google.com/SimpleTurtle.html#getBodyColor()), [getDistance](http://docs.google.com/SimpleTurtle.html#getDistance(int,%20int)), [getHeading](http://docs.google.com/SimpleTurtle.html#getHeading()), [getHeight](http://docs.google.com/SimpleTurtle.html#getHeight()), [getInfoColor](http://docs.google.com/SimpleTurtle.html#getInfoColor()), [getModelDisplay](http://docs.google.com/SimpleTurtle.html#getModelDisplay()), [getName](http://docs.google.com/SimpleTurtle.html#getName()), [getPen](http://docs.google.com/SimpleTurtle.html#getPen()), [getPenColor](http://docs.google.com/SimpleTurtle.html#getPenColor()), [getPenWidth](http://docs.google.com/SimpleTurtle.html#getPenWidth()), [getPicture](http://docs.google.com/SimpleTurtle.html#getPicture()), [getShellColor](http://docs.google.com/SimpleTurtle.html#getShellColor()), [getShowInfo](http://docs.google.com/SimpleTurtle.html#getShowInfo()), [getWidth](http://docs.google.com/SimpleTurtle.html#getWidth()), [getXPos](http://docs.google.com/SimpleTurtle.html#getXPos()), [getYPos](http://docs.google.com/SimpleTurtle.html#getYPos()), [hide](http://docs.google.com/SimpleTurtle.html#hide()), [isPenDown](http://docs.google.com/SimpleTurtle.html#isPenDown()), [isVisible](http://docs.google.com/SimpleTurtle.html#isVisible()), [moveTo](http://docs.google.com/SimpleTurtle.html#moveTo(int,%20int)), [paintComponent](http://docs.google.com/SimpleTurtle.html#paintComponent(java.awt.Graphics)), [penDown](http://docs.google.com/SimpleTurtle.html#penDown()), [penUp](http://docs.google.com/SimpleTurtle.html#penUp()), [setBodyColor](http://docs.google.com/SimpleTurtle.html#setBodyColor(java.awt.Color)), [setColor](http://docs.google.com/SimpleTurtle.html#setColor(java.awt.Color)), [setHeading](http://docs.google.com/SimpleTurtle.html#setHeading(double)), [setHeight](http://docs.google.com/SimpleTurtle.html#setHeight(int)), [setInfoColor](http://docs.google.com/SimpleTurtle.html#setInfoColor(java.awt.Color)), [setModelDisplay](http://docs.google.com/SimpleTurtle.html#setModelDisplay(ModelDisplay)), [setName](http://docs.google.com/SimpleTurtle.html#setName(java.lang.String)), [setPen](http://docs.google.com/SimpleTurtle.html#setPen(Pen)), [setPenColor](http://docs.google.com/SimpleTurtle.html#setPenColor(java.awt.Color)), [setPenDown](http://docs.google.com/SimpleTurtle.html#setPenDown(boolean)), [setPenWidth](http://docs.google.com/SimpleTurtle.html#setPenWidth(int)), [setPicture](http://docs.google.com/SimpleTurtle.html#setPicture(Picture)), [setShellColor](http://docs.google.com/SimpleTurtle.html#setShellColor(java.awt.Color)), [setShowInfo](http://docs.google.com/SimpleTurtle.html#setShowInfo(boolean)), [setVisible](http://docs.google.com/SimpleTurtle.html#setVisible(boolean)), [setWidth](http://docs.google.com/SimpleTurtle.html#setWidth(int)), [show](http://docs.google.com/SimpleTurtle.html#show()), [toString](http://docs.google.com/SimpleTurtle.html#toString()), [turn](http://docs.google.com/SimpleTurtle.html#turn(double)), [turnLeft](http://docs.google.com/SimpleTurtle.html#turnLeft()), [turnRight](http://docs.google.com/SimpleTurtle.html#turnRight()), [turnToFace](http://docs.google.com/SimpleTurtle.html#turnToFace(int,%20int)), [turnToFace](http://docs.google.com/SimpleTurtle.html#turnToFace(SimpleTurtle)), [updateDisplay](http://docs.google.com/SimpleTurtle.html#updateDisplay()) |

| **Methods inherited from class java.lang.**[**Object**](http://java.sun.com/javase/6/docs/api/java/lang/Object.html?is-external=true) |
| --- |
| [clone](http://java.sun.com/javase/6/docs/api/java/lang/Object.html?is-external=true#clone()), [equals](http://java.sun.com/javase/6/docs/api/java/lang/Object.html?is-external=true#equals(java.lang.Object)), [finalize](http://java.sun.com/javase/6/docs/api/java/lang/Object.html?is-external=true#finalize()), [getClass](http://java.sun.com/javase/6/docs/api/java/lang/Object.html?is-external=true#getClass()), [hashCode](http://java.sun.com/javase/6/docs/api/java/lang/Object.html?is-external=true#hashCode()), [notify](http://java.sun.com/javase/6/docs/api/java/lang/Object.html?is-external=true#notify()), [notifyAll](http://java.sun.com/javase/6/docs/api/java/lang/Object.html?is-external=true#notifyAll()), [wait](http://java.sun.com/javase/6/docs/api/java/lang/Object.html?is-external=true#wait()), [wait](http://java.sun.com/javase/6/docs/api/java/lang/Object.html?is-external=true#wait(long)), [wait](http://java.sun.com/javase/6/docs/api/java/lang/Object.html?is-external=true#wait(long,%20int)) |

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

### Deer

public **Deer**([ModelDisplay](http://docs.google.com/ModelDisplay.html) modelDisplayer)

Constructor that takes the model display (the original position will be randomally assigned

**Parameters:**modelDisplayer - thing which will display the model

### Deer

public **Deer**(int x,  
 int y,  
 [ModelDisplay](http://docs.google.com/ModelDisplay.html) modelDisplayer)

Constructor that takes the x and y and a model display to draw it on

**Parameters:**x - the starting x positiony - the starting y positionmodelDisplayer - the thing that displays the model

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

### getHead

public static [Deer](http://docs.google.com/Deer.html) **getHead**()

Method to get the head of the linked list of deer

**Returns:**the head of the linked list of deer (may be null)

### getNext

public [Deer](http://docs.google.com/Deer.html) **getNext**()

Method to get the next Deer in the linked list

**Returns:**the next deer in the linked list of deer or null

### init

public void **init**()

Method to initialize the new deer object

### act

public void **act**()

Method to act during a time step pick a random direction and move some random amount up to top speed

### die

public void **die**()

Method that handles when a deer dies

### main

public static void **main**([String](http://java.sun.com/javase/6/docs/api/java/lang/String.html?is-external=true)[] args)

main method for testing

| | [**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/DatabaseManager.html)   [**NEXT CLASS**](http://docs.google.com/DigitalPicture.html) | [**FRAMES**](http://docs.google.com/index.html?Deer.html)    [**NO FRAMES**](http://docs.google.com/Deer.html)     [**All Classes**](http://docs.google.com/allclasses-noframe.html) |
| SUMMARY: NESTED | FIELD | [CONSTR](#3znysh7) | [METHOD](#2et92p0) | DETAIL: FIELD | [CONSTR](#1t3h5sf) | [METHOD](#17dp8vu) |