

Class Sysfile

java.lang.Object

GameObject

Sysfile

```
class Sysfile
extends GameObject
```

A system file. Comes in three sizes. Increases CPU usage if junked.

Nested Class Summary

| Nested Classes | |
|--|--|
| Modifier and Type | Class and Description |
| static class | Sysfile.Size An enumeration for the three sizes of Sysfile |
| Nested classes/interfaces inherited from class GameObject | |
| GameObject.CollHandler | |

Field Summary

| Fields inherited from class GameObject |
|---|
| accel, bgg, bounds, collHandler, collRectOffset, isDead, lastKinematicsVars, position, sprite, velocity |

Constructor Summary

| Constructors | |
|---|--|
| Constructor and Description | |
| Sysfile (java.awt.Rectangle bounds, Sysfile.Size s) | Modifies the bounds to the object despawns off-screen. |

Method Summary

Methods

| Modifier and Type | Method and Description |
|-------------------|--|
| void | <code>collideWith (GameObject g)</code> All classes should override this method like so: <code>g.getCollHandler().to(this)</code> ; This code takes the CollHandler of the other object, and calls the handler appropriate for this object. |
| void | <code>onOutOfBounds ()</code> Destroys the sysfile once it leaves the boundaries of the screen. |

Methods inherited from class `GameObject`

`applyAccel, applyVelocity, calculateCollRectFromSprite, confine, confine, cycle, decelerate, decelerate, getAccel, getAreaRect, getBounds, getCollHandler, getCollRect, getCollRectOffset, getPosition, getSprite, getVelocity, kill, popKinematicsVars, setAccel, setBounds, setCollHandler, setCollRectOffset, setPosition, setSprite, setVelocity, stashKinematicsVars`

Methods inherited from class `java.lang.Object`

`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructor Detail

Sysfile

```
public Sysfile(java.awt.Rectangle bounds,
               Sysfile.Size s)
```

Modifies the bounds to the object despawns off-screen. This object's collision handlers are empty.

Parameters:

`bounds` - The boundaries of this object's creator

`s` - A size for this object

Method Detail

collideWith

```
public void collideWith(GameObject g)
```

Description copied from class: `GameObject`

All classes should override this method like so: `g.getCollHandler().to(this)`; This code takes the `CollHandler` of the other object, and calls the handler appropriate for this object. This way, handling collisions with various objects can be handled using overloading rather than e.g. object-identifying properties. The advantage is that the decision of which handler to call can be decided at compile-time. More technically, collision handlers have been implemented through the *visitor design pattern*, where implementations of `CollHandler` are the visitors. Note that `collideWith(g)` calls `g`'s handlers, not this object's.

Specified by:

`collideWith` in class `GameObject`

Parameters:

`g` - The other `GameObject`.

onOutOfBounds

```
public void onOutOfBounds()
```

Destroys the sysfile once it leaves the boundaries of the screen.

Overrides:

`onOutOfBounds` in class `GameObject`

[Package](#) [Class](#) [Use](#) [Tree](#) [Deprecated](#) [Index](#) [Help](#)

[Prev Class](#) [Next Class](#) [Frames](#) [No Frames](#) [All Classes](#)

Summary: [Nested](#) | [Field](#) | [Constr](#) | [Method](#) Detail: [Field](#) | [Constr](#) | [Method](#)