PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

PREV CLASS NEXT CLASS FRAMES NO FRAMES ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD

Class PlaneSprite

```
java.lang.Object
    java.awt.Component
        java.awt.Container
            javax.swing.JComponent
                javax.swing.JPanel
                    SpriteSheet
                        PlaneSprite
```

All Implemented Interfaces:

`java.awt.event.ActionListener, java.awt.event.KeyListener, java.awt.event.MouseListener, java.awt.event.MouseMotionListener, java.awt.image.ImageObserver, java.awt.MenuContainer, java.io.Serializable, java.lang.Runnable, java.util.EventListener, javax.accessibility.Accessible, javax.swing.event.MouseInputListener`

```
public class PlaneSprite
extends SpriteSheet
implements java.awt.event.ActionListener, java.lang.Runnable, java.awt.event.KeyListener, javax.swing.event.MouseInputListener
```

This is the PlaneSprite class and will manage the PlaneSprite properties. This class can move the plane, shoot missles, draw the plane, etc.

See Also:

`Serialized Form`

Nested Class Summary

Nested classes/interfaces inherited from class javax.swing.JPanel

`javax.swing.JPanel.AccessibleJPanel`

Nested classes/interfaces inherited from class javax.swing.JComponent

`javax.swing.JComponent.AccessibleJComponent`

Nested classes/interfaces inherited from class java.awt.Container

`java.awt.Container.AccessibleAWTContainer`

Nested classes/interfaces inherited from class java.awt.Component

`java.awt.Component.AccessibleAWTComponent, java.awt.Component.BaselineResizeBehavior, java.awt.Component.BltBufferStrategy, java.awt.Component.FlipBufferStrategy`

Field Summary

Fields

Modifier and Type	Field and Description
`private java.util.List<Missile>`	**`Ammo`**
`boolean`	**`didPlaneFire`**
`private java.awt.image.BufferedImage`	**`img1`**
`private boolean`	**`isPlaneHit`**
`private int`	**`maxAmmo`**
`private boolean`	**`missileFired`**
`java.util.List<Missile>`	**`missiles`**

`private boolean`	`planeDestroyed`
`private boolean`	`planeDown`
`private boolean`	`planeLeft`
`int`	`planeLife`
`private boolean`	`planeRight`
`private boolean`	`planeUp`
`private static java.awt.image.BufferedImage[]`	`sprites1`
`private boolean`	`usingKeyboard`
`private int`	`x`
`private int`	`y`

Fields inherited from class javax.swing.JComponent

`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

`accessibleContext, 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

Constructors

Constructor and Description
`PlaneSprite()` This is the constructor for creating a plane instance for the player.

Method Summary

All Methods   Instance Methods   Concrete Methods   Deprecated Methods

Modifier and Type	Method and Description
`void`	`actionPerformed(java.awt.event.ActionEvent arg0)` Invoke when action is used.
`void`	`addPlaneLife(int x)` This method will add life to the PlaneSprite.
`java.util.List<Missile>`	`ammo()` This method will return a List of the PlaneSprites current ammo.
`void`	`ammoLoad()` This will load 5 shots into the PlaneSprite missle launcher.
`boolean`	`didmissileFired()`
`boolean`	`didPlaneFire()` This will return the boolean for whether or not the plane fired.
`void`	`doDrawing(java.awt.Graphics g)` This method keeps is in charge of drawing the PlaneSprite on the canvas.
`java.awt.Rectangle`	`getBounds()` This will return the bounds of the PlaneSprite used for collision.

java.awt.Rectangle	getBounds2() This method will return the bounds of the plane sprite
int	getH() This will return the height of the PlaneSprite based on the height of the image.
boolean	getKeyboardStatus() This method lets the game know that the user is using the keyboard and flips the PlaneSprite variable.
int	getLife() This method will return the current life of the PlaneSprite
java.awt.image.BufferedImage	getPlane() This will return the current image of the plane.
java.awt.image.BufferedImage[]	getSprites() This method will return all the sprites associated with the PlaneSprite's movement.
int	getW() This method will return the width of the PlaneSprite.
int	getxPosition() This method will return the current x coordinate of the PlaneSprite.
int	getyPosition() This method will return the y coordinate of the PlaneSprite.
boolean	isDead() This will return true if the PlaneSprite life is 0 other wise it will return false.
boolean	isDestroyed() This method will return a boolean for whether or not the PlaneSprite has been destoryed.
void	isHit() This method will lower the PlaneSprite's health if it's been hit.
boolean	isPlaneHit() This method will return whether for not the plane has been hit.
void	keyPressed(java.awt.event.KeyEvent e) This method will check for key being pressed on the keyboard, and using the keyEvent code it will set the PlaneSprite movement variables.
void	keyReleased(java.awt.event.KeyEvent e) This is used to manage key released events.
void	keyTyped(java.awt.event.KeyEvent e) **Deprecated.** not in use
private void	loadImage()
void	missileFired(boolean fire)
java.util.List<Missile>	missiles()
void	mouseClicked(java.awt.event.MouseEvent e)
void	mouseDragged(java.awt.event.MouseEvent e) **Deprecated.** not in use
void	mouseEntered(java.awt.event.MouseEvent e) **Deprecated.** not in use
void	mouseExited(java.awt.event.MouseEvent e) **Deprecated.** not in use
void	mouseMoved(java.awt.event.MouseEvent e) This method will manage the mouse movement and allows for a little bit of wiggling in the mouse before moving the PlaneSprite.

void	`mousePressed(java.awt.event.MouseEvent e)` This method will set the fire to false, this is crucial in the in preventing the PlaneSprite from firing several bullet per frame.
void	`mouseReleased(java.awt.event.MouseEvent e)` This method will allow the plane to fire a missle on release of the mosue button.
void	`moveDown()` This method will move the PlaneSprite image on the canvas down on the y plane from it current location by 3.
void	`moveLeft()` This method will move the PlaneSprite image on the canvas over to the left by 3 on the x plane.
void	`moveRight()` This method will move the PlaneSprite image on the canvas over to the right by 3 on the x plane.
void	`moveUp()` This method will move the PlaneSprite iamge on the canvas up by 3 on the y plane.
Missile	`projectile()` This method is used to remove a missle from the list when the missle is fired.
void	`run()` **Deprecated.** not in use
void	`setPlaneDestroyed(boolean planeDestroyed)` This method will set if the PlaneSprite has been destoryed using a boolean.
void	`setPlaneDown(boolean i)`
void	`setPlaneLeft(boolean i)`
void	`setPlaneRight(boolean i)`
void	`setPlaneUp(boolean i)`
void	`setxPosition(int xPosition)` This method is used to set the x coordinate of the PlaneSprite.
void	`setyPosition(int yPosition)` This method is used to set the y coordinate of the PlaneSprite.

Methods inherited from class SpriteSheet

getHeight, getWidth

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, getActionForKeyStroke, getActionMap, getAlignmentX, getAlignmentY, getAncestorListeners, getAutoscrolls, getBaseline, getBaselineResizeBehavior, getBorder, getBounds, getClientProperty, getComponentGraphics, getComponentPopupMenu, getConditionForKeyStroke, getDebugGraphicsOptions, getDefaultLocale, getFontMetrics, getGraphics, getInheritsPopupMenu, getInputMap, getInputMap, getInputVerifier, getInsets, getInsets, getListeners, getLocation, getMaximumSize, getMinimumSize, getNextFocusableComponent, getPopupLocation, getPreferredSize, getRegisteredKeyStrokes, getRootPane, getSize, getToolTipLocation, getToolTipText, getToolTipText, getTopLevelAncestor, getTransferHandler, getVerifyInputWhenFocusTarget, getVetoableChangeListeners, getVisibleRect, getX, getY, grabFocus, hide, isDoubleBuffered, isLightweightComponent, isManagingFocus, isOpaque, isOptimizedDrawingEnabled, isPaintingForPrint, isPaintingOrigin, isPaintingTile, isRequestFocusEnabled, isValidateRoot, paint, paintBorder, paintChildren, paintComponent, paintImmediately, paintImmediately, 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

Methods inherited from class java.awt.Container

add, add, add, add, add, addContainerListener, addImpl, addPropertyChangeListener, addPropertyChangeListener, applyComponentOrientation, areFocusTraversalKeysSet, countComponents, deliverEvent, doLayout, findComponentAt, findComponentAt, getComponent, getComponentAt, getComponentAt, getComponentCount, getComponents, getComponentZOrder, getContainerListeners, getFocusTraversalKeys, getFocusTraversalPolicy, getLayout, getMousePosition, insets, invalidate, isAncestorOf, isFocusCycleRoot, isFocusCycleRoot, isFocusTraversalPolicyProvider, isFocusTraversalPolicySet, layout, list, list, locate, minimumSize, paintComponents, preferredSize, printComponents, processContainerEvent, processEvent, remove, remove, removeAll, removeContainerListener, setComponentZOrder, setFocusCycleRoot, setFocusTraversalPolicy, setFocusTraversalPolicyProvider, setLayout, transferFocusDownCycle, validate, validateTree

Methods inherited from class java.awt.Component

action, add, addComponentListener, addFocusListener, addHierarchyBoundsListener, addHierarchyListener, addInputMethodListener, addKeyListener, addMouseListener, addMouseMotionListener, addMouseWheelListener, bounds, checkImage, checkImage, coalesceEvents, contains, createImage, createImage, createVolatileImage, createVolatileImage, disableEvents, dispatchEvent, enable, enableEvents, enableInputMethods, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange, getBackground, getColorModel, getComponentListeners, getComponentOrientation, getCursor, getDropTarget, getFocusCycleRootAncestor, getFocusListeners, getFocusTraversalKeysEnabled, getFont, getForeground, getGraphicsConfiguration, getHierarchyBoundsListeners, getHierarchyListeners, getIgnoreRepaint, getInputContext, getInputMethodListeners, getInputMethodRequests, getKeyListeners, getLocale, getLocation, getLocationOnScreen, getMouseListeners, getMouseMotionListeners, getMousePosition, getMouseWheelListeners, getName, getParent, getPeer, getPropertyChangeListeners, getPropertyChangeListeners, getSize, getToolkit, getTreeLock, gotFocus, handleEvent, hasFocus, imageUpdate, inside, isBackgroundSet, isCursorSet, isDisplayable, isEnabled, isFocusable, isFocusOwner, isFocusTraversable, isFontSet, isForegroundSet, isLightweight, isMaximumSizeSet, isMinimumSizeSet, isPreferredSizeSet, isShowing, isValid, isVisible, keyDown, keyUp, list, list, list, location, lostFocus, mouseDown, mouseDrag, mouseEnter, mouseExit, mouseMove, mouseUp, move, nextFocus, paintAll, postEvent, prepareImage, prepareImage, processComponentEvent, processFocusEvent, processHierarchyBoundsEvent, processHierarchyEvent, processInputMethodEvent, processMouseWheelEvent, remove, removeComponentListener, removeFocusListener, removeHierarchyBoundsListener, removeHierarchyListener, removeInputMethodListener, removeKeyListener, removeMouseListener, removeMouseMotionListener, removeMouseWheelListener, removePropertyChangeListener, removePropertyChangeListener, repaint, repaint, repaint, resize, resize, setBounds, setBounds, setComponentOrientation, setCursor, setDropTarget, setFocusable, setFocusTraversalKeysEnabled, setIgnoreRepaint, setLocale, setLocation, setLocation, setName, setSize, setSize, show, show, size, toString, transferFocus, transferFocusBackward, transferFocusUpCycle

Methods inherited from class java.lang.Object

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

Field Detail

sprites1

```
private static java.awt.image.BufferedImage[] sprites1
```

maxAmmo

```
private final int maxAmmo
```

See Also:

Constant Field Values

Ammo

```
private final java.util.List<Missile> Ammo
```

planeLife

```
public int planeLife
```

didPlaneFire

```
public boolean didPlaneFire
```

missiles

```
public java.util.List<Missile> missiles
```

planeDestroyed

```
private boolean planeDestroyed
```

img1

```
private java.awt.image.BufferedImage img1
```

x

```
private int x
```

y

```
private int y
```

planeRight

```
private boolean planeRight
```

planeLeft

```
private boolean planeLeft
```

planeUp

```
private boolean planeUp
```

planeDown

```
private boolean planeDown
```

missileFired

```
private boolean missileFired
```

isPlaneHit

```
private boolean isPlaneHit
```

usingKeyboard

```
private boolean usingKeyboard
```

Constructor Detail

PlaneSprite

```
public PlaneSprite()
```

This is the constructor for creating a plane instance for the player.

Method Detail

ammoLoad

```
public void ammoLoad()
```

This will load 5 shots into the PlaneSprite missle launcher.

ammo

```
public java.util.List<Missile> ammo()
```

This method will return a List of the PlaneSprites current ammo.

Returns:

actionPerformed

```
public void actionPerformed(java.awt.event.ActionEvent arg0)
```

Invoke when action is used.

Specified by:

`actionPerformed` in interface `java.awt.event.ActionListener`

missiles

```
public java.util.List<Missile> missiles()
```

Returns:

didPlaneFire

```
public boolean didPlaneFire()
```

This will return the boolean for whether or not the plane fired.

Returns:

a boolean for if the planefired

doDrawing

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

This method keeps is in charge of drawing the PlaneSprite on the canvas. This method will draw the proper image for the plane movement based on the proper combination.

Parameters:

`g` - is the canvas for images and sprite to be drawn on.

getBounds

```
public java.awt.Rectangle getBounds()
```

This will return the bounds of the PlaneSprite used for collision.

Overrides:

getBounds in class java.awt.Component

Returns:

the bounds of the PlaneSprite

getW

```
public int getW()
```

This method will return the width of the PlaneSprite.

Returns:

the width of the PlaneSprite.

getH

```
public int getH()
```

This will return the height of the PlaneSprite based on the height of the image.

Returns:

the height of the PlaneSprite

getPlane

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

This will return the current image of the plane.

Returns:

the image of the sprite.

getSprites

```
public java.awt.image.BufferedImage[] getSprites()
```

This method will return all the sprites associated with the PlaneSprite's movement.

Overrides:

getSprites in class SpriteSheet

Returns:

a array of all the sprites

getxPosition

```
public int getxPosition()
```

This method will return the current x coordinate of the PlaneSprite.

Returns:

the x cordinate for the PlaneSprite

setxPosition

```
public void setxPosition(int xPosition)
```

This method is used to set the x coordinate of the PlaneSprite.

Parameters:

xPosition - the x coordinate you want to for the PlaneSprite.

getyPosition

```
public int getyPosition()
```

This method will return the y coordinate of the PlaneSprite.

Returns:

the x coordinate of PlaneSprite

setyPosition

```
public void setyPosition(int yPosition)
```

This method is used to set the y coordinate of the PlaneSprite.

Parameters:

yPosition - the y coordinate you want to set for the PlaneSprite.

isDestroyed

```
public boolean isDestroyed()
```

This method will return a boolean for whether or not the PlaneSprite has been destoryed.

Returns:

a boolean for ig the PlaneSprite has been destoryed.

setPlaneDestroyed

```
public void setPlaneDestroyed(boolean planeDestroyed)
```

This method will set if the PlaneSprite has been destoryed using a boolean.

Parameters:

planeDestroyed - a boolean for whether or not the plane is destoryed

keyPressed

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

This method will check for key being pressed on the keyboard, and using the keyEvent code it will set the PlaneSprite movement variables.

Specified by:

keyPressed in interface java.awt.event.KeyListener

Parameters:

e - the key that was pressed.

keyReleased

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

This is used to manage key released events. This will reset the PlaneSpirte variables aftered the key pressed event.

Specified by:

keyReleased in interface java.awt.event.KeyListener

Parameters:

e - the key released.

keyTyped

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

Deprecated. *not in use*

Abstact method -- Inherited from Abstract Class

Specified by:

`keyTyped in interface java.awt.event.KeyListener`

setPlaneDown

```
public void setPlaneDown(boolean i)
```

Parameters:

`i -`

setPlaneUp

```
public void setPlaneUp(boolean i)
```

Parameters:

`i -`

setPlaneLeft

```
public void setPlaneLeft(boolean i)
```

Parameters:

`i -`

setPlaneRight

```
public void setPlaneRight(boolean i)
```

Parameters:

`i -`

loadImage

```
private void loadImage()
```

missileFired

```
public void missileFired(boolean fire)
```

Parameters:

`fire -`

didmissileFired

```
public boolean didmissileFired()
```

Returns:

mouseClicked

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

Specified by:

`mouseClicked in interface java.awt.event.MouseListener`

Parameters:

`e -`

mouseDragged

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

Deprecated. *not in use*
Abstact method -- Inherited from Abstract Class

Specified by:

`mouseDragged in interface java.awt.event.MouseMotionListener`

mouseEntered

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

Deprecated. *not in use*
Abstact method -- Inherited from Abstract Class

Specified by:

`mouseEntered in interface java.awt.event.MouseListener`

mouseExited

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

Deprecated. *not in use*
Abstact method -- Inherited from Abstract Class

Specified by:

`mouseExited in interface java.awt.event.MouseListener`

mouseMoved

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

This method will manage the mouse movement and allows for a little bit of wiggling in the mouse before moving the PlaneSprite.

Specified by:

`mouseMoved in interface java.awt.event.MouseMotionListener`

Parameters:

`e - the MouseEvent for being moved.`

mousePressed

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

This method will set the fire to false, this is crucial in the in preventing the PlaneSprite from firing several bullet per frame.

Specified by:

`mousePressed in interface java.awt.event.MouseListener`

Parameters:

`e - the MouseEvent for being pressed.`

getKeyboardStatus

```
public boolean getKeyboardStatus()
```

This method lets the game know that the user is using the keyboard and flips the PlaneSprite variable.

Returns:

`a boolean for if the keyboard is in use.`

mouseReleased

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

This method will allow the plane to fire a missle on release of the mosue button.

Specified by:

`mouseReleased in interface java.awt.event.MouseListener`

Parameters:

`e - the MouseEvent for mousebutton released.`

moveDown

```
public void moveDown()
```

This method will move the PlaneSprite image on the canvas down on the y plane from it current location by 3.

moveLeft

```
public void moveLeft()
```

This method will move the PlaneSprite image on the canvas over to the left by 3 on the x plane.

moveRight

```
public void moveRight()
```

This method will move the PlaneSprite image on the canvas over to the right by 3 on the x plane.

moveUp

```
public void moveUp()
```

This method will move the PlaneSprite iamge on the canvas up by 3 on the y plane.

projectile

```
public Missile projectile()
```

This method is used to remove a missle from the list when the missle is fired.

Returns:

`the missle from the ammo list that was just "fired".`

run

```
public void run()
```

Deprecated. *not in use*
Abstact method -- Inherited from Abstract Class

Specified by:

`run in interface java.lang.Runnable`

isPlaneHit

```
public boolean isPlaneHit()
```

This method will return whether for not the plane has been hit.

Returns:

`a boolean of the status of if the plane as been hit.`

isHit

```
public void isHit()
```

This method will lower the PlaneSprite's health if it's been hit.

getLife

```
public int getLife()
```

This method will return the current life of the PlaneSprite

Returns:

`the current life of the PlaneSprite`

isDead

```
public boolean isDead()
```

This will return true if the PlaneSprite life is 0 other wise it will return false.

Returns:

`whether or not the plane is dead`

addPlaneLife

```
public void addPlaneLife(int x)
```

This method will add life to the PlaneSprite.

Parameters:

`x - this is the number of live you want to add to the PlaneSprite.`

getBounds2

```
public java.awt.Rectangle getBounds2()
```

This method will return the bounds of the plane sprite

Returns:

`the bounds of the planesprite`

PACKAGE CLASS USE TREE DEPRECATED INDEX HELP

PREV CLASS NEXT CLASS FRAMES NO FRAMES ALL CLASSES

SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIELD | CONSTR | METHOD