Crimsonportal

Package
crimsonportal.googlecode.com

crimsonportal.googlecode.com
Class Debug

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.Debug
```

public abstract class **Debug**
extends Object

Nested Class Summary

class	Debug.flagKey Debug.flagKey
class	Debug.flagValue Debug.flagValue

Field Summary

public static final	DEBUG_ERRORS Value: **3**
public static final	DEBUG_INFO Value: **1**
public static final	DEBUG_NONE Value: **0**
public static final	DEBUG_VERBOSE Value: **4**
public static final	DEBUG_WARNINGS Value: **2**
public static final	enabledLevel Value: **2**
protected static	flags

Constructor Summary

public	Debug()

Method Summary

static boolean	checkFlag(Debug.flagKey flag)

static Debug.flagValue	getFlagValue(Debug.flagKey flag)
static void	logEvent(String string)
static void	logMethod(String string)
static void	logWarning(String string)
static void	print(String string)
static void	setFlag(Debug.flagKey flag, Debug.flagValue value)

Methods inherited from class java.lang.Object

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

Fields

DEBUG_VERBOSE

public static final int **DEBUG_VERBOSE**

Constant value: **4**

DEBUG_ERRORS

public static final int **DEBUG_ERRORS**

Constant value: **3**

DEBUG_WARNINGS

public static final int **DEBUG_WARNINGS**

Constant value: **2**

DEBUG_INFO

public static final int **DEBUG_INFO**

Constant value: **1**

DEBUG_NONE

public static final int **DEBUG_NONE**

Constant value: **0**

enabledLevel

```
public static final int enabledLevel
```

Constant value: **2**

flags

```
protected static java.util.Map flags
```

Constructors

Debug

```
public Debug()
```

Methods

logMethod

```
public static void logMethod(String string)
```

logEvent

```
public static void logEvent(String string)
```

logWarning

```
public static void logWarning(String string)
```

print

```
public static void print(String string)
```

setFlag

```
public static void setFlag(Debug.flagKey flag,
        Debug.flagValue value)
```

getFlagValue

```
public static Debug.flagValue getFlagValue(Debug.flagKey flag)
```

(continued on next page)

(continued from last page)

checkFlag

```
public static boolean checkFlag(Debug.flagKey flag)
```

crimsonportal.googlecode.com
Class Debug.flagKey

```
java.lang.Object
   ┗-java.lang.Enum
      ┗-crimsonportal.googlecode.com.Debug.flagKey
```

All Implemented Interfaces:
Serializable, Comparable

---

public static final class **Debug.flagKey**
extends Enum

Field Summary

public static final	DISABLE_ENEMY_SPAWNING
public static final	PLAYER_MOVEMENT_VERTICAL

Method Summary

static Debug.flagKey	valueOf(String name)
static Debug.flagKey[]	values()

Methods inherited from class java.lang.Enum

clone, compareTo, equals, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class java.lang.Object

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

Methods inherited from interface java.lang.Comparable

compareTo

Fields

DISABLE_ENEMY_SPAWNING

public static final crimsonportal.googlecode.com.Debug.flagKey **DISABLE_ENEMY_SPAWNING**

(continued from last page)

PLAYER_MOVEMENT_VERTICAL

public static final crimsonportal.googlecode.com.Debug.flagKey
PLAYER_MOVEMENT_VERTICAL

Methods

values

public final static Debug.flagKey[] **values**()

---

valueOf

public static Debug.flagKey **valueOf**(String name)

crimsonportal.googlecode.com
Class Debug.flagValue

```
java.lang.Object
   |-java.lang.Enum
      |-crimsonportal.googlecode.com.Debug.flagValue
```

All Implemented Interfaces:
Serializable, Comparable

---

public static final class **Debug.flagValue**
extends Enum

Field Summary

public static final	ASCENDING
public static final	DESCENDING
public static final	FALSE
public static final	LEVEL
public static final	NOT_SET
public static final	TRUE

Method Summary

static Debug.flagValue	valueOf(String name)
static Debug.flagValue[]	values()

Methods inherited from class java.lang.Enum

clone, compareTo, equals, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class java.lang.Object

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

Methods inherited from interface java.lang.Comparable

compareTo

Fields

TRUE

```
public static final crimsonportal.googlecode.com.Debug.flagValue TRUE
```

FALSE

```
public static final crimsonportal.googlecode.com.Debug.flagValue FALSE
```

ASCENDING

```
public static final crimsonportal.googlecode.com.Debug.flagValue ASCENDING
```

DESCENDING

```
public static final crimsonportal.googlecode.com.Debug.flagValue DESCENDING
```

LEVEL

```
public static final crimsonportal.googlecode.com.Debug.flagValue LEVEL
```

NOT_SET

```
public static final crimsonportal.googlecode.com.Debug.flagValue NOT_SET
```

Methods

values

```
public final static Debug.flagValue[] values()
```

valueOf

```
public static Debug.flagValue valueOf(String name)
```

crimsonportal.googlecode.com
Class main

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.main
```

public class **main**
extends Object

Constructor Summary

public	main()

Method Summary

static void	main(String[] args)

Methods inherited from class `java.lang.Object`

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

Constructors

main

`public main()`

Methods

main

`public static void main(String[] args)`

Package

crimsonportal.googlecode.com.Controller

crimsonportal.googlecode.com.Controller
Class Controller

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.Controller.Controller
```

All Implemented Interfaces:
Observable

---

public abstract class **Controller**
extends Object
implements Observable

Constructor Summary

public	Controller()

Methods inherited from class `java.lang.Object`

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

Methods inherited from interface `crimsonportal.googlecode.com.Observer.Observable`

`addObserver, countObservers, notifyObservers, removeAllObservers, removeObserver`

Constructors

Controller

```
public Controller()
```

crimsonportal.googlecode.com.Controller
Class KeyController

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.Controller.KeyController
```

All Implemented Interfaces:
KeyPressObservable, java.awt.event.KeyListener

---

public class **KeyController**
extends Object
implements java.awt.event.KeyListener, KeyPressObservable

Constructor Summary

public	KeyController(PlayerUnit controlledPlayer, GameState gameState)

Method Summary

boolean	addObserver(Observer observer)
int	countObservers()
void	keyPressed(java.awt.event.KeyEvent e)
void	keyReleased(java.awt.event.KeyEvent e)
void	keyTyped(java.awt.event.KeyEvent e)
void	notifyObservers(java.awt.event.KeyEvent event)
void	removeAllObservers()
boolean	removeObserver(Observer observer)

Methods inherited from class `java.lang.Object`

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

Methods inherited from interface `java.awt.event.KeyListener`

`keyPressed, keyReleased, keyTyped`

Methods inherited from interface `crimsonportal.googlecode.com.Observer.Observable`

addObserver, countObservers, notifyObservers, removeAllObservers, removeObserver

Constructors

KeyController

```
public KeyController(PlayerUnit controlledPlayer,
                     GameState gameState)
```

Methods

keyPressed

```
public void keyPressed(java.awt.event.KeyEvent e)
```

keyReleased

```
public void keyReleased(java.awt.event.KeyEvent e)
```

keyTyped

```
public void keyTyped(java.awt.event.KeyEvent e)
```

notifyObservers

```
public void notifyObservers(java.awt.event.KeyEvent event)
```

addObserver

```
public boolean addObserver(Observer observer)
```

removeObserver

```
public boolean removeObserver(Observer observer)
```

removeAllObservers

```
public void removeAllObservers()
```

(continued from last page)

countObservers

```
public int countObservers()
```

crimsonportal.googlecode.com.Controller
Class MouseController

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.Controller.MouseController
```

public class **MouseController**
extends Object

Constructor Summary

public	MouseController()

Methods inherited from class `java.lang.Object`

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

Constructors

MouseController

`public` **`MouseController`**`()`

crimsonportal.googlecode.com.Controller
Class ShootListener

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.Controller.ShootListener
```

All Implemented Interfaces:
Observable, java.awt.event.MouseMotionListener, java.awt.event.MouseListener

---

public class **ShootListener**
extends Object
implements java.awt.event.MouseListener, java.awt.event.MouseMotionListener, Observable

---

Constructor Summary

public	ShootListener(Unit controlledUnit)

Method Summary

boolean	addObserver(Observer observer)
int	countObservers()
void	mouseClicked(java.awt.event.MouseEvent e)
void	mouseDragged(java.awt.event.MouseEvent e)
void	mouseEntered(java.awt.event.MouseEvent e)
void	mouseExited(java.awt.event.MouseEvent e)
void	mouseMoved(java.awt.event.MouseEvent e)
void	mousePressed(java.awt.event.MouseEvent e)
void	mouseReleased(java.awt.event.MouseEvent e)
void	notifyObservers(ShootEvent event)
void	removeAllObservers()
boolean	removeObserver(Observer observer)

Methods inherited from class java.lang.Object

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

Methods inherited from interface java.awt.event.MouseListener
mouseClicked, mouseEntered, mouseExited, mousePressed, mouseReleased

Methods inherited from interface java.awt.event.MouseMotionListener
mouseDragged, mouseMoved

Methods inherited from interface crimsonportal.googlecode.com.Observer.Observable
addObserver, countObservers, notifyObservers, removeAllObservers, removeObserver

Constructors

ShootListener

public **ShootListener**(Unit controlledUnit)

Methods

notifyObservers

public void **notifyObservers**(ShootEvent event)

removeAllObservers

public void **removeAllObservers**()

countObservers

public int **countObservers**()

addObserver

public boolean **addObserver**(Observer observer)

removeObserver

public boolean **removeObserver**(Observer observer)

(continued from last page)

mouseClicked

```
public void mouseClicked(java.awt.event.MouseEvent e)
```

mousePressed

```
public void mousePressed(java.awt.event.MouseEvent e)
```

mouseReleased

```
public void mouseReleased(java.awt.event.MouseEvent e)
```

mouseEntered

```
public void mouseEntered(java.awt.event.MouseEvent e)
```

mouseExited

```
public void mouseExited(java.awt.event.MouseEvent e)
```

mouseMoved

```
public void mouseMoved(java.awt.event.MouseEvent e)
```

mouseDragged

```
public void mouseDragged(java.awt.event.MouseEvent e)
```

crimsonportal.googlecode.com.Controller
Class TurnListener

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.Controller.TurnListener
```

All Implemented Interfaces:
Observable, java.awt.event.MouseMotionListener

---

public class **TurnListener**
extends Object
implements java.awt.event.MouseMotionListener, Observable

Constructor Summary

public	TurnListener(PlayerUnit controlledPlayer)

Method Summary

boolean	addObserver(Observer observer)
int	countObservers()
void	keyTyped(java.awt.event.KeyEvent e)
void	mouseDragged(java.awt.event.MouseEvent e)
void	mouseMoved(java.awt.event.MouseEvent e)
void	notifyObservers(PlayerTurnEvent event)
void	removeAllObservers()
boolean	removeObserver(Observer observer)

Methods inherited from class java.lang.Object

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

Methods inherited from interface java.awt.event.MouseMotionListener

mouseDragged, mouseMoved

Methods inherited from interface crimsonportal.googlecode.com.Observer.Observable

addObserver, countObservers, notifyObservers, removeAllObservers, removeObserver

Constructors

TurnListener

```
public TurnListener(PlayerUnit controlledPlayer)
```

Methods

mouseMoved

```
public void mouseMoved(java.awt.event.MouseEvent e)
```

mouseDragged

```
public void mouseDragged(java.awt.event.MouseEvent e)
```

keyTyped

```
public void keyTyped(java.awt.event.KeyEvent e)
```

notifyObservers

```
public void notifyObservers(PlayerTurnEvent event)
```

removeAllObservers

```
public void removeAllObservers()
```

countObservers

```
public int countObservers()
```

addObserver

```
public boolean addObserver(Observer observer)
```

removeObserver

```
public boolean removeObserver(Observer observer)
```

(continued from last page)

Package
crimsonportal.googlecode.com.Factories

crimsonportal.googlecode.com.Factories
Class EnemyUnitFactory

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.Factories.EnemyUnitFactory
```

public class **EnemyUnitFactory**
extends Object

Nested Class Summary

class	EnemyUnitFactory.enemyType EnemyUnitFactory.enemyType

Constructor Summary

public	EnemyUnitFactory()

Method Summary

static EnemyUnit	createEnemyUnit(EnemyUnitFactory.enemyType enemyType, Location location, GameObject target, GameState gameState)

Methods inherited from class java.lang.Object

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

Constructors

EnemyUnitFactory

```
public EnemyUnitFactory()
```

Methods

createEnemyUnit

```
public static EnemyUnit createEnemyUnit(EnemyUnitFactory.enemyType enemyType,
        Location location,
        GameObject target,
        GameState gameState)
```

crimsonportal.googlecode.com.Factories
Class EnemyUnitFactory.enemyType

```
java.lang.Object
   |
   +-java.lang.Enum
      |
      +-crimsonportal.googlecode.com.Factories.EnemyUnitFactory.enemyType
```

All Implemented Interfaces:
Serializable, Comparable

---

public static final class **EnemyUnitFactory.enemyType**
extends Enum

Field Summary

public static final	ENEMY_LARGE
public static final	ENEMY_MEDIUM
public static final	ENEMY_SMALL
public static final	ENEMY_TINY

Method Summary

static EnemyUnitFactory.enemyType	valueOf(String name)
static EnemyUnitFactory.enemyType[]	values()

Methods inherited from class java.lang.Enum

clone, compareTo, equals, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class java.lang.Object

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

Methods inherited from interface java.lang.Comparable

compareTo

Fields

(continued from last page)

ENEMY_TINY

```
public static final crimsonportal.googlecode.com.Factories.EnemyUnitFactory.enemyType
ENEMY_TINY
```

ENEMY_SMALL

```
public static final crimsonportal.googlecode.com.Factories.EnemyUnitFactory.enemyType
ENEMY_SMALL
```

ENEMY_MEDIUM

```
public static final crimsonportal.googlecode.com.Factories.EnemyUnitFactory.enemyType
ENEMY_MEDIUM
```

ENEMY_LARGE

```
public static final crimsonportal.googlecode.com.Factories.EnemyUnitFactory.enemyType
ENEMY_LARGE
```

Methods

values

```
public final static EnemyUnitFactory.enemyType[] values()
```

valueOf

```
public static EnemyUnitFactory.enemyType valueOf(String name)
```

crimsonportal.googlecode.com.Factories
Class PickupFactory

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.Factories.PickupFactory
```

public class **PickupFactory**
extends Object

Nested Class Summary

class	PickupFactory.pickupType PickupFactory.pickupType

Constructor Summary

public	PickupFactory()

Method Summary

static Pickup	createPickup(Location location, PickupFactory.pickupType pickupType, GameTime gameTime)
static Pickup	createRandomPickup(Location location, GameTime gameTime)

Methods inherited from class `java.lang.Object`

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

Constructors

PickupFactory

```
public PickupFactory()
```

Methods

createPickup

```
protected static Pickup createPickup(Location location,
        PickupFactory.pickupType pickupType,
        GameTime gameTime)
```

(continued from last page)

createRandomPickup

```
public static Pickup createRandomPickup(Location location,
        GameTime gameTime)
```

crimsonportal.googlecode.com.Factories
Class PickupFactory.pickupType

```
java.lang.Object
   |
   +-java.lang.Enum
      |
      +-crimsonportal.googlecode.com.Factories.PickupFactory.pickupType
```

All Implemented Interfaces:
Serializable, Comparable

---

public static final class **PickupFactory.pickupType**
extends Enum

Field Summary

public static final	PICKUP_HEALTH
public static final	PICKUP_NUKE
public static final	PICKUP_SHRINK
public static final	PICKUP_SPEEDBOOST

Method Summary

static PickupFactory.pickupType	valueOf(String name)
static PickupFactory.pickupType[]	values()

Methods inherited from class java.lang.Enum

clone, compareTo, equals, getDeclaringClass, hashCode, name, ordinal, toString, valueOf

Methods inherited from class java.lang.Object

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

Methods inherited from interface java.lang.Comparable

compareTo

Fields

(continued from last page)

PICKUP_SPEEDBOOST

public static final crimsonportal.googlecode.com.Factories.PickupFactory.pickupType **PICKUP_SPEEDBOOST**

PICKUP_SHRINK

public static final crimsonportal.googlecode.com.Factories.PickupFactory.pickupType **PICKUP_SHRINK**

PICKUP_HEALTH

public static final crimsonportal.googlecode.com.Factories.PickupFactory.pickupType **PICKUP_HEALTH**

PICKUP_NUKE

public static final crimsonportal.googlecode.com.Factories.PickupFactory.pickupType **PICKUP_NUKE**

Methods

values

public final static PickupFactory.pickupType[] **values**()

---

valueOf

public static PickupFactory.pickupType **valueOf**(String name)

Package

crimsonportal.googlecode.com.GameSettings

crimsonportal.googlecode.com.GameSettings
Class ObjectSizes

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.GameSettings.ObjectSizes
```

public abstract class **ObjectSizes**
extends Object

Field Summary

public static	BULLET_DAMAGE_PISTOL
public static final	BULLET_SIZE_PISTOL Value: **4.0**
public static final	BULLET_SPEED_PISTOL Value: **10.0**
public static final	PICKUP_SIZE Value: **20**
public static final	PLAYER_SIZE Value: **15**

Constructor Summary

public	ObjectSizes()

Methods inherited from class `java.lang.Object`

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

Fields

PICKUP_SIZE

`public static final int` **`PICKUP_SIZE`**

Constant value: **20**

PLAYER_SIZE

`public static final int` **`PLAYER_SIZE`**

(continued from last page)

Constant value: **15**

BULLET_SIZE_PISTOL

`public static final double` **BULLET_SIZE_PISTOL**

Constant value: **4.0**

BULLET_SPEED_PISTOL

`public static final double` **BULLET_SPEED_PISTOL**

Constant value: **10.0**

BULLET_DAMAGE_PISTOL

`public static int` **BULLET_DAMAGE_PISTOL**

Constructors

ObjectSizes

`public` **ObjectSizes**`()`

crimsonportal.googlecode.com.GameSettings
Class Timers

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.GameSettings.Timers
```

public abstract class **Timers**
extends Object

Field Summary

public static final	MOVE_BULLET_PISTOL Value: **50.0**
public static final	MOVE_ENEMIES Value: **50.0**
public static final	SPAWN_BULLET Value: **50.0**
public static final	SPAWN_NEW_ENEMY Value: **100.0**

Constructor Summary

public	Timers()

Methods inherited from class `java.lang.Object`

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

Fields

SPAWN_NEW_ENEMY

`public static final double` **`SPAWN_NEW_ENEMY`**

Constant value: **`100.0`**

MOVE_ENEMIES

`public static final double` **`MOVE_ENEMIES`**

Constant value: **`50.0`**

SPAWN_BULLET

```
public static final double SPAWN_BULLET
```

Constant value: **50.0**

MOVE_BULLET_PISTOL

```
public static final double MOVE_BULLET_PISTOL
```

Constant value: **50.0**

Constructors

Timers

```
public Timers()
```

Package
crimsonportal.googlecode.com.gui

crimsonportal.googlecode.com.gui
Class Animation

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.gui.Animation
```

Direct Known Subclasses:
NukeAnimation

---

public abstract class **Animation**
extends Object

Constructor Summary

public	Animation(Location centreLocation)

Method Summary

abstract boolean	drawOnto(java.awt.Graphics2D graphics)
int	getCentreX()
int	getCentreY()
long	getElapsedTimeMS()
double	getPercentageComplete()
abstract double	getTotalAnimationTimeMS()
double	getXScale(java.awt.Graphics2D g)
double	getYScale(java.awt.Graphics2D g)

Methods inherited from class `java.lang.Object`

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

Constructors

Animation

`public` **`Animation`**`(Location centreLocation)`

Methods

drawOnto

```
public abstract boolean drawOnto(java.awt.Graphics2D graphics)
```

getCentreX

```
protected int getCentreX()
```

getCentreY

```
protected int getCentreY()
```

getXScale

```
protected double getXScale(java.awt.Graphics2D g)
```

getYScale

```
protected double getYScale(java.awt.Graphics2D g)
```

getElapsedTimeMS

```
protected long getElapsedTimeMS()
```

getPercentageComplete

```
protected double getPercentageComplete()
```

getTotalAnimationTimeMS

```
protected abstract double getTotalAnimationTimeMS()
```

crimsonportal.googlecode.com.gui
Class GameCanvas

```
java.lang.Object
   ┗-java.awt.Component
      ┗-java.awt.Container
         ┗-javax.swing.JComponent
            ┗-javax.swing.JPanel
               ┗-crimsonportal.googlecode.com.gui.GameCanvas
```

All Implemented Interfaces:
Runnable, PlayerShootObservable, Observer, Serializable, java.awt.MenuContainer, java.awt.image.ImageObserver, Serializable, javax.accessibility.Accessible

---

public class **GameCanvas**
extends javax.swing.JPanel
implements javax.accessibility.Accessible, Serializable, java.awt.image.ImageObserver, java.awt.MenuContainer, Serializable, Observer, PlayerShootObservable, Runnable

---

Field Summary

protected	spriteProxy

Fields inherited from class javax.swing.JComponent

accessibleContext, listenerList, TOOL_TIP_TEXT_KEY, ui, UNDEFINED_CONDITION, WHEN_ANCESTOR_OF_FOCUSED_COMPONENT, WHEN_FOCUSED, WHEN_IN_FOCUSED_WINDOW

Fields inherited from class java.awt.Component

BOTTOM_ALIGNMENT, CENTER_ALIGNMENT, LEFT_ALIGNMENT, RIGHT_ALIGNMENT, TOP_ALIGNMENT

Fields inherited from interface java.awt.image.ImageObserver

ABORT, ALLBITS, ERROR, FRAMEBITS, HEIGHT, PROPERTIES, SOMEBITS, WIDTH

Constructor Summary

public	GameCanvas(GameController gameController)

Method Summary

boolean	addObserver(Observer observer)
int	countObservers()
GameController	getGameController()

void	notifyObservers(ShootEvent event)
void	paintComponent(java.awt.Graphics g)
void	removeAllObservers()
boolean	removeObserver(Observer observer)
void	run()
void	update(GameStateChangedEvent event)

Methods inherited from class javax.swing.JPanel

getAccessibleContext, getUI, getUIClassID, paramString, setUI, updateUI

Methods inherited from class javax.swing.JComponent

addAncestorListener, addNotify, addVetoableChangeListener, computeVisibleRect, contains, createToolTip, disable, enable, firePropertyChange, firePropertyChange, firePropertyChange, fireVetoableChange, getAccessibleContext, getActionForKeyStroke, getActionMap, getAlignmentX, getAlignmentY, getAncestorListeners, getAutoscrolls, getBorder, getBounds, getClientProperty, getComponentGraphics, getComponentPopupMenu, getConditionForKeyStroke, getDebugGraphicsOptions, getDefaultLocale, getFontMetrics, getGraphics, getHeight, getInheritsPopupMenu, getInputMap, getInputMap, getInputVerifier, getInsets, getInsets, getListeners, getLocation, getMaximumSize, getMinimumSize, getNextFocusableComponent, getPopupLocation, getPreferredSize, getRegisteredKeyStrokes, getRootPane, getSize, getToolTipLocation, getToolTipText, getToolTipText, getTopLevelAncestor, getTransferHandler, getUIClassID, getVerifyInputWhenFocusTarget, getVetoableChangeListeners, getVisibleRect, getWidth, getX, getY, grabFocus, isDoubleBuffered, isLightweightComponent, isManagingFocus, isOpaque, isOptimizedDrawingEnabled, isPaintingTile, isRequestFocusEnabled, isValidateRoot, paint, paintBorder, paintChildren, paintComponent, paintImmediately, paintImmediately, paramString, print, printAll, printBorder, printChildren, printComponent, processComponentKeyEvent, processKeyBinding, processKeyEvent, processMouseEvent, processMouseMotionEvent, putClientProperty, registerKeyboardAction, registerKeyboardAction, removeAncestorListener, removeNotify, removeVetoableChangeListener, repaint, repaint, requestDefaultFocus, requestFocus, requestFocus, requestFocusInWindow, requestFocusInWindow, resetKeyboardActions, reshape, revalidate, scrollRectToVisible, setActionMap, setAlignmentX, setAlignmentY, setAutoscrolls, setBackground, setBorder, setComponentPopupMenu, setDebugGraphicsOptions, setDefaultLocale, setDoubleBuffered, setEnabled, setFocusTraversalKeys, setFont, setForeground, setInheritsPopupMenu, setInputMap, setInputVerifier, setMaximumSize, setMinimumSize, setNextFocusableComponent, setOpaque, setPreferredSize, setRequestFocusEnabled, setToolTipText, setTransferHandler, setUI, setVerifyInputWhenFocusTarget, setVisible, unregisterKeyboardAction, update, updateUI

Methods inherited from class java.awt.Container

add, add, add, add, add, addContainerListener, addImpl, addNotify, addPropertyChangeListener, addPropertyChangeListener, applyComponentOrientation, areFocusTraversalKeysSet, countComponents, deliverEvent, doLayout, findComponentAt, findComponentAt, getAlignmentX, getAlignmentY, getComponent, getComponentAt, getComponentAt, getComponentCount, getComponents, getComponentZOrder, getContainerListeners, getFocusTraversalKeys, getFocusTraversalPolicy, getInsets, getLayout, getListeners, getMaximumSize, getMinimumSize, getMousePosition, getPreferredSize, insets, invalidate, isAncestorOf, isFocusCycleRoot, isFocusCycleRoot, isFocusTraversalPolicyProvider, isFocusTraversalPolicySet, layout, list, list, locate, minimumSize, paint, paintComponents, paramString, preferredSize, print, printComponents, processContainerEvent, processEvent, remove, remove, removeAll, removeContainerListener, removeNotify, setComponentZOrder, setFocusCycleRoot, setFocusTraversalKeys, setFocusTraversalPolicy, setFocusTraversalPolicyProvider, setFont, setLayout, transferFocusBackward, transferFocusDownCycle, update, validate, validateTree

Methods inherited from class `java.awt.Component`

action, add, addComponentListener, addFocusListener, addHierarchyBoundsListener, addHierarchyListener, addInputMethodListener, addKeyListener, addMouseListener, addMouseMotionListener, addMouseWheelListener, addNotify, addPropertyChangeListener, addPropertyChangeListener, applyComponentOrientation, areFocusTraversalKeysSet, bounds, checkImage, checkImage, coalesceEvents, contains, contains, createImage, createImage, createVolatileImage, createVolatileImage, deliverEvent, disable, disableEvents, dispatchEvent, doLayout, enable, enable, enableEvents, enableInputMethods, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, getAccessibleContext, getAlignmentX, getAlignmentY, getBackground, getBounds, getBounds, getColorModel, getComponentAt, getComponentAt, getComponentListeners, getComponentOrientation, getCursor, getDropTarget, getFocusCycleRootAncestor, getFocusListeners, getFocusTraversalKeys, getFocusTraversalKeysEnabled, getFont, getFontMetrics, getForeground, getGraphics, getGraphicsConfiguration, getHeight, getHierarchyBoundsListeners, getHierarchyListeners, getIgnoreRepaint, getInputContext, getInputMethodListeners, getInputMethodRequests, getKeyListeners, getListeners, getLocale, getLocation, getLocation, getLocationOnScreen, getMaximumSize, getMinimumSize, getMouseListeners, getMouseMotionListeners, getMousePosition, getMouseWheelListeners, getName, getParent, getPeer, getPreferredSize, getPropertyChangeListeners, getPropertyChangeListeners, getSize, getSize, getToolkit, getTreeLock, getWidth, getX, getY, gotFocus, handleEvent, hasFocus, hide, imageUpdate, inside, invalidate, isBackgroundSet, isCursorSet, isDisplayable, isDoubleBuffered, isEnabled, isFocusable, isFocusCycleRoot, isFocusOwner, isFocusTraversable, isFontSet, isForegroundSet, isLightweight, isMaximumSizeSet, isMinimumSizeSet, isOpaque, isPreferredSizeSet, isShowing, isValid, isVisible, keyDown, keyUp, layout, list, list, list, list, list, locate, location, lostFocus, minimumSize, mouseDown, mouseDrag, mouseEnter, mouseExit, mouseMove, mouseUp, move, nextFocus, paint, paintAll, paramString, postEvent, preferredSize, prepareImage, prepareImage, print, printAll, processComponentEvent, processEvent, processFocusEvent, processHierarchyBoundsEvent, processHierarchyEvent, processInputMethodEvent, processKeyEvent, processMouseEvent, processMouseMotionEvent, processMouseWheelEvent, remove, removeComponentListener, removeFocusListener, removeHierarchyBoundsListener, removeHierarchyListener, removeInputMethodListener, removeKeyListener, removeMouseListener, removeMouseMotionListener, removeMouseWheelListener, removeNotify, removePropertyChangeListener, removePropertyChangeListener, repaint, repaint, repaint, repaint, requestFocus, requestFocus, requestFocusInWindow, requestFocusInWindow, reshape, resize, resize, setBackground, setBounds, setBounds, setComponentOrientation, setCursor, setDropTarget, setEnabled, setFocusable, setFocusTraversalKeys, setFocusTraversalKeysEnabled, setFont, setForeground, setIgnoreRepaint, setLocale, setLocation, setLocation, setMaximumSize, setMinimumSize, setName, setPreferredSize, setSize, setSize, setVisible, show, show, size, toString, transferFocus, transferFocusBackward, transferFocusUpCycle, update, validate

Methods inherited from class `java.lang.Object`

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

Methods inherited from interface `java.awt.image.ImageObserver`

imageUpdate

Methods inherited from interface `java.awt.MenuContainer`

getFont, postEvent, remove

Methods inherited from interface javax.accessibility.Accessible
getAccessibleContext

Methods inherited from interface crimsonportal.googlecode.com.Observer.Observer
update

Methods inherited from interface crimsonportal.googlecode.com.Observer.Observable
addObserver, countObservers, notifyObservers, removeAllObservers, removeObserver

Methods inherited from interface java.lang.Runnable
run

Fields

spriteProxy

```
protected crimsonportal.googlecode.com.Proxy.SpriteProxy spriteProxy
```

Constructors

GameCanvas

```
public GameCanvas(GameController gameController)
```

Methods

paintComponent

```
public void paintComponent(java.awt.Graphics g)
```

update

```
public void update(GameStateChangedEvent event)
```

getGameController

```
protected GameController getGameController()
```

(continued from last page)

run

```
public void run()
```

---

notifyObservers

```
public void notifyObservers(ShootEvent event)
```

---

addObserver

```
public boolean addObserver(Observer observer)
```

---

removeObserver

```
public boolean removeObserver(Observer observer)
```

---

removeAllObservers

```
public void removeAllObservers()
```

---

countObservers

```
public int countObservers()
```

crimsonportal.googlecode.com.gui
Class GameFrame

```
java.lang.Object
   ┗-java.awt.Component
      ┗-java.awt.Container
         ┗-java.awt.Window
            ┗-java.awt.Frame
               ┗-javax.swing.JFrame
                  ┗-crimsonportal.googlecode.com.gui.GameFrame
```

All Implemented Interfaces:

Serializable, java.awt.MenuContainer, java.awt.image.ImageObserver, javax.accessibility.Accessible, java.awt.MenuContainer, javax.swing.RootPaneContainer, javax.accessibility.Accessible, javax.swing.WindowConstants

---

public class **GameFrame**
extends javax.swing.JFrame

Field Summary

public	bgImage
protected	canvas
protected	map

Fields inherited from class `javax.swing.JFrame`

accessibleContext, EXIT_ON_CLOSE, rootPane, rootPaneCheckingEnabled

Fields inherited from class `java.awt.Frame`

CROSSHAIR_CURSOR, DEFAULT_CURSOR, E_RESIZE_CURSOR, HAND_CURSOR, ICONIFIED, MAXIMIZED_BOTH, MAXIMIZED_HORIZ, MAXIMIZED_VERT, MOVE_CURSOR, N_RESIZE_CURSOR, NE_RESIZE_CURSOR, NORMAL, NW_RESIZE_CURSOR, S_RESIZE_CURSOR, SE_RESIZE_CURSOR, SW_RESIZE_CURSOR, TEXT_CURSOR, W_RESIZE_CURSOR, WAIT_CURSOR

Fields inherited from class `java.awt.Component`

BOTTOM_ALIGNMENT, CENTER_ALIGNMENT, LEFT_ALIGNMENT, RIGHT_ALIGNMENT, TOP_ALIGNMENT

Fields inherited from interface `java.awt.image.ImageObserver`

ABORT, ALLBITS, ERROR, FRAMEBITS, HEIGHT, PROPERTIES, SOMEBITS, WIDTH

Fields inherited from interface `javax.swing.WindowConstants`

DISPOSE_ON_CLOSE, DO_NOTHING_ON_CLOSE, EXIT_ON_CLOSE, HIDE_ON_CLOSE

Constructor Summary

public	GameFrame(GameCanvas canvas, Map map)

Method Summary

void	initBGImage()

Methods inherited from class javax.swing.JFrame

addImpl, createRootPane, frameInit, getAccessibleContext, getContentPane, getDefaultCloseOperation, getGlassPane, getJMenuBar, getLayeredPane, getRootPane, isDefaultLookAndFeelDecorated, isRootPaneCheckingEnabled, paramString, processWindowEvent, remove, setContentPane, setDefaultCloseOperation, setDefaultLookAndFeelDecorated, setGlassPane, setIconImage, setJMenuBar, setLayeredPane, setLayout, setRootPane, setRootPaneCheckingEnabled, update

Methods inherited from class java.awt.Frame

addNotify, finalize, getAccessibleContext, getCursorType, getExtendedState, getFrames, getIconImage, getMaximizedBounds, getMenuBar, getState, getTitle, isResizable, isUndecorated, paramString, remove, removeNotify, setCursor, setExtendedState, setIconImage, setMaximizedBounds, setMenuBar, setResizable, setState, setTitle, setUndecorated

Methods inherited from class java.awt.Window

addNotify, addPropertyChangeListener, addPropertyChangeListener, addWindowFocusListener, addWindowListener, addWindowStateListener, applyResourceBundle, applyResourceBundle, createBufferStrategy, createBufferStrategy, dispose, finalize, getAccessibleContext, getBufferStrategy, getFocusableWindowState, getFocusCycleRootAncestor, getFocusOwner, getFocusTraversalKeys, getGraphicsConfiguration, getInputContext, getListeners, getLocale, getMostRecentFocusOwner, getOwnedWindows, getOwner, getToolkit, getWarningString, getWindowFocusListeners, getWindowListeners, getWindowStateListeners, hide, isActive, isAlwaysOnTop, isFocusableWindow, isFocusCycleRoot, isFocused, isLocationByPlatform, isShowing, pack, postEvent, processEvent, processWindowEvent, processWindowFocusEvent, processWindowStateEvent, removeWindowFocusListener, removeWindowListener, removeWindowStateListener, setAlwaysOnTop, setBounds, setCursor, setFocusableWindowState, setFocusCycleRoot, setLocationByPlatform, setLocationRelativeTo, show, toBack, toFront

Methods inherited from class java.awt.Container

add, add, add, add, add, addContainerListener, addImpl, addNotify, addPropertyChangeListener, addPropertyChangeListener, applyComponentOrientation, areFocusTraversalKeysSet, countComponents, deliverEvent, doLayout, findComponentAt, findComponentAt, getAlignmentX, getAlignmentY, getComponent, getComponentAt, getComponentAt, getComponentCount, getComponents, getComponentZOrder, getContainerListeners, getFocusTraversalKeys, getFocusTraversalPolicy, getInsets, getLayout, getListeners, getMaximumSize, getMinimumSize, getMousePosition, getPreferredSize, insets, invalidate, isAncestorOf, isFocusCycleRoot, isFocusCycleRoot, isFocusTraversalPolicyProvider, isFocusTraversalPolicySet, layout, list, list, locate, minimumSize, paint, paintComponents, paramString, preferredSize, print, printComponents, processContainerEvent, processEvent, remove, remove, removeAll, removeContainerListener, removeNotify, setComponentZOrder, setFocusCycleRoot, setFocusTraversalKeys, setFocusTraversalPolicy, setFocusTraversalPolicyProvider, setFont, setLayout, transferFocusBackward, transferFocusDownCycle, update, validate, validateTree

Methods inherited from class `java.awt.Component`

action, add, addComponentListener, addFocusListener, addHierarchyBoundsListener, addHierarchyListener, addInputMethodListener, addKeyListener, addMouseListener, addMouseMotionListener, addMouseWheelListener, addNotify, addPropertyChangeListener, addPropertyChangeListener, applyComponentOrientation, areFocusTraversalKeysSet, bounds, checkImage, checkImage, coalesceEvents, contains, contains, createImage, createImage, createVolatileImage, createVolatileImage, deliverEvent, disable, disableEvents, dispatchEvent, doLayout, enable, enable, enableEvents, enableInputMethods, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, getAccessibleContext, getAlignmentX, getAlignmentY, getBackground, getBounds, getBounds, getColorModel, getComponentAt, getComponentAt, getComponentListeners, getComponentOrientation, getCursor, getDropTarget, getFocusCycleRootAncestor, getFocusListeners, getFocusTraversalKeys, getFocusTraversalKeysEnabled, getFont, getFontMetrics, getForeground, getGraphics, getGraphicsConfiguration, getHeight, getHierarchyBoundsListeners, getHierarchyListeners, getIgnoreRepaint, getInputContext, getInputMethodListeners, getInputMethodRequests, getKeyListeners, getListeners, getLocale, getLocation, getLocation, getLocationOnScreen, getMaximumSize, getMinimumSize, getMouseListeners, getMouseMotionListeners, getMousePosition, getMouseWheelListeners, getName, getParent, getPeer, getPreferredSize, getPropertyChangeListeners, getPropertyChangeListeners, getSize, getSize, getToolkit, getTreeLock, getWidth, getX, getY, gotFocus, handleEvent, hasFocus, hide, imageUpdate, inside, invalidate, isBackgroundSet, isCursorSet, isDisplayable, isDoubleBuffered, isEnabled, isFocusable, isFocusCycleRoot, isFocusOwner, isFocusTraversable, isFontSet, isForegroundSet, isLightweight, isMaximumSizeSet, isMinimumSizeSet, isOpaque, isPreferredSizeSet, isShowing, isValid, isVisible, keyDown, keyUp, layout, list, list, list, list, list, locate, location, lostFocus, minimumSize, mouseDown, mouseDrag, mouseEnter, mouseExit, mouseMove, mouseUp, move, nextFocus, paint, paintAll, paramString, postEvent, preferredSize, prepareImage, prepareImage, print, printAll, processComponentEvent, processEvent, processFocusEvent, processHierarchyBoundsEvent, processHierarchyEvent, processInputMethodEvent, processKeyEvent, processMouseEvent, processMouseMotionEvent, processMouseWheelEvent, remove, removeComponentListener, removeFocusListener, removeHierarchyBoundsListener, removeHierarchyListener, removeInputMethodListener, removeKeyListener, removeMouseListener, removeMouseMotionListener, removeMouseWheelListener, removeNotify, removePropertyChangeListener, removePropertyChangeListener, repaint, repaint, repaint, repaint, requestFocus, requestFocus, requestFocusInWindow, requestFocusInWindow, reshape, resize, resize, setBackground, setBounds, setBounds, setComponentOrientation, setCursor, setDropTarget, setEnabled, setFocusable, setFocusTraversalKeys, setFocusTraversalKeysEnabled, setFont, setForeground, setIgnoreRepaint, setLocale, setLocation, setLocation, setMaximumSize, setMinimumSize, setName, setPreferredSize, setSize, setSize, setVisible, show, show, size, toString, transferFocus, transferFocusBackward, transferFocusUpCycle, update, validate

Methods inherited from class `java.lang.Object`

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

Methods inherited from interface `java.awt.image.ImageObserver`

imageUpdate

Methods inherited from interface `java.awt.MenuContainer`

getFont, postEvent, remove

Methods inherited from interface javax.accessibility.Accessible
getAccessibleContext

Methods inherited from interface java.awt.MenuContainer
getFont, postEvent, remove

Methods inherited from interface javax.accessibility.Accessible
getAccessibleContext

Methods inherited from interface javax.swing.RootPaneContainer
getContentPane, getGlassPane, getLayeredPane, getRootPane, setContentPane, setGlassPane, setLayeredPane

Fields

canvas

protected crimsonportal.googlecode.com.gui.GameCanvas **canvas**

map

protected crimsonportal.googlecode.com.ObjectModel.Map **map**

bgImage

public java.awt.Image **bgImage**

Constructors

GameFrame

public **GameFrame**(GameCanvas canvas,
Map map)

Methods

initBGImage

public void **initBGImage**()

crimsonportal.googlecode.com.gui
Class HUDPanel

```
java.lang.Object
   ┗-java.awt.Component
      ┗-java.awt.Container
         ┗-javax.swing.JComponent
            ┗-javax.swing.JPanel
               ┗-crimsonportal.googlecode.com.gui.HUDPanel
```

All Implemented Interfaces:
GameStateChangedObserver, Serializable, java.awt.MenuContainer, java.awt.image.ImageObserver, Serializable, javax.accessibility.Accessible

---

public class **HUDPanel**
extends javax.swing.JPanel
implements javax.accessibility.Accessible, Serializable, java.awt.image.ImageObserver, java.awt.MenuContainer, Serializable, GameStateChangedObserver

---

Fields inherited from class `javax.swing.JComponent`
`accessibleContext`, `listenerList`, `TOOL_TIP_TEXT_KEY`, `ui`, `UNDEFINED_CONDITION`, `WHEN_ANCESTOR_OF_FOCUSED_COMPONENT`, `WHEN_FOCUSED`, `WHEN_IN_FOCUSED_WINDOW`

Fields inherited from class `java.awt.Component`
`BOTTOM_ALIGNMENT`, `CENTER_ALIGNMENT`, `LEFT_ALIGNMENT`, `RIGHT_ALIGNMENT`, `TOP_ALIGNMENT`

Fields inherited from interface `java.awt.image.ImageObserver`
`ABORT`, `ALLBITS`, `ERROR`, `FRAMEBITS`, `HEIGHT`, `PROPERTIES`, `SOMEBITS`, `WIDTH`

Constructor Summary

public	HUDPanel(GameController gameController, int width)

Method Summary

void	update(GameStateChangedEvent event)

Methods inherited from class `javax.swing.JPanel`
`getAccessibleContext`, `getUI`, `getUIClassID`, `paramString`, `setUI`, `updateUI`

Methods inherited from class `javax.swing.JComponent`

addAncestorListener, addNotify, addVetoableChangeListener, computeVisibleRect, contains, createToolTip, disable, enable, firePropertyChange, firePropertyChange, firePropertyChange, fireVetoableChange, getAccessibleContext, getActionForKeyStroke, getActionMap, getAlignmentX, getAlignmentY, getAncestorListeners, getAutoscrolls, getBorder, getBounds, getClientProperty, getComponentGraphics, getComponentPopupMenu, getConditionForKeyStroke, getDebugGraphicsOptions, getDefaultLocale, getFontMetrics, getGraphics, getHeight, getInheritsPopupMenu, getInputMap, getInputMap, getInputVerifier, getInsets, getInsets, getListeners, getLocation, getMaximumSize, getMinimumSize, getNextFocusableComponent, getPopupLocation, getPreferredSize, getRegisteredKeyStrokes, getRootPane, getSize, getToolTipLocation, getToolTipText, getToolTipText, getTopLevelAncestor, getTransferHandler, getUIClassID, getVerifyInputWhenFocusTarget, getVetoableChangeListeners, getVisibleRect, getWidth, getX, getY, grabFocus, isDoubleBuffered, isLightweightComponent, isManagingFocus, isOpaque, isOptimizedDrawingEnabled, isPaintingTile, isRequestFocusEnabled, isValidateRoot, paint, paintBorder, paintChildren, paintComponent, paintImmediately, paintImmediately, paramString, print, printAll, printBorder, printChildren, printComponent, processComponentKeyEvent, processKeyBinding, processKeyEvent, processMouseEvent, processMouseMotionEvent, putClientProperty, registerKeyboardAction, registerKeyboardAction, removeAncestorListener, removeNotify, removeVetoableChangeListener, repaint, repaint, requestDefaultFocus, requestFocus, requestFocus, requestFocusInWindow, requestFocusInWindow, resetKeyboardActions, reshape, revalidate, scrollRectToVisible, setActionMap, setAlignmentX, setAlignmentY, setAutoscrolls, setBackground, setBorder, setComponentPopupMenu, setDebugGraphicsOptions, setDefaultLocale, setDoubleBuffered, setEnabled, setFocusTraversalKeys, setFont, setForeground, setInheritsPopupMenu, setInputMap, setInputVerifier, setMaximumSize, setMinimumSize, setNextFocusableComponent, setOpaque, setPreferredSize, setRequestFocusEnabled, setToolTipText, setTransferHandler, setUI, setVerifyInputWhenFocusTarget, setVisible, unregisterKeyboardAction, update, updateUI

Methods inherited from class `java.awt.Container`

add, add, add, add, add, addContainerListener, addImpl, addNotify, addPropertyChangeListener, addPropertyChangeListener, applyComponentOrientation, areFocusTraversalKeysSet, countComponents, deliverEvent, doLayout, findComponentAt, findComponentAt, getAlignmentX, getAlignmentY, getComponent, getComponentAt, getComponentAt, getComponentCount, getComponents, getComponentZOrder, getContainerListeners, getFocusTraversalKeys, getFocusTraversalPolicy, getInsets, getLayout, getListeners, getMaximumSize, getMinimumSize, getMousePosition, getPreferredSize, insets, invalidate, isAncestorOf, isFocusCycleRoot, isFocusCycleRoot, isFocusTraversalPolicyProvider, isFocusTraversalPolicySet, layout, list, list, locate, minimumSize, paint, paintComponents, paramString, preferredSize, print, printComponents, processContainerEvent, processEvent, remove, remove, removeAll, removeContainerListener, removeNotify, setComponentZOrder, setFocusCycleRoot, setFocusTraversalKeys, setFocusTraversalPolicy, setFocusTraversalPolicyProvider, setFont, setLayout, transferFocusBackward, transferFocusDownCycle, update, validate, validateTree

Methods inherited from class `java.awt.Component`

action, add, addComponentListener, addFocusListener, addHierarchyBoundsListener, addHierarchyListener, addInputMethodListener, addKeyListener, addMouseListener, addMouseMotionListener, addMouseWheelListener, addNotify, addPropertyChangeListener, addPropertyChangeListener, applyComponentOrientation, areFocusTraversalKeysSet, bounds, checkImage, checkImage, coalesceEvents, contains, contains, createImage, createImage, createVolatileImage, createVolatileImage, deliverEvent, disable, disableEvents, dispatchEvent, doLayout, enable, enable, enableEvents, enableInputMethods, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, getAccessibleContext, getAlignmentX, getAlignmentY, getBackground, getBounds, getBounds, getColorModel, getComponentAt, getComponentAt, getComponentListeners, getComponentOrientation, getCursor, getDropTarget, getFocusCycleRootAncestor, getFocusListeners, getFocusTraversalKeys, getFocusTraversalKeysEnabled, getFont, getFontMetrics, getForeground, getGraphics, getGraphicsConfiguration, getHeight, getHierarchyBoundsListeners, getHierarchyListeners, getIgnoreRepaint, getInputContext, getInputMethodListeners, getInputMethodRequests, getKeyListeners, getListeners, getLocale, getLocation, getLocation, getLocationOnScreen, getMaximumSize, getMinimumSize, getMouseListeners, getMouseMotionListeners, getMousePosition, getMouseWheelListeners, getName, getParent, getPeer, getPreferredSize, getPropertyChangeListeners, getPropertyChangeListeners, getSize, getSize, getToolkit, getTreeLock, getWidth, getX, getY, gotFocus, handleEvent, hasFocus, hide, imageUpdate, inside, invalidate, isBackgroundSet, isCursorSet, isDisplayable, isDoubleBuffered, isEnabled, isFocusable, isFocusCycleRoot, isFocusOwner, isFocusTraversable, isFontSet, isForegroundSet, isLightweight, isMaximumSizeSet, isMinimumSizeSet, isOpaque, isPreferredSizeSet, isShowing, isValid, isVisible, keyDown, keyUp, layout, list, list, list, list, list, locate, location, lostFocus, minimumSize, mouseDown, mouseDrag, mouseEnter, mouseExit, mouseMove, mouseUp, move, nextFocus, paint, paintAll, paramString, postEvent, preferredSize, prepareImage, prepareImage, print, printAll, processComponentEvent, processEvent, processFocusEvent, processHierarchyBoundsEvent, processHierarchyEvent, processInputMethodEvent, processKeyEvent, processMouseEvent, processMouseMotionEvent, processMouseWheelEvent, remove, removeComponentListener, removeFocusListener, removeHierarchyBoundsListener, removeHierarchyListener, removeInputMethodListener, removeKeyListener, removeMouseListener, removeMouseMotionListener, removeMouseWheelListener, removeNotify, removePropertyChangeListener, removePropertyChangeListener, repaint, repaint, repaint, repaint, requestFocus, requestFocus, requestFocusInWindow, requestFocusInWindow, reshape, resize, resize, setBackground, setBounds, setBounds, setComponentOrientation, setCursor, setDropTarget, setEnabled, setFocusable, setFocusTraversalKeys, setFocusTraversalKeysEnabled, setFont, setForeground, setIgnoreRepaint, setLocale, setLocation, setLocation, setMaximumSize, setMinimumSize, setName, setPreferredSize, setSize, setSize, setVisible, show, show, size, toString, transferFocus, transferFocusBackward, transferFocusUpCycle, update, validate

Methods inherited from class `java.lang.Object`

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

Methods inherited from interface `java.awt.image.ImageObserver`

imageUpdate

Methods inherited from interface `java.awt.MenuContainer`

getFont, postEvent, remove

Methods inherited from interface javax.accessibility.Accessible
getAccessibleContext

Methods inherited from interface crimsonportal.googlecode.com.Observer.GameState.GameStateChangedObserver
update

Methods inherited from interface crimsonportal.googlecode.com.Observer.Observer
update

Constructors

HUDPanel

```
public HUDPanel(GameController gameController,
                int width)
```

Methods

update

```
public void update(GameStateChangedEvent event)
```

Package

crimsonportal.googlecode.com.gui.Animations

crimsonportal.googlecode.com.gui.Animations
Class NukeAnimation

```
java.lang.Object
   |-crimsonportal.googlecode.com.gui.Animation
      |-crimsonportal.googlecode.com.gui.Animations.NukeAnimation
```

public class **NukeAnimation**
extends Animation

Field Summary

protected	maxRadius

Constructor Summary

public	NukeAnimation(Location centreLocation, int maxRadius)

Method Summary

boolean	drawOnto(java.awt.Graphics2D g)
double	getTotalAnimationTimeMS()

Methods inherited from class crimsonportal.googlecode.com.gui.Animation

drawOnto, getCentreX, getCentreY, getElapsedTimeMS, getPercentageComplete, getTotalAnimationTimeMS, getXScale, getYScale

Methods inherited from class java.lang.Object

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

Fields

maxRadius

protected int **maxRadius**

Constructors

(continued from last page)

NukeAnimation

```
public NukeAnimation(Location centreLocation,
                     int maxRadius)
```

Methods

drawOnto

```
public boolean drawOnto(java.awt.Graphics2D g)
```

---

getTotalAnimationTimeMS

```
protected double getTotalAnimationTimeMS()
```

Package

crimsonportal.googlecode.com.ObjectModel

crimsonportal.googlecode.com.ObjectModel
Class Bullet

```
java.lang.Object
   |-crimsonportal.googlecode.com.ObjectModel.GameObject
      |-crimsonportal.googlecode.com.ObjectModel.Bullet
```

public class **Bullet**
extends GameObject

Represents a bullet (that is, any bullet fired by any player which moves across the game field)

Field Summary

protected	attackDamage The base damage which will be dealt to an object which this bullet attacks
protected	moveSpeed The logical movement speed of this bullet.
protected	shooter The GameObject (typically, but not necessarily, a Unit) which fired this bullet
protected	strategy The Strategy used by this bullet, used to define features such as the way in which the bullet will move

Fields inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject

location, rotation, size

Constructor Summary

public	Bullet(double size, GameObject shooter, Location location, Strategy strategy, double moveSpeed, int attackDamage) Constructs a new bullet with the given minimum parameters

Method Summary

Bullet	clone() Creates and returns a clone (that is, an exact copy) of the current Bullet, such that the two bullets (this and the clone) will have the identical properties but will be independant of one another.
int	getAttackDamage() Returns the base damage inflicted by this bullet on GameObjects it collides with
double	getMoveSpeed() Returns the logical speed with which this bullet will move.
double	getRotation() Gets the direction which the bullet is currently facing, in radians.

String	getSpriteFilename() Specifies the name of the filename which represents the graphical representation of this game object.
Strategy	getStrategy() Returns the Strategy which will be used to determine the movement path of this bullet
void	move() Moves the current bullet in a manner dictated by its current strategy.
void	setAttackDamage(int attackDamage) Sets the base damage which will be inflicted by this bullet on GameObjects it collides with
void	setMoveSpeed(double moveSpeed) Sets the logical speed with which this bullet will move.
void	setStrategy(Strategy strategy) Sets the Strategy which should be used to determine the way in which this bullet moves

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject

getCentreOfObject, getRotation, getSize, getSpriteFilename, setCentreOfObject, setRotation, setSize, testOverlapsWith

Methods inherited from class java.lang.Object

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

Fields

strategy

protected crimsonportal.googlecode.com.ObjectModel.Strategy **strategy**

The Strategy used by this bullet, used to define features such as the way in which the bullet will move

moveSpeed

protected double **moveSpeed**

The logical movement speed of this bullet. Note that this value is independant of any pixel-related speed which may be rendered by a GUI

attackDamage

protected int **attackDamage**

The base damage which will be dealt to an object which this bullet attacks

shooter

protected crimsonportal.googlecode.com.ObjectModel.GameObject **shooter**

The GameObject (typically, but not necessarily, a Unit) which fired this bullet

(continued from last page)

Constructors

Bullet

```
public Bullet(double size,
              GameObject shooter,
              Location location,
              Strategy strategy,
              double moveSpeed,
              int attackDamage)
```

Constructs a new bullet with the given minimum parameters

Parameters:

`size` - the size (radius) of the bullet. Note that this is the logical size, and is not necessarily the same size the bullet may be rendered by any particular GUI element
`shooter` - the GameObject (typically, but not necessarily, a Unit) which fired the bullet
`location` - the starting location of the bullet, usually (but not necessarily) the same as the location of the shooter
`strategy` - the `Strategy` which defines the way in which the bullet will move
`moveSpeed` - the speed with which the bullet will travel. Note that this is the logical size, and is not necessarily the same pixel value used when any particular GUI renders the bullet
`attackDamage` - the base damage this bullet will inflict on a GameObject if it collides with it

Methods

getMoveSpeed

```
public double getMoveSpeed()
```

Returns the logical speed with which this bullet will move. Note that this is independant of the pixel distance the bullet may travel when rendered by any particular GUI

Returns:

the movement speed of this bullet

---

setMoveSpeed

```
protected void setMoveSpeed(double moveSpeed)
```

Sets the logical speed with which this bullet will move. Note that this is independant of the pixel distance the bullet may travel when rendered by any particular GUI, and is relative in scale to an appropriate logical container (such as a `Map`)

Parameters:

`moveSpeed` - the new logical movement speed of this bullet

---

getAttackDamage

```
public int getAttackDamage()
```

Returns the base damage inflicted by this bullet on GameObjects it collides with

Returns:

this bullet's base attack damage

See Also:

`attack(EnemyUnit)`

(continued from last page)

setAttackDamage

`protected void setAttackDamage(int attackDamage)`

Sets the base damage which will be inflicted by this bullet on GameObjects it collides with

Parameters:
`attackDamage` - the new base attack damage for this bullet

See Also:
`attack(EnemyUnit)`

---

getStrategy

`public Strategy getStrategy()`

Returns the `Strategy` which will be used to determine the movement path of this bullet

Returns:
a Strategy which dictates the way in which this bullet will move

See Also:
`move()`

---

setStrategy

`protected void setStrategy(Strategy strategy)`

Sets the Strategy which should be used to determine the way in which this bullet moves

Parameters:
`strategy` - a Strategy which will determine the way in which this bullet moves

See Also:
`move()`

---

getSpriteFilename

`public String getSpriteFilename()`

Specifies the name of the filename which represents the graphical representation of this game object. For instances of the Bullet class, this returns the filename representing the image used by GUIs to render a bullet (that is, the filename of the image which looks like a bullet)

Note that the specifics of this rendering process are not defined here, but are deferred to the presentation-related classes.

Returns:
a string representing the filename which is used by GUIs to determine the sprite for Bullets

---

move

`public void move()`

Moves the current bullet in a manner dictated by its current strategy. This is currently implemented to mean moving in a straight line towards the target specified by the strategy

See Also:
`Strategy.getTarget()`

getRotation

`public double` **`getRotation`**`()`

Gets the direction which the bullet is currently facing, in radians. Overrides the `GameObject.getRotation()` method to keep each bullet rotated towards its current target (as defined in its Strategy)

For more details on the details of the possible values returned, see the javadoc comments for `GameObject.rotation`

Returns:
the angle which this bullet is facing, in radians

See Also:
`Strategy.getTarget()`

clone

`public` `Bullet` **`clone`**`()`

Creates and returns a clone (that is, an exact copy) of the current Bullet, such that the two bullets (`this` and the clone) will have the identical properties but will be independant of one another.

Returns:
an identical copy of the current bullet, which can be modified in any way without affecting the current bullet

crimsonportal.googlecode.com.ObjectModel
Class EnemyUnit

```
java.lang.Object
   ┗-crimsonportal.googlecode.com.ObjectModel.GameObject
      ┗-crimsonportal.googlecode.com.ObjectModel.Unit
         ┗-crimsonportal.googlecode.com.ObjectModel.EnemyUnit
```

public abstract class **EnemyUnit**
extends Unit

Field Summary

protected	attackDamage
protected	attackSpeed

Fields inherited from class crimsonportal.googlecode.com.ObjectModel.Unit

gameState, health, moveSpeed, pickups, strategy

Fields inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject

location, rotation, size

Constructor Summary

protected	EnemyUnit(double size, double startingHealth, int attackDamage, double attackSpeed, int moveSpeed, Location location, GameObject target, GameState gameState)

Method Summary

void	attack(PlayerUnit player)
abstract EnemyUnit	clone()
int	getAttackDamage()
double	getAttackSpeed()
double	getRotation()
abstract String	getSpriteFilename()
void	move()

void	setAttackDamage(int attackDamage)
void	setAttackSpeed(double attackSpeed)

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.Unit

addPickup, getHealth, getMoveSpeed, getStrategy, moveTo, setHealth, setMoveSpeed, setStrategy

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject

getCentreOfObject, getRotation, getSize, getSpriteFilename, setCentreOfObject, setRotation, setSize, testOverlapsWith

Methods inherited from class java.lang.Object

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

Fields

attackDamage

protected int **attackDamage**

attackSpeed

protected double **attackSpeed**

Constructors

EnemyUnit

```
protected EnemyUnit(double size,
                    double startingHealth,
                    int attackDamage,
                    double attackSpeed,
                    int moveSpeed,
                    Location location,
                    GameObject target,
                    GameState gameState)
```

Methods

getAttackDamage

public int **getAttackDamage**()

setAttackDamage

```
protected void setAttackDamage(int attackDamage)
```

getAttackSpeed

```
public double getAttackSpeed()
```

setAttackSpeed

```
protected void setAttackSpeed(double attackSpeed)
```

attack

```
public void attack(PlayerUnit player)
```

clone

```
public abstract EnemyUnit clone()
```

move

```
protected void move()
```

getRotation

```
public double getRotation()
```

Returns the rotation of this game object (that is, the angle that this game object is facing).

getSpriteFilename

```
public abstract String getSpriteFilename()
```

Abstract method to be implemented by inheriting classes and which specifies the name of the filename which represents the graphical representation of this game object. For example, if a class called Box is implemented which inherits from GameObject, and a Box is graphically represented by a file named "Box.jpg" then this method might be implemented in the Box class to return the string "Box.jpg". This method allows instances of Game Objects, which are logical objects, to be represented in the presentation layer by having graphical classes query the Game Object's getSpriteFilename method and rendering that file. Note that the specifics of this rendering process are not defined here, but are deferred to the presentation -related classes.

crimsonportal.googlecode.com.ObjectModel
Class GameController

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.ObjectModel.GameController
```

All Implemented Interfaces:
Runnable, Observer

---

public class **GameController**
extends Object
implements Observer, Runnable

Field Summary

protected	gameCanvas
protected	gameState

Constructor Summary

public	GameController()

Method Summary

boolean	addObserver(Observer observer)
int	countObservers()
GameCanvas	getGameCanvas()
GameState	getGameState()
void	notifyObservers(GameStateChangedEvent event)
void	removeAllObservers()
boolean	removeObserver(Observer observer)
void	run()
void	spawnEnemy()
void	update(GameStateChangedEvent event)

Methods inherited from class java.lang.Object

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

Methods inherited from interface crimsonportal.googlecode.com.Observer.Observer
update

Methods inherited from interface java.lang.Runnable
run

Fields

gameState

protected crimsonportal.googlecode.com.ObjectModel.GameState **gameState**

gameCanvas

protected crimsonportal.googlecode.com.gui.GameCanvas **gameCanvas**

Constructors

GameController

public **GameController**()

Methods

getGameState

public GameState **getGameState**()

getGameCanvas

public GameCanvas **getGameCanvas**()

spawnEnemy

public void **spawnEnemy**()

(continued from last page)

run

```
public void run()
```

removeAllObservers

```
public void removeAllObservers()
```

notifyObservers

```
public void notifyObservers(GameStateChangedEvent event)
```

addObserver

```
public boolean addObserver(Observer observer)
```

removeObserver

```
public boolean removeObserver(Observer observer)
```

countObservers

```
public int countObservers()
```

update

```
public void update(GameStateChangedEvent event)
```

crimsonportal.googlecode.com.ObjectModel
Class GameObject

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.ObjectModel.GameObject
```

Direct Known Subclasses:
Bullet, LocationObject, Pickup, Unit

---

public abstract class **GameObject**
extends Object

An abstract class which all game objects must extend, which provides storage for core properties as well as functionality common to all objects within the game scope.

Field Summary

protected	location The location of this game object, represented as the location at which the centre of the game object is logically positioned.
protected	rotation The rotation of this object (that is, the direction this game object is facing) in radians.
protected	size The size (diameter) of this game object

Constructor Summary

public	GameObject(double size, Location location) Constructor which creates a game object with given minimum properties.

Method Summary

Location	getCentreOfObject() Returns the current location of the centre of the game object
double	getRotation() Returns the rotation of this game object (that is, the angle that this game object is facing).
double	getSize() Returns the size (diameter) of the game object.
abstract String	getSpriteFilename() Abstract method to be implemented by inheriting classes and which specifies the name of the filename which represents the graphical representation of this game object.
void	setCentreOfObject(Location location) Sets the location of this game object.
void	setRotation(double rotation) Rotates this game object to face a given direction

void	setSize(double size) Sets the size (diameter) of the game object.
boolean	testOverlapsWith(GameObject obj) Tests whether this game object and another game object are so close that they are overlapping (that is, whether they are touching at all).

Methods inherited from class `java.lang.Object`

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

Fields

location

`protected crimsonportal.googlecode.com.ObjectModel.Location` **`location`**

The location of this game object, represented as the location at which the centre of the game object is logically positioned.

See Also:
- `getCentreOfObject()`
- `setCentreOfObject(Location)`

size

`protected double` **`size`**

The size (diameter) of this game object

See Also:
- `getSize()`

rotation

`protected double` **`rotation`**

The rotation of this object (that is, the direction this game object is facing) in radians. Note that a value of 0 indicates the object is facing East, with increasing values indicating a turn clockwise around the Z axis, such that:

Rotation (radians)	Degree equivalent	Direction faced
0	0°	Due East
PI / 2	90°	Due South
PI	180°	Due West
3 * PI / 2	270°	Due North

See Also:
- `getRotation()`
- `setRotation(double)`

Constructors

(continued from last page)

GameObject

```
public GameObject(double size,
                  Location location)
```

Constructor which creates a game object with given minimum properties.

Parameters:

`size` - the size (diameter) of the game object. Note that this is the logical size, and is not necessarily the physical size which the game object may be rendered by any particular GUI element.
`location` - the location of the game object (given as a Location object which represents the centre of the game object)

Methods

getCentreOfObject

```
public Location getCentreOfObject()
```

Returns the current location of the centre of the game object

Returns:

the position of the centre of the game object

---

getSize

```
public double getSize()
```

Returns the size (diameter) of the game object. Note that this is the logical size, and is not necessarily the physical size which the game object may be rendered by any particular GUI element.

Returns:

the diameter of the game object

See Also:

`setSize(double)`

---

setSize

```
protected void setSize(double size)
```

Sets the size (diameter) of the game object. Note that this is the logical size, and is not necessarily the physical size which the game object may be rendered by any particular GUI element.

Parameters:

`size` - the new diameter of the game object

See Also:

`getSize()`

---

getRotation

```
public double getRotation()
```

Returns the rotation of this game object (that is, the angle that this game object is facing).

Returns:

the rotation of the game object, in radians.

(continued from last page)

See Also:
details of the rotation value returned

setRotation

```
protected void setRotation(double rotation)
```

Rotates this game object to face a given direction

Parameters:
`rotation` - the angle to face, in radians.

See Also:
details of the rotation parameter

setCentreOfObject

```
protected void setCentreOfObject(Location location)
```

Sets the location of this game object.

Parameters:
`location` - the location representing the centre of the game object

testOverlapsWith

```
public boolean testOverlapsWith(GameObject obj)
```

Tests whether this game object and another game object are so close that they are overlapping (that is, whether they are touching at all). Note that calling `a.testOverlapsWith(b)` and `b.testOverlapsWith(a)` must return the same values in all situations.

Parameters:
`obj` - the game object which should be checked for an overlap with the current game objects

Returns:
true if `this` and obj are overlapping, and false if they are not

getSpriteFilename

```
public abstract String getSpriteFilename()
```

Abstract method to be implemented by inheriting classes and which specifies the name of the filename which represents the graphical representation of this game object. For example, if a class called Box is implemented which inherits from GameObject, and a Box is graphically represented by a file named "Box.jpg" then this method might be implemented in the Box class to return the string "Box.jpg". This method allows instances of Game Objects, which are logical objects, to be represented in the presentation layer by having graphical classes query the Game Object's getSpriteFilename method and rendering that file. Note that the specifics of this rendering process are not defined here, but are deferred to the presentation -related classes.

Returns:
a string representing the filename which is used to find the sprite for the overriding type of Game Object.

crimsonportal.googlecode.com.ObjectModel
Class GameState

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.ObjectModel.GameState
```

All Implemented Interfaces:
GameStateChangedObservable, PlayerMoveObserver

---

public class **GameState**
extends Object
implements PlayerMoveObserver, GameStateChangedObservable

Represents the culmination of all aspects of the current state of a game.

Field Summary

protected	controller The GameController which controlls this GameState
public static final	landscapeName The name of the landscape to be used (which is independant of the name used to load the Map) TODO: Replace this with a better option than a static constant, which ties together better with the name used for loading the Map Value: **terrain_peak**
protected	terrain The model of the physical 3D terrain over which GameObjects in this GameState are positioned

Constructor Summary

public	GameState(GameController gameController) Creates a new GameState.

Method Summary

void	addAnimation(Animation animation)
boolean	addObserver(Observer observer)
int	countObservers()
Iterator	getAnimations()
Iterator	getBullets() Retrieves an Iterator which allows access to details of all bullets which are active in this GameState.
Iterator	getEnemies() Retrieves an Iterator which allows access to details of all the enemy units in this GameState.

Iterator	getEnemiesNear(Location location, double rangeRadius) Returns an iterator which will iterate over all the EnemyUnits within a given distance of a given location
GameController	getGameController()
Iterator	getGameObjects() Retrieves an Iterator which allows access to details of all Game Objects in this GameState.
GameTime	getGameTime() Gets details about the current game time in this game state.
Map	getMap() Returns the map which is being used for the current GameState, which includes details about the logical 2D environment over which GameObjects in this GameState are positioned
int	getNumBullets() Returns the number of bullets which are active in this GameState.
int	getNumEnemies() Returns the number of enemy units active in the current GameState.
int	getNumPlayers() Returns the number of players active in the current GameState.
Iterator	getPickups() Retrieves an Iterator which allows access to details of all pickups which are active in this GameState.
Iterator	getPlayers() Retrieves an Iterator which allows access to details of the players in this GameState.
Terrain	getTerrain() Returns the terrain which is being used for the current GameState, which includes details about the physical 3D terrain over which GameObjects in this GameState are positioned
boolean	isSpawningBullets() Determines whether or not there is an active unit which is busy firing, and by implication whether or not another bullet should be fired
void	killEnemy(EnemyUnit enemy) Removes an enemy unit from the current GameState.
void	killUnit(Unit unit)
void	loadTerrain(String terrainFilename) Loads a terrain model (that is, the 3D model of the terrain being used in this game) from a given external file.
void	notifyObservers(GameStateChangedEvent event)
void	removeAllObservers()
boolean	removeObserver(Observer observer)
void	spawnBullet()

void	spawnEnemy(EnemyUnit enemyUnit) Spawns (creates) an enemy unit in the current GameState.
void	spawnPickup() Spawns (creates) a pickup, and adds it to the GameState.
void	spawnPlayer(Location location) Spawns (creates) a PlayerUnit at a given location, and adds it to the GameState.
void	update(PlayerMoveEvent event)
void	update(ShootEvent event)

Methods inherited from class `java.lang.Object`

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

Methods inherited from interface `crimsonportal.googlecode.com.Observer.Player.Move.PlayerMoveObserver`

update

Methods inherited from interface `crimsonportal.googlecode.com.Observer.Observer`

update

Methods inherited from interface `crimsonportal.googlecode.com.Observer.Observable`

addObserver, countObservers, notifyObservers, removeAllObservers, removeObserver

Fields

terrain

`protected crimsonportal.googlecode.com.terrain.Terrain` **`terrain`**

The model of the physical 3D terrain over which GameObjects in this GameState are positioned

See Also:
getTerrain()

landscapeName

`public static final java.lang.String` **`landscapeName`**

The name of the landscape to be used (which is independant of the name used to load the Map) TODO: Replace this with a better option than a static constant, which ties together better with the name used for loading the Map
Constant value: **`terrain_peak`**

controller

`protected crimsonportal.googlecode.com.ObjectModel.GameController` **`controller`**

The GameController which controlls this GameState

Constructors

GameState

```
public GameState(GameController gameController)
```

Creates a new GameState. This game state will have all the relevant GameObjects collections (players, enemies, pickups, etc) as well as helper class instances (such as a map, a game time and observers) initialised, but it will not have its terrain loaded and the terrain must be loaded separately by a subsequent call to `loadTerrain(String)`. This is done to maintain best-practices of not allowing exceptions to be thrown from constructors, which could conceivably happen while loading a terrain from an external file.

Methods

loadTerrain

```
public void loadTerrain(String terrainFilename)
  throws InvalidTerrainException
```

Loads a terrain model (that is, the 3D model of the terrain being used in this game) from a given external file.

Parameters:

`terrainFilename` - the absolute or relative name of the file to be used to load the terrain. This method simply wraps a call to `Terrain.loadTerrain(String)` and so the terrain filename provided must conform to the requirements dictated by that method.

Throws:

`crimsonportal.googlecode.com.terrain.InvalidTerrainException` - if the terrain file indicated by `terrainFilename` is invalid, could not be read, or does not contain a valid terrain.

See Also:

`Terrain.loadTerrain(String)`

---

getTerrain

```
public Terrain getTerrain()
```

Returns the terrain which is being used for the current GameState, which includes details about the physical 3D terrain over which GameObjects in this GameState are positioned

Returns:

the terrain which was created for this GameState during initialization and which models the 3D terrain of the environment of this GameState

See Also:

`loadTerrain(String)`

---

getPlayers

```
public Iterator getPlayers()
```

Retrieves an Iterator which allows access to details of the players in this GameState. This includes all friendly and opposition human-controlled units

Returns:

a templatized Iterator which will traverse through each of the players in the current GameState

---

(continued from last page)

getEnemiesNear

```
public Iterator getEnemiesNear(Location location,
         double rangeRadius)
```

Returns an iterator which will iterate over all the EnemyUnits within a given distance of a given location

Parameters:

`location` - the location which is being used in comparison against the enemy units' locations
`rangeRadius` - the maximum distance from `location` which returned enemies may be

Returns:

an iterator over the collection of EnemyUnits which are within a distance of `rangeRadius` of `location`

See Also:

`Location.getDistanceFrom(GameObject)`

---

getNumPlayers

```
public int getNumPlayers()
```

Returns the number of players active in the current GameState. This includes both friendly and opposition human-controlled players.

Returns:

The size of the internal players Collection, which represents all the players in the current GameState

See Also:

`getPlayers()`

---

getEnemies

```
public Iterator getEnemies()
```

Retrieves an Iterator which allows access to details of all the enemy units in this GameState. Note that this iterator will not include enemy players, but will only include computer-controlled enemy units.

Returns:

a templatized Iterator which will traverse through each of the enemy units in the current GameState

See Also:

`getNumEnemies()`
`spawnEnemy(EnemyUnit)`

---

getNumEnemies

```
public int getNumEnemies()
```

Returns the number of enemy units active in the current GameState. Note that this does not refer to all enemies (including enemy players) of the current player, but rather to all computer-controlled enemy units.

Returns:

the number of computer-controlled enemy units in the current GameState

See Also:

`getEnemies()`

(continued from last page)

getBullets

`public Iterator getBullets()`

Retrieves an Iterator which allows access to details of all bullets which are active in this GameState. This refers to all bullets (including those which are controlled by the all players, as well as computer-controlled enemy units)

Returns:
a templatized Iterator which will traverse through each of the bullets in the current GameState

See Also:
`getNumBullets()`

---

getNumBullets

`public int getNumBullets()`

Returns the number of bullets which are active in this GameState. This refers to all bullets (including those which are controlled by the all players, as well as computer-controlled enemy units)

Returns:
the number of bullets in the current GameState

See Also:
`getBullets()`

---

spawnEnemy

`protected void spawnEnemy(EnemyUnit enemyUnit)`

Spawns (creates) an enemy unit in the current GameState. Immediately prior to calling this method, the enemyUnit will be available through calls to methods such as `getEnemies()` and `getGameObjects()`

Parameters:
`enemyUnit` - the enemy unit to be registered with this GameState

See Also:
`getEnemies()`

---

killEnemy

`protected void killEnemy(EnemyUnit enemy)`

Removes an enemy unit from the current GameState. This method will not affect the enemy provided, but will simply remove it from the internal list of enemy unit, preventing it from being tracked further by this GameState, and thus by any classes which use this GameState to determine the active enemy units. If the given EnemyUnit is not in the current GameState, the method will simply return.

Parameters:
`enemy` - the enemy unit to be removed from this GameState.

See Also:
`getEnemies()`
`spawnEnemy(EnemyUnit)`

---

killUnit

`protected void killUnit(Unit unit)`

getPickups

`public Iterator getPickups()`

Retrieves an Iterator which allows access to details of all pickups which are active in this GameState. This refers to all pickups which have not been collected (that is, only those that are "lying on the ground").

Returns:
a templatized Iterator which will traverse through each of the bullets in the current GameState

See Also:
`getNumBullets()`

getGameController

`public GameController getGameController()`

getGameObjects

`public Iterator getGameObjects()`

Retrieves an Iterator which allows access to details of all Game Objects in this GameState. This refers to all instances of each of the following types of GameObject:

- Pickups (see `getPickups()`
- Enemy units (see `getEnemies()`
- Bullets (see `getBullets()`
- Player units (see `getPlayers()`)

Returns:
a templatized Iterator which will traverse through each of the bullets in the current GameState

getAnimations

`public Iterator getAnimations()`

addAnimation

`public void addAnimation(Animation animation)`

spawnPlayer

`protected void spawnPlayer(Location location)`

Spawns (creates) a PlayerUnit at a given location, and adds it to the GameState. This can be either a friendly player or an opposition-controlled player.

Parameters:
`location` - the location at which the player should be created.

(continued from last page)

See Also:
getPlayers()

spawnPickup

```
protected void spawnPickup()
```

Spawns (creates) a pickup, and adds it to the GameState. The details of the pickup which will be spawned are controlled by the PickupFactory - that is, the caller cannot choose the type or location of the pickup which will be spawned.

getGameTime

```
public GameTime getGameTime()
```

Gets details about the current game time in this game state.

Returns:
this GameState's GameTime

getMap

```
public Map getMap()
```

Returns the map which is being used for the current GameState, which includes details about the logical 2D environment over which GameObjects in this GameState are positioned

Returns:
the Map which was created for this GameState during initialization and which represents the 2D environment of this GameState

update

```
public void update(PlayerMoveEvent event)
```

update

```
public void update(ShootEvent event)
```

spawnBullet

```
public void spawnBullet()
```

isSpawningBullets

```
public boolean isSpawningBullets()
```

Determines whether or not there is an active unit which is busy firing, and by implication whether or not another bullet should be fired

Returns:
true if there is a GameObject in the current GameState which is firing bullets, false if there is not

(continued from last page)

removeAllObservers

```
public void removeAllObservers()
```

notifyObservers

```
public void notifyObservers(GameStateChangedEvent event)
```

addObserver

```
public boolean addObserver(Observer observer)
```

removeObserver

```
public boolean removeObserver(Observer observer)
```

countObservers

```
public int countObservers()
```

crimsonportal.googlecode.com.ObjectModel
Class GameTime

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.ObjectModel.GameTime
```

All Implemented Interfaces:
Comparable

---

public class **GameTime**
extends Object
implements Comparable

Constructor Summary

public	GameTime(boolean isPaused)
public	GameTime(long numSeconds)
public	GameTime(GameTime gameTime)

Method Summary

GameTime	clone()
int	compareTo(GameTime g)
long	getNumMilliseconds()
int	getNumSeconds()
boolean	isAfter(GameTime rhs)
boolean	isPaused()
void	pauseTimer()
void	resetTime()
void	startTimer()
String	toString()

Methods inherited from class java.lang.Object

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

Methods inherited from interface java.lang.Comparable
compareTo

Constructors

GameTime

```
public GameTime(boolean isPaused)
```

GameTime

```
public GameTime(long numSeconds)
```

GameTime

```
public GameTime(GameTime gameTime)
```

Methods

startTimer

```
public void startTimer()
```

pauseTimer

```
public void pauseTimer()
```

getNumMilliseconds

```
public long getNumMilliseconds()
```

getNumSeconds

```
public int getNumSeconds()
```

resetTime

```
protected void resetTime()
```

clone

public GameTime **clone**()

isPaused

public boolean **isPaused**()

compareTo

public int **compareTo**(GameTime g)

isAfter

public boolean **isAfter**(GameTime rhs)

toString

public String **toString**()

crimsonportal.googlecode.com.ObjectModel
Class GameTimer

```
java.lang.Object
   |
   +-java.lang.Thread
      |
      +-crimsonportal.googlecode.com.ObjectModel.GameTimer
```

All Implemented Interfaces:
Runnable

---

public class **GameTimer**
extends Thread

Field Summary

protected static	gameTime
protected	gameTimerTasks

Fields inherited from class `java.lang.Thread`

`MAX_PRIORITY`, `MIN_PRIORITY`, `NORM_PRIORITY`

Method Summary

void	addTimer(GameTime time, GameTimerAction action)
void	checkTimers()
static GameTimer	getInstance()
static GameTimer	getInstance(GameTime gameTime)
void	run()

Methods inherited from class `java.lang.Thread`

`activeCount, checkAccess, countStackFrames, currentThread, destroy, dumpStack, enumerate, getAllStackTraces, getContextClassLoader, getDefaultUncaughtExceptionHandler, getId, getName, getPriority, getStackTrace, getState, getThreadGroup, getUncaughtExceptionHandler, holdsLock, interrupt, interrupted, isAlive, isDaemon, isInterrupted, join, join, join, resume, run, setContextClassLoader, setDaemon, setDefaultUncaughtExceptionHandler, setName, setPriority, setUncaughtExceptionHandler, sleep, sleep, start, stop, stop, suspend, toString, yield`

Methods inherited from class `java.lang.Object`

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

Methods inherited from interface java.lang.Runnable
run

Fields

gameTime

protected static crimsonportal.googlecode.com.ObjectModel.GameTime **gameTime**

gameTimerTasks

protected java.util.Vector **gameTimerTasks**

Methods

run

public void **run**()

getInstance

public static GameTimer **getInstance**(GameTime gameTime)

getInstance

public static GameTimer **getInstance**()

addTimer

public void **addTimer**(GameTime time,
GameTimerAction action)

checkTimers

public void **checkTimers**()

crimsonportal.googlecode.com.ObjectModel
Class GameTimerAction

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.ObjectModel.GameTimerAction
```

Direct Known Subclasses:
PickupExpirationAction

---

public abstract class **GameTimerAction**
extends Object

Constructor Summary

public	GameTimerAction()

Method Summary

abstract void	trigger()

Methods inherited from class `java.lang.Object`

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

Constructors

GameTimerAction

`public` **`GameTimerAction`**`()`

Methods

trigger

`public abstract void` **`trigger`**`()`

crimsonportal.googlecode.com.ObjectModel
Class GameTimerTask

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.ObjectModel.GameTimerTask
```

public class **GameTimerTask**
extends Object

Field Summary

protected	triggerAction
protected	triggerTime

Constructor Summary

public	GameTimerTask(GameTime triggerTime, GameTimerAction triggerAction)

Method Summary

boolean	checkTriggered(GameTime currentGameTime)
GameTime	getTriggerTime()
String	toString()

Methods inherited from class java.lang.Object

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

Fields

triggerTime

protected crimsonportal.googlecode.com.ObjectModel.GameTime **triggerTime**

triggerAction

protected crimsonportal.googlecode.com.ObjectModel.GameTimerAction **triggerAction**

Constructors

(continued from last page)

GameTimerTask

```
public GameTimerTask(GameTime triggerTime,
                     GameTimerAction triggerAction)
```

Methods

checkTriggered

```
public boolean checkTriggered(GameTime currentGameTime)
```

---

getTriggerTime

```
protected GameTime getTriggerTime()
```

---

toString

```
public String toString()
```

crimsonportal.googlecode.com.ObjectModel
Class Location

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.ObjectModel.Location
```

public class **Location**
extends Object

Representation of a location (that is, a position which has an X and Y coordinate). This is similar to java's built-in Dimension object, except that the coordinates are expressed as doubles

Constructor Summary

public	`Location(double x, double y)` Creates a new Location with given coordinates.

Method Summary

`Location`	`clone()` Creates a new location which is identical to the current location, that is, it has the same X- and Y-coordinates
`boolean`	`equals(Location rhs)` Tests whether two locations are identical, that is whether they have the same X- and Y-coordinates
`double`	`getDistanceFrom(GameObject object)` Returns the Euclidean (straight-line) distance between this location and a given GameObject
`double`	`getX()` Returns the X-coordinate (horizontal position) of this location
`double`	`getY()` Returns the Y-coordinate (vertical position) of this location
`void`	`setX(double x)` Sets the X-coordinate (horizontal position) of this location
`void`	`setY(double y)` Sets the Y-coordinate (vertical position) of this location
`String`	`toString()` Returns this object converted to a String representation.

Methods inherited from class `java.lang.Object`

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

Constructors

(continued from last page)

Location

```
public Location(double x,
                double y)
```

Creates a new Location with given coordinates. Note that the coordinates are given a context by the way in which they are used - for example, a Location can be given coordinates of (10,20) but the scale of the coordinates is defined by the way in which it is used, not by the Location itself.

Parameters:

`x` - the X-coordinate (horizontal position, on the X-axis) of the new location
`y` - the Y-coordinate (vertical position, on the Y-axis) of the new location

Methods

getX

```
public double getX()
```

Returns the X-coordinate (horizontal position) of this location

Returns:

this location's position on the X-axis

---

setX

```
protected void setX(double x)
```

Sets the X-coordinate (horizontal position) of this location

Parameters:

`x` - the new position of this location on the X-axis

---

getY

```
public double getY()
```

Returns the Y-coordinate (vertical position) of this location

Returns:

this location's position on the Y-axis

---

setY

```
protected void setY(double y)
```

Sets the Y-coordinate (vertical position) of this location

Parameters:

`y` - the new position of this location on the Y-axis

---

getDistanceFrom

```
public double getDistanceFrom(GameObject object)
```

Returns the Euclidean (straight-line) distance between this location and a given GameObject

Parameters:

(continued from last page)

`object` - the object for which the distance from the current location should be returned

Returns:
the straight-line distance between `this` and the location of `object`

---

clone

`public Location clone()`

Creates a new location which is identical to the current location, that is, it has the same X- and Y-coordinates

Returns:
an exact copy of the location `this`

---

equals

`public boolean equals(Location rhs)`

Tests whether two locations are identical, that is whether they have the same X- and Y-coordinates

Parameters:
`rhs` - the location which should be compared (to `this`)

Returns:
true if the given location is identical to this location, false otherwise

---

toString

`public String toString()`

Returns this object converted to a String representation. Returns a string which contains "Location[" + Y-coordinate + "," + X-coordinate + "]"

Returns:
a String representation of this location

crimsonportal.googlecode.com.ObjectModel
Class LocationObject

```
java.lang.Object
   ┗-crimsonportal.googlecode.com.ObjectModel.GameObject
      ┗-crimsonportal.googlecode.com.ObjectModel.LocationObject
```

public class **LocationObject**
extends GameObject

A wrapper class which encapsulates a `Location` within a `GameObject` to allow

Fields inherited from class `crimsonportal.googlecode.com.ObjectModel.GameObject`
`location`, `rotation`, `size`

Constructor Summary

public	`LocationObject(Location location)`
public	`LocationObject(double x, double y)`

Method Summary

String	`getSpriteFilename()`

Methods inherited from class `crimsonportal.googlecode.com.ObjectModel.GameObject`
`getCentreOfObject`, `getRotation`, `getSize`, `getSpriteFilename`, `setCentreOfObject`, `setRotation`, `setSize`, `testOverlapsWith`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

LocationObject

```
public LocationObject(Location location)
```

LocationObject

```
public LocationObject(double x,
                      double y)
```

(continued from last page)

Methods

getSpriteFilename

```
public String getSpriteFilename()
```

Abstract method to be implemented by inheriting classes and which specifies the name of the filename which represents the graphical representation of this game object. For example, if a class called Box is implemented which inherits from GameObject, and a Box is graphically represented by a file named "Box.jpg" then this method might be implemented in the Box class to return the string "Box.jpg". This method allows instances of Game Objects, which are logical objects, to be represented in the presentation layer by having graphical classes query the Game Object's getSpriteFilename method and rendering that file. Note that the specifics of this rendering process are not defined here, but are deferred to the presentation -related classes.

crimsonportal.googlecode.com.ObjectModel
Class Map

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.ObjectModel.Map
```

public class **Map**
extends Object

Field Summary

protected	bgImage
protected	height
protected	offsetX
protected	offsetY
protected	width

Constructor Summary

public	Map()

Method Summary

java.awt.Image	getBGImage()
int	getHeight()
String	getImageFilename()
int	getOffsetX()
int	getOffsetY()
Location	getRandomLocation()
java.awt.Dimension	getSize()
int	getWidth()
void	setHeight(int height)

void	setOffsetX(int offsetX)
void	setOffsetY(int offsetY)
void	setWidth(int width)

Methods inherited from class java.lang.Object

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

Fields

width

protected int **width**

height

protected int **height**

offsetX

protected int **offsetX**

offsetY

protected int **offsetY**

bgImage

protected java.awt.Image **bgImage**

Constructors

Map

public **Map**()

Methods

(continued from last page)

getWidth

```
public int getWidth()
```

---

getHeight

```
public int getHeight()
```

---

getSize

```
public java.awt.Dimension getSize()
```

---

setWidth

```
protected void setWidth(int width)
```

---

setHeight

```
protected void setHeight(int height)
```

---

getOffsetX

```
public int getOffsetX()
```

---

getOffsetY

```
public int getOffsetY()
```

---

setOffsetX

```
protected void setOffsetX(int offsetX)
```

---

setOffsetY

```
protected void setOffsetY(int offsetY)
```

---

getImageFilename

```
public String getImageFilename()
```

(continued from last page)

getBGImage

```
public java.awt.Image getBGImage()
```

getRandomLocation

```
public Location getRandomLocation()
```

crimsonportal.googlecode.com.ObjectModel
Class Pickup

```
java.lang.Object
   |-crimsonportal.googlecode.com.ObjectModel.GameObject
      |-crimsonportal.googlecode.com.ObjectModel.Pickup
```

Direct Known Subclasses:
PickupSingleUse, PickupTimed, WeaponPickup

---

public abstract class **Pickup**
extends GameObject

An abstract class which will be inherited from (indirectly) by any object which can be picked up by a Unit.

Inheriting subclasses will need to implement all the required methods in order to provide the specifics of the pickup, while the basic functionality common to all pickups (such as the mechanism for the expiration of a pickup - that is, when it disappears and is no longer collectable if it is not collected after a certain amount of time) will always be handled by this abstract class.

The following points should be considered when writing inheriting classes:

- All overriding classes **must** be sure to call the `Pickup(Location, GameTime)` constructor specified by this abstract class
- The functionality of the pickup (that is, the handling of what actions are performed when a Pickup is collected by a unit) must be provided by the inheriting class in the `applyTo` method, which will be called when the Pickup is collected.
- Likewise, the `unapplyTo` method must be provided by inheriting classes to provide the details of what actions are taken when the effects of a Pickup are to be removed from a Player (for example, if the Pickup is dropped, it's effect runs out, etc)

Fields inherited from class `crimsonportal.googlecode.com.ObjectModel.GameObject`
`location`, `rotation`, `size`

Constructor Summary

public	`Pickup(Location location, GameTime expirationTime)` Creates a Pickup at a given location, and sets its expiration time (the time at which the pickup will disappear if it is not collected).

Method Summary

abstract void	`applyTo(GameTime gameTime, Unit unit)` Defines the actions to be taken when this Pickup is collected by a Unit.
GameTime	`getExpirationTime()` Returns the time at which this pickup will expire (that is, disappear) if it is not collected
void	`setExpirationTime(GameTime expirationTime)` Sets the time at which this pickup will expire (that is, disappear) if it is not collected
abstract void	`unapplyTo(Unit unit)` Defines the actions to be taken when the effects of this Pickup are removed from a Unit.

Methods inherited from class `crimsonportal.googlecode.com.ObjectModel.GameObject`

getCentreOfObject, getRotation, getSize, getSpriteFilename, setCentreOfObject, setRotation, setSize, testOverlapsWith

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

Pickup

```
public Pickup(Location location,
              GameTime expirationTime)
```

Creates a Pickup at a given location, and sets its expiration time (the time at which the pickup will disappear if it is not collected).

This method **must** be called by all constructors in all inherited classes.

Parameters:

`location` - the location at which the pickup should be created, specified as the centre of the pickup

`expirationTime` - the GameTime at which the pickup will expire - that is, the GameTime at which the pickup will disappear if it has not been collected

Methods

getExpirationTime

```
public final GameTime getExpirationTime()
```

Returns the time at which this pickup will expire (that is, disappear) if it is not collected

Returns:

the time after which this pickup is no longer available to be collected

setExpirationTime

```
protected final void setExpirationTime(GameTime expirationTime)
```

Sets the time at which this pickup will expire (that is, disappear) if it is not collected

Parameters:

`expirationTime` - the time after which this pickup should no longer be available to be collected

applyTo

```
public abstract void applyTo(GameTime gameTime,
         Unit unit)
```

Defines the actions to be taken when this Pickup is collected by a Unit. Overriding methods must be sure to implement any effects of the pickup in this method, which will be called when the Pickup is collected. The effects of this pickup, if it is not permanent, should be reversed in the `unapplyTo(Unit)` method.

Note: To maintain the appropriate protection of the GameModel package classes, overriding methods should make use of the `PickupProxy` class in order to modify the given Unit

(continued from last page)

Parameters:

`gameTime` - will be set to contain the time at which the collection of the pickup happened (that is, the current game time at the time the applyTo method is called)
`unit` - the Unit which is collecting the Pickup

---

unapplyTo

```
public abstract void unapplyTo(Unit unit)
```

Defines the actions to be taken when the effects of this Pickup are removed from a Unit. Overriding methods must be sure to implement the 'undoing' of any effects of the pickup in this method, which will be called when the Pickup is removed from a Unit (if this is applicable), for example if the pickup runs out or is discarded.

Note: To maintain the appropriate protection of the GameModel package classes, overriding methods should make use of the `PickupProxy` class in order to modify the given Unit

Parameters:

`unit` - the Unit which had collected the Pickup and which the pickup should now be removed from

crimsonportal.googlecode.com.ObjectModel
Class PickupExpirationAction

```
java.lang.Object
   |-crimsonportal.googlecode.com.ObjectModel.GameTimerAction
      |-crimsonportal.googlecode.com.ObjectModel.PickupExpirationAction
```

public class **PickupExpirationAction**
extends GameTimerAction

Field Summary

protected	pickup
protected	unitHoldingPickup

Constructor Summary

public	PickupExpirationAction(Pickup pickup, Unit unitHoldingPickup)

Method Summary

String	toString()
void	trigger()

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.GameTimerAction

trigger

Methods inherited from class java.lang.Object

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

Fields

pickup

protected crimsonportal.googlecode.com.ObjectModel.Pickup **pickup**

unitHoldingPickup

protected crimsonportal.googlecode.com.ObjectModel.Unit **unitHoldingPickup**

(continued from last page)

Constructors

PickupExpirationAction

```
public PickupExpirationAction(Pickup pickup,
                              Unit unitHoldingPickup)
```

Methods

trigger

```
public void trigger()
```

---

toString

```
public String toString()
```

crimsonportal.googlecode.com.ObjectModel
Class PickupProxy

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.ObjectModel.PickupProxy
```

public class **PickupProxy**
extends Object

Constructor Summary

public	PickupProxy()

Method Summary

static void	addAnimation(Animation animation)
static Iterator	getEnemiesNear(GameObject object, double rangeRadius) Returns an iterator which will iterate over all the EnemyUnits within a given distance of a given GameObject
static Iterator	getEnemiesNear(Location location, double rangeRadius) Returns an iterator which will iterate over all the EnemyUnits within a given distance of a given location
static void	increaseUnitHealth(Unit unit, double healthValue)
static void	killUnit(Collection units)
static void	killUnit(Unit unit)
static void	scaleUnitHealth(Collection units, double healthValue)
static void	scaleUnitMoveSpeed(Collection units, double moveSpeedMultiplier)
static void	scaleUnitMoveSpeed(Unit unit, double moveSpeedMultiplier)
static void	scaleUnitSize(Collection units, double sizeScale)
static void	scaleUnitSize(Unit unit, double sizeScale)
static void	setGameState(GameState gameState)

Methods inherited from class java.lang.Object

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

Constructors

PickupProxy

```
public PickupProxy()
```

Methods

setGameState

```
public static void setGameState(GameState gameState)
```

scaleUnitSize

```
public static void scaleUnitSize(Unit unit,
         double sizeScale)
```

scaleUnitSize

```
public static void scaleUnitSize(Collection units,
         double sizeScale)
```

increaseUnitHealth

```
public static void increaseUnitHealth(Unit unit,
         double healthValue)
```

scaleUnitHealth

```
public static void scaleUnitHealth(Collection units,
         double healthValue)
```

scaleUnitMoveSpeed

```
public static void scaleUnitMoveSpeed(Unit unit,
         double moveSpeedMultiplier)
```

scaleUnitMoveSpeed

```
public static void scaleUnitMoveSpeed(Collection units,
         double moveSpeedMultiplier)
```

(continued from last page)

killUnit

```
public static void killUnit(Unit unit)
```

killUnit

```
public static void killUnit(Collection units)
```

getEnemiesNear

```
public static Iterator getEnemiesNear(Location location,
         double rangeRadius)
```

Returns an iterator which will iterate over all the EnemyUnits within a given distance of a given location

Parameters:

`location` - the location which is being used in comparison against the enemy units' locations
`rangeRadius` - the maximum distance from `location` which returned enemies may be

Returns:

an iterator over the collection of EnemyUnits which are within a distance of `rangeRadius` of `location`

See Also:

`Location.getDistanceFrom(GameObject)`

getEnemiesNear

```
public static Iterator getEnemiesNear(GameObject object,
         double rangeRadius)
```

Returns an iterator which will iterate over all the EnemyUnits within a given distance of a given GameObject

Parameters:

`object` - the object which is being used in comparison against the enemy units' locations
`rangeRadius` - the maximum distance from `location` which returned enemies may be

Returns:

an iterator over the collection of EnemyUnits which are within a distance of `rangeRadius` of `object`

See Also:

`Location.getDistanceFrom(GameObject)`

addAnimation

```
public static void addAnimation(Animation animation)
```

crimsonportal.googlecode.com.ObjectModel
Class PickupSingleUse

```
java.lang.Object
   |-crimsonportal.googlecode.com.ObjectModel.GameObject
      |-crimsonportal.googlecode.com.ObjectModel.Pickup
         |-crimsonportal.googlecode.com.ObjectModel.PickupSingleUse
```

Direct Known Subclasses:
PickupHealth, PickupNuke

---

public abstract class **PickupSingleUse**
extends Pickup

An abstract class which will be inherited from by any Pickup which can be picked up by a Unit and which provides a 'Single Use' pickup - that is, it is applied to a Unit and is never unapplied. For example, subclasses of the `PickupSingleUse` class might include pickups which permanently increase a Unit's movement speed, or which give a once-off increase to a Unit's health

Inheriting subclasses will need to implement all the required methods in order to provide the specifics of the pickup, while this abstract class hides some unnecessary detail in the inherited `Pickup` abstract class, such as the implementation of the `unapplyTo` method.

The following points should be considered when writing inheriting classes:

- All overriding classes **must** be sure to call the `PickupSingleUse(Location, GameTime)` constructor specified by this abstract class
- The functionality of the pickup (that is, the handling of what actions are performed when a Pickup is collected by a unit) must be provided by the inheriting class in the `applyTo` method, which will be called when the Pickup is collected.
- The `Pickup.unapplyTo(Unit)` method is hidden from inheriting classes by this class, and so it will not be possible for inheriting pickups to have any 'undo' effects. If this behaviour is not desired, it may be better to consider extending from the `PickupTimed` class in stead of from PickupSingleUse

---

Fields inherited from class `crimsonportal.googlecode.com.ObjectModel.GameObject`
`location`, `rotation`, `size`

Constructor Summary

public	`PickupSingleUse(Location location, GameTime expirationTime)`

Method Summary

void	`unapplyTo(Unit unit)` Defines the actions to be taken when the effects of this Pickup are removed from a Unit.

Methods inherited from class `crimsonportal.googlecode.com.ObjectModel.Pickup`
`applyTo`, `getExpirationTime`, `setExpirationTime`, `unapplyTo`

Methods inherited from class `crimsonportal.googlecode.com.ObjectModel.GameObject`
`getCentreOfObject`, `getRotation`, `getSize`, `getSpriteFilename`, `setCentreOfObject`, `setRotation`, `setSize`, `testOverlapsWith`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

PickupSingleUse

```
public PickupSingleUse(Location location,
                       GameTime expirationTime)
```

Methods

unapplyTo

```
public final void unapplyTo(Unit unit)
```

Defines the actions to be taken when the effects of this Pickup are removed from a Unit. This method does nothing, but serves to hide the abstract method in the underlying `Pickup` class from inheriting classes.

Parameters:

`unit` - the Unit which had collected the Pickup and which the pickup should now be removed from

See Also:

`Pickup.unapplyTo(Unit)`

crimsonportal.googlecode.com.ObjectModel
Class PickupTimed

```
java.lang.Object
   ┗-crimsonportal.googlecode.com.ObjectModel.GameObject
      ┗-crimsonportal.googlecode.com.ObjectModel.Pickup
         ┗-crimsonportal.googlecode.com.ObjectModel.PickupTimed
```

Direct Known Subclasses:
PickupGrow, PickupSpeed

---

public abstract class **PickupTimed**
extends Pickup

Fields inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject
location, rotation, size

Constructor Summary

public	PickupTimed(Location location, GameTime expirationTime)

Method Summary

abstract int	getEffectDurationSeconds()
void	startExpirationTimer(GameTime gameTime, Unit unitToUnapplyTo)

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.Pickup
applyTo, getExpirationTime, setExpirationTime, unapplyTo

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject
getCentreOfObject, getRotation, getSize, getSpriteFilename, setCentreOfObject, setRotation, setSize, testOverlapsWith

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

---

Constructors

(continued from last page)

PickupTimed

```
public PickupTimed(Location location,
                   GameTime expirationTime)
```

Methods

startExpirationTimer

```
protected final void startExpirationTimer(GameTime gameTime,
          Unit unitToUnapplyTo)
```

---

getEffectDurationSeconds

```
public abstract int getEffectDurationSeconds()
```

crimsonportal.googlecode.com.ObjectModel
Class PlayerTurnEvent

```
java.lang.Object
   |
   +-java.util.EventObject
      |
      +-crimsonportal.googlecode.com.ObjectModel.PlayerTurnEvent
```

All Implemented Interfaces:
Serializable

---

public class **PlayerTurnEvent**
extends EventObject

Fields inherited from class `java.util.EventObject`
`source`

Constructor Summary

public	PlayerTurnEvent(PlayerUnit playerToMove, double newRotation)
public	PlayerTurnEvent(PlayerUnit playerToMove, Location turnTowardsLocation)

Method Summary

double	getRotation()

Methods inherited from class `java.util.EventObject`
`getSource, toString`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

PlayerTurnEvent

```
public PlayerTurnEvent(PlayerUnit playerToMove,
                       double newRotation)
```

(continued from last page)

PlayerTurnEvent

```
public PlayerTurnEvent(PlayerUnit playerToMove,
                       Location turnTowardsLocation)
```

Methods

getRotation

```
public double getRotation()
```

crimsonportal.googlecode.com.ObjectModel
Class PlayerUnit

```
java.lang.Object
   ┗-crimsonportal.googlecode.com.ObjectModel.GameObject
      ┗-crimsonportal.googlecode.com.ObjectModel.Unit
         ┗-crimsonportal.googlecode.com.ObjectModel.PlayerUnit
```

All Implemented Interfaces:
Observer, PlayerMoveObservable

---

public class **PlayerUnit**
extends Unit
implements PlayerMoveObservable, Observer

Field Summary

protected	DEFAULT_HEALTH

Fields inherited from class crimsonportal.googlecode.com.ObjectModel.Unit

gameState, health, moveSpeed, pickups, strategy

Fields inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject

location, rotation, size

Constructor Summary

public	PlayerUnit(Location location, int moveSpeed, GameState gameState)
public	PlayerUnit(Location location, double moveSpeed, Weapon weapon, GameState gameState)

Method Summary

boolean	addObserver(Observer observer)
PlayerUnit	clone()
int	countObservers()
String	getSpriteFilename()
Weapon	getWeapon()
void	moveTo(Location location)

void	notifyObservers(PlayerMoveEvent event)
void	notifyObservers(PlayerTurnEvent event)
void	removeAllObservers()
boolean	removeObserver(Observer observer)
void	setWeapon(Weapon weapon)
String	toString()
void	update(PlayerTurnEvent e)

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.Unit

addPickup, getHealth, getMoveSpeed, getStrategy, moveTo, setHealth, setMoveSpeed, setStrategy

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject

getCentreOfObject, getRotation, getSize, getSpriteFilename, setCentreOfObject, setRotation, setSize, testOverlapsWith

Methods inherited from class java.lang.Object

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

Methods inherited from interface crimsonportal.googlecode.com.Observer.Observable

addObserver, countObservers, notifyObservers, removeAllObservers, removeObserver

Methods inherited from interface crimsonportal.googlecode.com.Observer.Observer

update

Fields

DEFAULT_HEALTH

```
protected double DEFAULT_HEALTH
```

Constructors

PlayerUnit

```
public PlayerUnit(Location location,
                  int moveSpeed,
                  GameState gameState)
```

(continued from last page)

PlayerUnit

```
public PlayerUnit(Location location,
                  double moveSpeed,
                  Weapon weapon,
                  GameState gameState)
```

Methods

getWeapon

```
public Weapon getWeapon()
```

setWeapon

```
protected void setWeapon(Weapon weapon)
```

clone

```
public PlayerUnit clone()
```

getSpriteFilename

```
public String getSpriteFilename()
```

Abstract method to be implemented by inheriting classes and which specifies the name of the filename which represents the graphical representation of this game object. For example, if a class called Box is implemented which inherits from GameObject, and a Box is graphically represented by a file named "Box.jpg" then this method might be implemented in the Box class to return the string "Box.jpg". This method allows instances of Game Objects, which are logical objects, to be represented in the presentation layer by having graphical classes query the Game Object's getSpriteFilename method and rendering that file. Note that the specifics of this rendering process are not defined here, but are deferred to the presentation -related classes.

notifyObservers

```
public void notifyObservers(PlayerMoveEvent event)
```

addObserver

```
public boolean addObserver(Observer observer)
```

removeObserver

```
public boolean removeObserver(Observer observer)
```

removeAllObservers

```
public void removeAllObservers()
```

countObservers

```
public int countObservers()
```

notifyObservers

```
public void notifyObservers(PlayerTurnEvent event)
```

moveTo

```
public void moveTo(Location location)
```

update

```
public void update(PlayerTurnEvent e)
```

toString

```
public String toString()
```

crimsonportal.googlecode.com.ObjectModel
Class Strategy

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.ObjectModel.Strategy
```

public class **Strategy**
extends Object

Constructor Summary

public	Strategy(GameObject target)
public	Strategy(Location target)

Method Summary

Strategy	clone()
GameObject	getTarget()
void	setTarget(GameObject target)

Methods inherited from class `java.lang.Object`

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

Constructors

Strategy

`public` **`Strategy`**`(GameObject target)`

Strategy

`public` **`Strategy`**`(Location target)`

Methods

getTarget

`public GameObject` **`getTarget`**`()`

(continued from last page)

setTarget

```
protected void setTarget(GameObject target)
```

clone

```
public Strategy clone()
```

crimsonportal.googlecode.com.ObjectModel
Class Unit

```
java.lang.Object
   |-crimsonportal.googlecode.com.ObjectModel.GameObject
      |-crimsonportal.googlecode.com.ObjectModel.Unit
```

Direct Known Subclasses:
EnemyUnit, PlayerUnit

---

public abstract class **Unit**
extends GameObject

Field Summary

protected	gameState
protected	health
protected	moveSpeed
protected	pickups
protected	strategy

Fields inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject

location, rotation, size

Constructor Summary

public	Unit(double size, Location location, Strategy strategy, GameState gameState)

Method Summary

void	addPickup(Pickup pickup)
double	getHealth()
double	getMoveSpeed()
Strategy	getStrategy()
void	moveTo(Location newLocation)

void	setHealth(double health)
void	setMoveSpeed(double moveSpeed)
void	setStrategy(Strategy strategy)

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject

getCentreOfObject, getRotation, getSize, getSpriteFilename, setCentreOfObject, setRotation, setSize, testOverlapsWith

Methods inherited from class java.lang.Object

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

Fields

strategy

protected crimsonportal.googlecode.com.ObjectModel.Strategy **strategy**

health

protected double **health**

moveSpeed

protected double **moveSpeed**

gameState

protected crimsonportal.googlecode.com.ObjectModel.GameState **gameState**

pickups

protected java.util.Collection **pickups**

Constructors

(continued from last page)

Unit

```
public Unit(double size,
            Location location,
            Strategy strategy,
            GameState gameState)
```

Methods

getStrategy

```
public Strategy getStrategy()
```

---

setStrategy

```
protected void setStrategy(Strategy strategy)
```

---

getHealth

```
public double getHealth()
```

---

setHealth

```
protected void setHealth(double health)
```

---

getMoveSpeed

```
public double getMoveSpeed()
```

---

setMoveSpeed

```
protected void setMoveSpeed(double moveSpeed)
```

---

moveTo

```
public void moveTo(Location newLocation)
```

---

addPickup

```
public void addPickup(Pickup pickup)
```

crimsonportal.googlecode.com.ObjectModel
Class Weapon

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.ObjectModel.Weapon
```

public class **Weapon**
extends Object

Constructor Summary

public	Weapon(int clipSize, float firingRate)

Method Summary

Weapon	clone()
int	getClipSize()
float	getFiringRate()
void	setClipSize(int clipSize)
void	setFiringRate(float firingRate)

Methods inherited from class `java.lang.Object`

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

Constructors

Weapon

```
public Weapon(int clipSize,
              float firingRate)
```

Methods

getFiringRate

```
public float getFiringRate()
```

(continued from last page)

setFiringRate

```
protected void setFiringRate(float firingRate)
```

---

getClipSize

```
public int getClipSize()
```

---

setClipSize

```
protected void setClipSize(int clipSize)
```

---

clone

```
public Weapon clone()
```

crimsonportal.googlecode.com.ObjectModel
Class WeaponPickup

```
java.lang.Object
   ┗-crimsonportal.googlecode.com.ObjectModel.GameObject
      ┗-crimsonportal.googlecode.com.ObjectModel.Pickup
         ┗-crimsonportal.googlecode.com.ObjectModel.WeaponPickup
```

public class **WeaponPickup**
extends Pickup

Fields inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject
location, rotation, size

Constructor Summary

public	WeaponPickup(Location location, GameTime expirationTime, Weapon weapon)

Method Summary

void	applyTo(GameTime elapseGameTime, Unit unit)
WeaponPickup	clone()
String	getSpriteFilename()
Weapon	getWeapon()
void	setWeapon(Weapon weapon)
void	unapplyTo(Unit unit)

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.Pickup
applyTo, getExpirationTime, setExpirationTime, unapplyTo

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject
getCentreOfObject, getRotation, getSize, getSpriteFilename, setCentreOfObject, setRotation, setSize, testOverlapsWith

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

WeaponPickup

```
public WeaponPickup(Location location,
                    GameTime expirationTime,
                    Weapon weapon)
```

Methods

getWeapon

```
public Weapon getWeapon()
```

---

setWeapon

```
protected void setWeapon(Weapon weapon)
```

---

applyTo

```
public void applyTo(GameTime elapseGameTime,
          Unit unit)
```

Defines the actions to be taken when this Pickup is collected by a Unit. Overriding methods must be sure to implement any effects of the pickup in this method, which will be called when the Pickup is collected. The effects of this pickup, if it is not permanent, should be reversed in the `Pickup.unapplyTo(Unit)` method.

Note: To maintain the appropriate protection of the GameModel package classes, overriding methods should make use of the `PickupProxy` class in order to modify the given Unit

---

unapplyTo

```
public void unapplyTo(Unit unit)
```

Defines the actions to be taken when the effects of this Pickup are removed from a Unit. Overriding methods must be sure to implement the 'undoing' of any effects of the pickup in this method, which will be called when the Pickup is removed from a Unit (if this is applicable), for example if the pickup runs out or is discarded.

Note: To maintain the appropriate protection of the GameModel package classes, overriding methods should make use of the `PickupProxy` class in order to modify the given Unit

---

clone

```
public WeaponPickup clone()
```

---

getSpriteFilename

```
public String getSpriteFilename()
```

(continued from last page)

Abstract method to be implemented by inheriting classes and which specifies the name of the filename which represents the graphical representation of this game object. For example, if a class called Box is implemented which inherits from GameObject, and a Box is graphically represented by a file named "Box.jpg" then this method might be implemented in the Box class to return the string "Box.jpg". This method allows instances of Game Objects, which are logical objects, to be represented in the presentation layer by having graphical classes query the Game Object's getSpriteFilename method and rendering that file. Note that the specifics of this rendering process are not defined here, but are deferred to the presentation -related classes.

Package

crimsonportal.googlecode.com.ObjectModel.Pickups

crimsonportal.googlecode.com.ObjectModel.Pickups
Class PickupGrow

```
java.lang.Object
   ┗-crimsonportal.googlecode.com.ObjectModel.GameObject
      ┗-crimsonportal.googlecode.com.ObjectModel.Pickup
         ┗-crimsonportal.googlecode.com.ObjectModel.PickupTimed
            ┗-crimsonportal.googlecode.com.ObjectModel.Pickups.PickupGrow
```

public class **PickupGrow**
extends PickupTimed

Field Summary

protected static final	EFFECT_DURATION_SECONDS Value: **10**
protected	sizeMultiplier

Fields inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject

location, rotation, size

Constructor Summary

public	PickupGrow(Location location, GameTime expirationTime, double sizeMultiplier)

Method Summary

void	applyTo(GameTime expirationTime, Unit unit)
int	getEffectDurationSeconds()
String	getSpriteFilename()
String	toString()
void	unapplyTo(Unit unit)

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.PickupTimed

getEffectDurationSeconds, startExpirationTimer

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.Pickup

applyTo, getExpirationTime, setExpirationTime, unapplyTo

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject
getCentreOfObject, getRotation, getSize, getSpriteFilename, setCentreOfObject, setRotation, setSize, testOverlapsWith

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

sizeMultiplier

```
protected double sizeMultiplier
```

EFFECT_DURATION_SECONDS

```
protected static final int EFFECT_DURATION_SECONDS
```

Constant value: **10**

Constructors

PickupGrow

```
public PickupGrow(Location location,
                  GameTime expirationTime,
                  double sizeMultiplier)
```

Methods

applyTo

```
public void applyTo(GameTime expirationTime,
         Unit unit)
```

Defines the actions to be taken when this Pickup is collected by a Unit. Overriding methods must be sure to implement any effects of the pickup in this method, which will be called when the Pickup is collected. The effects of this pickup, if it is not permanent, should be reversed in the Pickup.unapplyTo(Unit) method.

Note: To maintain the appropriate protection of the GameModel package classes, overriding methods should make use of the PickupProxy class in order to modify the given Unit

unapplyTo

```
public void unapplyTo(Unit unit)
```

(continued from last page)

Defines the actions to be taken when the effects of this Pickup are removed from a Unit. Overriding methods must be sure to implement the 'undoing' of any effects of the pickup in this method, which will be called when the Pickup is removed from a Unit (if this is applicable), for example if the pickup runs out or is discarded.

Note: To maintain the appropriate protection of the GameModel package classes, overriding methods should make use of the `PickupProxy` class in order to modify the given Unit

---

getSpriteFilename

```
public String getSpriteFilename()
```

Abstract method to be implemented by inheriting classes and which specifies the name of the filename which represents the graphical representation of this game object. For example, if a class called Box is implemented which inherits from GameObject, and a Box is graphically represented by a file named "Box.jpg" then this method might be implemented in the Box class to return the string "Box.jpg". This method allows instances of Game Objects, which are logical objects, to be represented in the presentation layer by having graphical classes query the Game Object's getSpriteFilename method and rendering that file. Note that the specifics of this rendering process are not defined here, but are deferred to the presentation -related classes.

---

getEffectDurationSeconds

```
public int getEffectDurationSeconds()
```

---

toString

```
public String toString()
```

crimsonportal.googlecode.com.ObjectModel.Pickups
Class PickupHealth

```
java.lang.Object
   ┗-crimsonportal.googlecode.com.ObjectModel.GameObject
      ┗-crimsonportal.googlecode.com.ObjectModel.Pickup
         ┗-crimsonportal.googlecode.com.ObjectModel.PickupSingleUse
            ┗-crimsonportal.googlecode.com.ObjectModel.Pickups.PickupHealth
```

public class **PickupHealth**
extends PickupSingleUse

Field Summary

protected	healthValue

Fields inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject

location, rotation, size

Constructor Summary

public	PickupHealth(Location location, GameTime expirationTime, double healthValue)

Method Summary

void	applyTo(GameTime gameTime, Unit unit)
String	getSpriteFilename()
String	toString()

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.PickupSingleUse

unapplyTo

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.Pickup

applyTo, getExpirationTime, setExpirationTime, unapplyTo

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject

getCentreOfObject, getRotation, getSize, getSpriteFilename, setCentreOfObject, setRotation, setSize, testOverlapsWith

Methods inherited from class java.lang.Object

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

Fields

healthValue

```
protected double healthValue
```

Constructors

PickupHealth

```
public PickupHealth(Location location,
                    GameTime expirationTime,
                    double healthValue)
```

Methods

applyTo

```
public void applyTo(GameTime gameTime,
         Unit unit)
```

Defines the actions to be taken when this Pickup is collected by a Unit. Overriding methods must be sure to implement any effects of the pickup in this method, which will be called when the Pickup is collected. The effects of this pickup, if it is not permanent, should be reversed in the `Pickup.unapplyTo(Unit)` method.

Note: To maintain the appropriate protection of the GameModel package classes, overriding methods should make use of the `PickupProxy` class in order to modify the given Unit

getSpriteFilename

```
public String getSpriteFilename()
```

Abstract method to be implemented by inheriting classes and which specifies the name of the filename which represents the graphical representation of this game object. For example, if a class called Box is implemented which inherits from GameObject, and a Box is graphically represented by a file named "Box.jpg" then this method might be implemented in the Box class to return the string "Box.jpg". This method allows instances of Game Objects, which are logical objects, to be represented in the presentation layer by having graphical classes query the Game Object's getSpriteFilename method and rendering that file. Note that the specifics of this rendering process are not defined here, but are deferred to the presentation -related classes.

toString

```
public String toString()
```

crimsonportal.googlecode.com.ObjectModel.Pickups
Class PickupNuke

```
java.lang.Object
   |-crimsonportal.googlecode.com.ObjectModel.GameObject
      |-crimsonportal.googlecode.com.ObjectModel.Pickup
         |-crimsonportal.googlecode.com.ObjectModel.PickupSingleUse
            |-crimsonportal.googlecode.com.ObjectModel.Pickups.PickupNuke
```

public class **PickupNuke**
extends PickupSingleUse

Fields inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject
location, rotation, size

Constructor Summary

public	PickupNuke(Location location, GameTime expirationTime)

Method Summary

void	applyTo(GameTime gameTime, Unit unit)
String	getSpriteFilename() Specifies the name of the filename which represents the graphical representation of this game object.

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.PickupSingleUse
unapplyTo

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.Pickup
applyTo, getExpirationTime, setExpirationTime, unapplyTo

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject
getCentreOfObject, getRotation, getSize, getSpriteFilename, setCentreOfObject, setRotation, setSize, testOverlapsWith

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructors

(continued from last page)

PickupNuke

```
public PickupNuke(Location location,
                  GameTime expirationTime)
```

Methods

applyTo

```
public void applyTo(GameTime gameTime,
         Unit unit)
```

Defines the actions to be taken when this Pickup is collected by a Unit. Overriding methods must be sure to implement any effects of the pickup in this method, which will be called when the Pickup is collected. The effects of this pickup, if it is not permanent, should be reversed in the `Pickup.unapplyTo(Unit)` method.

Note: To maintain the appropriate protection of the GameModel package classes, overriding methods should make use of the `PickupProxy` class in order to modify the given Unit

---

getSpriteFilename

```
public String getSpriteFilename()
```

Specifies the name of the filename which represents the graphical representation of this game object. For instances of the PickupNuke class, this returns the filename representing the image used by GUIs to render a Nuke pickup (that is, the filename of the image which looks like a Nuke pickup)

Note that the specifics of this rendering process are not defined here, but are deferred to the presentation-related classes.

Returns:

a string representing the filename which is used by GUIs to determine the sprite for Nukes

crimsonportal.googlecode.com.ObjectModel.Pickups
Class PickupSpeed

```
java.lang.Object
   ┗-crimsonportal.googlecode.com.ObjectModel.GameObject
      ┗-crimsonportal.googlecode.com.ObjectModel.Pickup
         ┗-crimsonportal.googlecode.com.ObjectModel.PickupTimed
            ┗-crimsonportal.googlecode.com.ObjectModel.Pickups.PickupSpeed
```

public class **PickupSpeed**
extends PickupTimed

Field Summary

protected static final	EFFECT_DURATION_SECONDS Value: **10**
protected	speedMultiplier

Fields inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject

location, rotation, size

Constructor Summary

public	PickupSpeed(Location location, GameTime expirationTime, double speedMultiplier)

Method Summary

void	applyTo(GameTime gameTime, Unit unit)
PickupSpeed	clone()
int	getEffectDurationSeconds()
String	getSpriteFilename() Specifies the name of the filename which represents the graphical representation of this game object.
String	toString()
void	unapplyTo(Unit unit)

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.PickupTimed

getEffectDurationSeconds, startExpirationTimer

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.Pickup
applyTo, getExpirationTime, setExpirationTime, unapplyTo

Methods inherited from class crimsonportal.googlecode.com.ObjectModel.GameObject
getCentreOfObject, getRotation, getSize, getSpriteFilename, setCentreOfObject, setRotation, setSize, testOverlapsWith

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Fields

speedMultiplier

```
protected double speedMultiplier
```

EFFECT_DURATION_SECONDS

```
protected static final int EFFECT_DURATION_SECONDS
```

Constant value: **10**

Constructors

PickupSpeed

```
public PickupSpeed(Location location,
                   GameTime expirationTime,
                   double speedMultiplier)
```

Methods

applyTo

```
public void applyTo(GameTime gameTime,
         Unit unit)
```

Defines the actions to be taken when this Pickup is collected by a Unit. Overriding methods must be sure to implement any effects of the pickup in this method, which will be called when the Pickup is collected. The effects of this pickup, if it is not permanent, should be reversed in the Pickup.unapplyTo(Unit) method.

Note: To maintain the appropriate protection of the GameModel package classes, overriding methods should make use of the PickupProxy class in order to modify the given Unit

unapplyTo

```
public void unapplyTo(Unit unit)
```

(continued from last page)

Defines the actions to be taken when the effects of this Pickup are removed from a Unit. Overriding methods must be sure to implement the 'undoing' of any effects of the pickup in this method, which will be called when the Pickup is removed from a Unit (if this is applicable), for example if the pickup runs out or is discarded.

Note: To maintain the appropriate protection of the GameModel package classes, overriding methods should make use of the `PickupProxy` class in order to modify the given Unit

---

getEffectDurationSeconds

```
public int getEffectDurationSeconds()
```

---

clone

```
public PickupSpeed clone()
```

---

getSpriteFilename

```
public String getSpriteFilename()
```

Specifies the name of the filename which represents the graphical representation of this game object. For instances of the PickupSpeed class, this returns the filename representing the image used by GUIs to render a Speed pickup (that is, the filename of the image which looks like a Speed pickup)

Note that the specifics of this rendering process are not defined here, but are deferred to the presentation-related classes.

Returns:

a string representing the filename which is used by GUIs to determine the sprite for Speed pickups

---

toString

```
public String toString()
```

Package
crimsonportal.googlecode.com.Observer

crimsonportal.googlecode.com.Observer
Interface Event

public interface **Event**
extends

crimsonportal.googlecode.com.Observer
Interface Observable

All Subinterfaces:
PlayerShootObservable, MoveTimerObservable, PlayerMoveObservable, KeyPressObservable, GameStateChangedObservable

All Known Implementing Classes:
ObserverGroup, Controller, ShootListener, TurnListener

---

public interface **Observable**
extends

Method Summary

boolean	addObserver(Observer observer)
int	countObservers()
void	notifyObservers(EventObject event)
void	removeAllObservers()
boolean	removeObserver(Observer observer)

Methods

notifyObservers

```
public void notifyObservers(EventObject event)
```

---

addObserver

```
public boolean addObserver(Observer observer)
```

---

removeObserver

```
public boolean removeObserver(Observer observer)
```

---

removeAllObservers

```
public void removeAllObservers()
```

(continued from last page)

countObservers

```
public int countObservers()
```

crimsonportal.googlecode.com.Observer
Interface Observer

All Subinterfaces:
PlayerMoveObserver, KeyPressObserver, GameStateChangedObserver

All Known Implementing Classes:
GameCanvas, PlayerTurnObserver, ShootObserver, GameController, PlayerUnit

---

public interface **Observer**
extends

Method Summary

void	update(EventObject event)

Methods

update

```
public void update(EventObject event)
```

crimsonportal.googlecode.com.Observer
Class ObserverGroup

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.Observer.ObserverGroup
```

All Implemented Interfaces:
Observable

---

public class **ObserverGroup**
extends Object
implements Observable

Constructor Summary

public	ObserverGroup()

Method Summary

boolean	addObserver(Observer o)
int	countObservers()
void	notifyObservers(EventObject e)
void	removeAllObservers()
boolean	removeObserver(Observer o)

Methods inherited from class `java.lang.Object`

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

Methods inherited from interface `crimsonportal.googlecode.com.Observer.Observable`

`addObserver, countObservers, notifyObservers, removeAllObservers, removeObserver`

Constructors

ObserverGroup

`public ObserverGroup()`

Methods

(continued from last page)

removeAllObservers

```
public void removeAllObservers()
```

---

addObserver

```
public boolean addObserver(Observer o)
```

---

removeObserver

```
public boolean removeObserver(Observer o)
```

---

notifyObservers

```
public void notifyObservers(EventObject e)
```

---

countObservers

```
public int countObservers()
```

Package

crimsonportal.googlecode.com.Observer.GameState

crimsonportal.googlecode.com.Observer.GameState
Class GameStateChangedEvent

```
java.lang.Object
   |
   +-java.util.EventObject
         |
         +-crimsonportal.googlecode.com.Observer.GameState.GameStateChangedEvent
```

All Implemented Interfaces:
Serializable

---

public class **GameStateChangedEvent**
extends EventObject

Fields inherited from class `java.util.EventObject`
`source`

Constructor Summary

public	`GameStateChangedEvent(GameState gameState)`

Methods inherited from class `java.util.EventObject`
`getSource, toString`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

GameStateChangedEvent

`public` **`GameStateChangedEvent`**`(GameState gameState)`

crimsonportal.googlecode.com.Observer.GameState
Interface GameStateChangedObservable

All Superinterfaces:
Observable

All Known Implementing Classes:
GameState

---

public interface **GameStateChangedObservable**
extends Observable

Methods inherited from interface `crimsonportal.googlecode.com.Observer.Observable`
`addObserver`, `countObservers`, `notifyObservers`, `removeAllObservers`, `removeObserver`

---

crimsonportal.googlecode.com.Observer.GameState
Interface GameStateChangedObserver

All Superinterfaces:
Observer

All Known Implementing Classes:
HUDPanel

---

public interface **GameStateChangedObserver**
extends Observer

Method Summary

void	update(GameStateChangedEvent event)

Methods inherited from interface crimsonportal.googlecode.com.Observer.Observer

update

Methods

update

```
public void update(GameStateChangedEvent event)
```

Package

crimsonportal.googlecode.com.Observer.KeyPress

crimsonportal.googlecode.com.Observer.KeyPress
Interface KeyPressObservable

All Superinterfaces:
Observable

All Known Implementing Classes:
KeyController

---

public interface **KeyPressObservable**
extends Observable

Methods inherited from interface `crimsonportal.googlecode.com.Observer.Observable`
`addObserver`, `countObservers`, `notifyObservers`, `removeAllObservers`, `removeObserver`

---

crimsonportal.googlecode.com.Observer.KeyPress
Interface KeyPressObserver

All Superinterfaces:
Observer

All Known Implementing Classes:
MoveTimer

---

public interface **KeyPressObserver**
extends Observer

Method Summary

void	update(java.awt.event.KeyEvent event)

Methods inherited from interface crimsonportal.googlecode.com.Observer.Observer

update

Methods

update

```
public void update(java.awt.event.KeyEvent event)
```

Package

crimsonportal.googlecode.com.Observer.Player.Move

crimsonportal.googlecode.com.Observer.Player.Move
Class MoveTimer

```
java.lang.Object
   ┗-java.util.Timer
      ┗-crimsonportal.googlecode.com.Observer.Player.Move.MoveTimer
```

All Implemented Interfaces:
KeyPressObserver, MoveTimerObservable

---

public class **MoveTimer**
extends Timer
implements MoveTimerObservable, KeyPressObserver

Constructor Summary

public	MoveTimer(PlayerUnit playerToMove, GameState gameState)

Method Summary

boolean	addObserver(Observer observer)
int	countObservers()
double	getMovementX()
double	getMovementY()
void	notifyObservers(MoveTimerEvent event)
void	removeAllObservers()
boolean	removeObserver(Observer observer)
void	setMovementX(int moveAmountX)
void	setMovementY(int moveAmountY)
void	update(java.awt.event.KeyEvent event)

Methods inherited from class `java.util.Timer`

`cancel, purge, schedule, schedule, schedule, schedule, scheduleAtFixedRate, scheduleAtFixedRate`

Methods inherited from class `java.lang.Object`

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

Methods inherited from interface crimsonportal.googlecode.com.Observer.Observable
addObserver, countObservers, notifyObservers, removeAllObservers, removeObserver

Methods inherited from interface crimsonportal.googlecode.com.Observer.KeyPress.KeyPressObserver
update

Methods inherited from interface crimsonportal.googlecode.com.Observer.Observer
update

Constructors

MoveTimer

```
public MoveTimer(PlayerUnit playerToMove,
                 GameState gameState)
```

Methods

setMovementX

```
public void setMovementX(int moveAmountX)
```

setMovementY

```
public void setMovementY(int moveAmountY)
```

getMovementX

```
public double getMovementX()
```

getMovementY

```
public double getMovementY()
```

notifyObservers

```
public void notifyObservers(MoveTimerEvent event)
```

(continued from last page)

addObserver

```
public boolean addObserver(Observer observer)
```

removeObserver

```
public boolean removeObserver(Observer observer)
```

removeAllObservers

```
public void removeAllObservers()
```

countObservers

```
public int countObservers()
```

update

```
public void update(java.awt.event.KeyEvent event)
```

crimsonportal.googlecode.com.Observer.Player.Move
Class MoveTimerEvent

```
java.lang.Object
   |
   +-java.util.EventObject
      |
      +-crimsonportal.googlecode.com.Observer.Player.Move.MoveTimerEvent
```

All Implemented Interfaces:
Serializable

---

public class **MoveTimerEvent**
extends EventObject

Fields inherited from class `java.util.EventObject`
`source`

Constructor Summary

public	`MoveTimerEvent(MoveTimer object)`

Methods inherited from class `java.util.EventObject`
`getSource, toString`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

MoveTimerEvent

`public` **`MoveTimerEvent`**`(MoveTimer object)`

crimsonportal.googlecode.com.Observer.Player.Move
Interface MoveTimerObservable

All Superinterfaces:
Observable

All Known Implementing Classes:
MoveTimer

---

public interface **MoveTimerObservable**
extends Observable

Methods inherited from interface `crimsonportal.googlecode.com.Observer.Observable`
`addObserver`, `countObservers`, `notifyObservers`, `removeAllObservers`, `removeObserver`

---

crimsonportal.googlecode.com.Observer.Player.Move
Class PlayerMoveEvent

```
java.lang.Object
   |
   +-java.util.EventObject
      |
      +-crimsonportal.googlecode.com.Observer.Player.Move.PlayerMoveEvent
```

All Implemented Interfaces:
Serializable

---

public class **PlayerMoveEvent**
extends EventObject

Fields inherited from class `java.util.EventObject`
`source`

Constructor Summary

public	`PlayerMoveEvent(PlayerUnit playerToMove, GameState gameState, double moveAmountX, double moveAmountY)`

Method Summary

`GameState`	`getGameState()`
`double`	`getMoveAmountX()`
`double`	`getMoveAmountY()`

Methods inherited from class `java.util.EventObject`
`getSource, toString`

Methods inherited from class `java.lang.Object`
`clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait`

Constructors

PlayerMoveEvent

```
public PlayerMoveEvent(PlayerUnit playerToMove,
                       GameState gameState,
                       double moveAmountX,
                       double moveAmountY)
```

Methods

getGameState

```
public GameState getGameState()
```

getMoveAmountX

```
public double getMoveAmountX()
```

getMoveAmountY

```
public double getMoveAmountY()
```

crimsonportal.googlecode.com.Observer.Player.Move
Interface PlayerMoveObservable

All Superinterfaces:
Observable

All Known Implementing Classes:
PlayerUnit

---

public interface **PlayerMoveObservable**
extends Observable

Methods inherited from interface `crimsonportal.googlecode.com.Observer.Observable`
`addObserver`, `countObservers`, `notifyObservers`, `removeAllObservers`, `removeObserver`

---

crimsonportal.googlecode.com.Observer.Player.Move
Interface PlayerMoveObserver

All Superinterfaces:
Observer

All Known Implementing Classes:
GameState

---

public interface **PlayerMoveObserver**
extends Observer

Method Summary

void	update(PlayerMoveEvent event)

Methods inherited from interface crimsonportal.googlecode.com.Observer.Observer

update

Methods

update

```
public void update(PlayerMoveEvent event)
```

Package

crimsonportal.googlecode.com.Observer.Player.Shoot

crimsonportal.googlecode.com.Observer.Player.Shoot
Interface PlayerShootObservable

All Superinterfaces:
Observable

All Known Implementing Classes:
GameCanvas, ShootObserver

---

public interface **PlayerShootObservable**
extends Observable

Methods inherited from interface `crimsonportal.googlecode.com.Observer.Observable`
`addObserver`, `countObservers`, `notifyObservers`, `removeAllObservers`, `removeObserver`

---

crimsonportal.googlecode.com.Observer.Player.Shoot
Class ShootEvent

```
java.lang.Object
   |
   +-java.util.EventObject
      |
      +-crimsonportal.googlecode.com.Observer.Player.Shoot.ShootEvent
```

All Implemented Interfaces:
Serializable

---

public class **ShootEvent**
extends EventObject

Fields inherited from class java.util.EventObject
source

Constructor Summary

public	ShootEvent(boolean isShootStart, Unit shooter, Bullet bullet)

Method Summary

Bullet	getBullet()
boolean	isStartEvent()
boolean	isStopEvent()

Methods inherited from class java.util.EventObject
getSource, toString

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

---

Constructors

ShootEvent

```
public ShootEvent(boolean isShootStart,
                  Unit shooter,
                  Bullet bullet)
```

(continued from last page)

Methods

getBullet

public Bullet **getBullet**()

isStartEvent

public boolean **isStartEvent**()

isStopEvent

public boolean **isStopEvent**()

crimsonportal.googlecode.com.Observer.Player.Shoot
Class ShootObserver

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.Observer.Player.Shoot.ShootObserver
```

All Implemented Interfaces:
PlayerShootObservable, Observer

---

public class **ShootObserver**
extends Object
implements Observer, PlayerShootObservable

Constructor Summary

public	ShootObserver()

Method Summary

boolean	addObserver(Observer observer)
int	countObservers()
void	notifyObservers(ShootEvent event)
void	removeAllObservers()
boolean	removeObserver(Observer observer)
void	update(ShootEvent e)

Methods inherited from class java.lang.Object

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

Methods inherited from interface crimsonportal.googlecode.com.Observer.Observer

update

Methods inherited from interface crimsonportal.googlecode.com.Observer.Observable

addObserver, countObservers, notifyObservers, removeAllObservers, removeObserver

---

Constructors

(continued from last page)

ShootObserver

```
public ShootObserver()
```

Methods

update

```
public void update(ShootEvent e)
```

---

notifyObservers

```
public void notifyObservers(ShootEvent event)
```

---

addObserver

```
public boolean addObserver(Observer observer)
```

---

removeObserver

```
public boolean removeObserver(Observer observer)
```

---

removeAllObservers

```
public void removeAllObservers()
```

---

countObservers

```
public int countObservers()
```

Package

crimsonportal.googlecode.com.Observer.Player.Turn

crimsonportal.googlecode.com.Observer.Player.Turn
Interface PlayerTurnObservable

public interface **PlayerTurnObservable**
extends

Method Summary

boolean	addObserver(Observer observer)
int	countObservers()
void	notifyObservers(PlayerTurnEvent event)
void	removeAllObservers()
boolean	removeObserver(Observer observer)

Methods

notifyObservers

public void **notifyObservers**(PlayerTurnEvent event)

addObserver

public boolean **addObserver**(Observer observer)

removeObserver

public boolean **removeObserver**(Observer observer)

removeAllObservers

public void **removeAllObservers**()

countObservers

public int **countObservers**()

crimsonportal.googlecode.com.Observer.Player.Turn
Class PlayerTurnObserver

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.Observer.Player.Turn.PlayerTurnObserver
```

All Implemented Interfaces:
Observer

---

public class **PlayerTurnObserver**
extends Object
implements Observer

Field Summary

protected	controlledPlayer

Constructor Summary

public	PlayerTurnObserver(PlayerUnit playerToTurn)

Method Summary

void	update(PlayerTurnEvent e)

Methods inherited from class `java.lang.Object`

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

Methods inherited from interface `crimsonportal.googlecode.com.Observer.Observer`

update

Fields

controlledPlayer

`protected crimsonportal.googlecode.com.ObjectModel.PlayerUnit` **`controlledPlayer`**

Constructors

PlayerTurnObserver

`public` **`PlayerTurnObserver`**`(PlayerUnit playerToTurn)`

(continued from last page)

Methods

update

```
public void update(PlayerTurnEvent e)
```

Package
crimsonportal.googlecode.com.Proxy

crimsonportal.googlecode.com.Proxy
Class Proxy

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.Proxy.Proxy
```

Direct Known Subclasses:
SpriteProxy

---

public abstract class **Proxy**
extends Object

Constructor Summary

public	Proxy()

Method Summary

abstract Object	generateProxyObject(String key)
Object	get(String key)

Methods inherited from class `java.lang.Object`

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

Constructors

Proxy

```
public Proxy()
```

Methods

get

```
public final Object get(String key)
```

---

generateProxyObject

```
public abstract Object generateProxyObject(String key)
```

crimsonportal.googlecode.com.Proxy
Interface ProxyObject

All Known Implementing Classes:
Sprite

public interface **ProxyObject**
extends

crimsonportal.googlecode.com.Proxy
Class Sprite

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.Proxy.Sprite
```

All Implemented Interfaces:
ProxyObject

---

public class **Sprite**
extends Object
implements ProxyObject

Constructor Summary

public	Sprite(String filename)

Method Summary

java.awt.image.BufferedImage	toImage()

Methods inherited from class java.lang.Object
clone, equals, finalize, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

---

Constructors

Sprite

```
public Sprite(String filename)
```

Methods

toImage

```
public java.awt.image.BufferedImage toImage()
```

crimsonportal.googlecode.com.Proxy
Class SpriteProxy

```
java.lang.Object
   |-crimsonportal.googlecode.com.Proxy.Proxy
      |-crimsonportal.googlecode.com.Proxy.SpriteProxy
```

public class **SpriteProxy**
extends Proxy

Constructor Summary

public	SpriteProxy()

Method Summary

Sprite	generateProxyObject(String key)

Methods inherited from class crimsonportal.googlecode.com.Proxy.Proxy

generateProxyObject, get

Methods inherited from class java.lang.Object

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

Constructors

SpriteProxy

public **SpriteProxy**()

Methods

generateProxyObject

public Sprite **generateProxyObject**(String key)

Package
crimsonportal.googlecode.com.terrain

crimsonportal.googlecode.com.terrain
Class InvalidTerrainException

```
java.lang.Object
   |-java.lang.Throwable
      |-java.lang.Exception
         |-crimsonportal.googlecode.com.terrain.InvalidTerrainException
```

All Implemented Interfaces:
Serializable

---

public class **InvalidTerrainException**
extends Exception

Constructor Summary

public	InvalidTerrainException(String msg)

Methods inherited from class `java.lang.Throwable`

`fillInStackTrace, getCause, getLocalizedMessage, getMessage, getStackTrace, initCause, printStackTrace, printStackTrace, printStackTrace, setStackTrace, toString`

Methods inherited from class `java.lang.Object`

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

Constructors

InvalidTerrainException

`public` **`InvalidTerrainException`**`(String msg)`

crimsonportal.googlecode.com.terrain
Class Terrain

```
java.lang.Object
   |
   +-crimsonportal.googlecode.com.terrain.Terrain
```

public class **Terrain**
extends Object

Field Summary

protected	data
protected	height
public	highestX
public	highestY
protected	width

Constructor Summary

public	Terrain(int height, int width)

Method Summary

static String	byteToHex(byte b)
static double	cleanDegrees(double degrees)
Location	convertTerrainToMapLocation(int y, int x, Map map)
int	getHeightAt(double y, double x, int mapHeight, int mapWidth)
int	getHeightAt(double y, double x, Map map)
int	getHeightAt(int y, int x)
int	getHeightAt(Location location, Map map)
Location	getMoveWithGradient(Location fromPoint, Location toPoint, Map map)
int	getPeakHeight()

void	loadTerrain(String filename)
static int	unsignedByteToInt(byte b)

Methods inherited from class java.lang.Object

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

Fields

data

protected byte **data**

height

protected int **height**

width

protected int **width**

highestY

public int **highestY**

highestX

public int **highestX**

Constructors

Terrain

public **Terrain**(int height,
int width)

Methods

(continued from last page)

loadTerrain

```
public void loadTerrain(String filename)
  throws FileNotFoundException,
         InvalidTerrainException,
         IOException
```

getHeightAt

```
protected int getHeightAt(int y,
         int x)
```

getHeightAt

```
public int getHeightAt(double y,
         double x,
         int mapHeight,
         int mapWidth)
```

getPeakHeight

```
public int getPeakHeight()
```

getHeightAt

```
public int getHeightAt(double y,
         double x,
         Map map)
```

getHeightAt

```
public int getHeightAt(Location location,
         Map map)
```

unsignedByteToInt

```
protected static int unsignedByteToInt(byte b)
```

getMoveWithGradient

```
public Location getMoveWithGradient(Location fromPoint,
         Location toPoint,
         Map map)
```

byteToHex

```
protected static String byteToHex(byte b)
```

cleanDegrees

```
protected static double cleanDegrees(double degrees)
```

convertTerrainToMapLocation

```
public Location convertTerrainToMapLocation(int y,
                                            int x,
                                            Map map)
```

Index

H

I

K

L

M

N

O

P

R

S

T

U

V

W