Class TelaJogo

java.lang.Object
  java.awt.Component
    java.awt.Container
      javax.swing.JComponent
        javax.swing.JPanel
          TelaBase
            TelaJogo

All Implemented Interfaces:

ActionListener, ImageObserver, MenuContainer, Serializable, EventListener, Accessible

---

```
public class TelaJogo
extends TelaBase
```

Classe da Tela principal do jogo

Version:

1.0

Author:

Arthur dos Santos Rezende

See Also:

Serialized Form

Nested Class Summary

Nested Classes

Modifier and Type	Class	Description
private class	TelaJogo.AlgodogDoce	Classe para o Algodog Doce, um inimigo que rola na direção do jogador e espalha algodão pelo caminho
private class	TelaJogo.Alho	Classe para o Alho, um "mini boss" que se divide em dois ao ser atingido
private class	TelaJogo.Armandibula	Classe para o Armandíbula, um inimigo que morde quem pisa nele enquanto dorme
private class	TelaJogo.Chocochato	Classe para o Chocochato, um inimigo resistente que não ataca o jogador, mas proteje outros inimigos atrás dele
class	TelaJogo.DungeonManager	Classe de gerenciamento das dungeons
private class	TelaJogo.Flyme	Classe para o Flyme, basicamente um slime que voa
class	TelaJogo.GameKeyAdapter	Classe para o leitor te teclas responsável por controlar o jogador
private class	TelaJogo.GigaBot	Classe para o Gigabot, um "mini boss" que basicamente é um slimebot maior e mais resistente
private class	TelaJogo.Inimigo	Classe para os inimigos do jogo
private class	TelaJogo.Malandranha	Classe para o Malandranha, um inimigo que pode escalar paredes e persegue o jogador constantemente

private class	TelaJogo.Morcerango	Classe para o Morcerango, um inimigo que dorme em paredes e acorda caso outro inimigo na mesma dungeon seja estourado.
private class	TelaJogo.Prato	Classe para o Prato, um inimigo que atira facas em uma determinada direção
private class	TelaJogo.QueijoBoxer	Classe para o Queijo Boxer, um inimigo com um braço extensível que usa para socar o jogador.
private class	TelaJogo.Slime	Classe para o Slime, o imimigo básico
private class	TelaJogo.SlimeBot	Classe para o Slimebot, um inimigo que cria um raio laser em uma determinada direção

Nested classes/interfaces inherited from class TelaBase

TelaBase.EstadoJogo

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

JPanel.AccessibleJPanel

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

JComponent.AccessibleJComponent

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

Container.AccessibleAWTContainer

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

Component.AccessibleAWTComponent, Component.BaselineResizeBehavior, Component.BltBufferStrategy, Component.FlipBufferStrategy

Field Summary

Fields

Modifier and Type	Field	Description
private ArrayList<Alert>	alertas	Uma lista contendo todas as partículas de alerta presentes na tela
private Image	alertImage	Imagens únicas
private Image	algodaoImg	Imagens únicas
private Image[]	algodogDoceImgs	Arrays de imagens
private Image[]	alhoImgs	Arrays de imagens
private Image[]	armandibulaImgs	Arrays de imagens
private Image[]	backgroundImgs	Arrays de imagens

private Player	batata	O personagem jogável do jogo (Duque Batata)
private Image[]	batataImgs	Arrays de imagens
private Image[]	bracoImgs	Arrays de imagens
private Image[]	cenouraImgs	Arrays de imagens
private ArrayList<Projetil>	cenouras	Uma lista para gerenciar os projéteis do jogador (as cenouras)
private Image[]	chocochatoImgs	Arrays de imagens
private TelaJogo.DungeonManager	dungeonManager	Variável usada para gerenciar a dungeon atual
private int	enemyCount	Guarda a quantidade de inimigos na tela
private Image[]	facaImgs	Arrays de imagens
private Image[]	flymeImgs	Arrays de imagens
private TelaJogo.GameKeyAdapter	gameKeyAdapter	O leitor te teclas responsável por controlar o jogador
private Image[]	gigaBotImgs	Arrays de imagens
private Image[]	gLaserImgs	Arrays de imagens
private ImageIcon	iconPause	Imagem do ícone de pausa
private ArrayList<TelaJogo.Inimigo>	inimigos	Uma lista contendo todos os inimigos presentes na tela
private static final long	INTERVALO_TIRO	O intervalo entre cada tiro do jogador (0,3 s)
private Image[]	laserImgs	Arrays de imagens
private TelaJogo.DungeonManager.DungeonLay	layout	O layout da dungeon atual
private Image[]	luvaImgs	Arrays de imagens
private Image[]	malandranhaImgs	Arrays de imagens
private Image[]	morcerangoImgs	Arrays de imagens
private ArrayList<ObjetoColidivel>	objetosColidiveis	Uma lista contendo todos os objetos colidíveis presentes na tela
private Image	paredeImg	Imagens únicas
private ArrayList<Parede>	paredes	Uma lista contendo todas as paredes presentes na tela
private JButton	pauseButton	Botão para pausar o jogo
private Image	pofImage	Imagens únicas
private ArrayList<Pof>	pofs	Uma lista contendo todas as partículas de pof!
private Porta	porta	A porta onde o jogador deve entrar para passar para a próxima dungeon

private Image[]	portaImgs	Arrays de imagens
private Image[]	pratoImgs	Arrays de imagens
private Image[]	queijoBoxerImgs	Arrays de imagens
private Image[]	slimeBotImgs	Arrays de imagens
private Image[]	slimeImgs	Arrays de imagens

Fields inherited from class TelaBase

ALTURA_TELA, efeito, estado, INTERVALO, LARGURA_TELA, musica, NOME_FONTE, save, TAMANHO_BLOCO, timer

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	Description
TelaJogo(MusicPlayer musica)	Construtor da tela de jogo

Method Summary

All Methods | **Instance Methods** | **Concrete Methods**

Modifier and Type	Method	Description
void	actionPerformed(ActionEvent e)	Manipula eventos de ação.
void	carregarImagens()	Carrega as imagens necessárias para a tela de jogo.
private void	carregarProximaDungeon()	Carrega a próxima dungeon
private void	cleanImgArray(Image[] imgs)	Limpa um array de imagem
private void	cleanKeyListeners()	Limpa o leitor de teclas

void	**cleanUp**()	Realiza limpeza de recursos antes da tela ser descartada.
void	**desenharTela**(Graphics g)	Renderiza os elementos visuais da tela de jogo, incluindo todos os objetos colidíveis e partículas.
private void	**gameOver**()	Carrega a tela de Game Over
void	**loadBoss**(int bossNum)	Carrega a tela de batalha contra um boss.
private void	**mostrarPausa**()	Mostra o menu de pausa.
private void	**newSave**(File saveFile)	Cria um novo arquivo de save com o valor padrão (1).
void	**pauseButton**()	Configura o botão de pausa.
void	**readSaveData**()	Lê o arquivo de save para determinar qual mundo deve ser carregado
private void	**resetKeyState**()	Garante que todas as teclas são liberadas
private void	**softClean**()	Método de limpeza parcial, usado individualmente para a transição entre dungeons
void	**start**()	Inicia o jogo
private void	**verificarColisaoParede**(**ObjetoColidivel** entity)	Gerencia a colisão entre paredes e outros objetos colidíveis
private void	**voltarParaMenu**()	Carrega a tela inicial

Methods inherited from class TelaBase

paintComponent, saveData

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, getHeight, getInheritsPopupMenu, getInputMap, getInputMap, getInputVerifier, getInsets, getInsets, getListeners, getLocation, getMaximumSize, getMinimumSize, getNextFocusableComponent, getPopupLocation, getPreferredSize, getRegisteredKeyStrokes, getRootPane, getSize, getToolTipLocation, getToolTipText, getToolTipText, getTopLevelAncestor, getTransferHandler, getVerifyInputWhenFocusTarget, getVetoableChangeListeners, getVisibleRect, getWidth, getX, getY, grabFocus, hide, isDoubleBuffered, isLightweightComponent, isManagingFocus, isOpaque, isOptimizedDrawingEnabled, isPaintingForPrint,

isPaintingOrigin, isPaintingTile, isRequestFocusEnabled, isValidateRoot, paint, paintBorder, paintChildren, 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, getBounds, 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, 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, requestFocus, requestFocus, requestFocusInWindow, resize, resize, setBounds, setBounds, setComponentOrientation, setCursor, setDropTarget, setFocusable, setFocusTraversalKeysEnabled, setIgnoreRepaint, setLocale, setLocation, setLocation, setMixingCutoutShape, 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 Details

dungeonManager

```
private TelaJogo.DungeonManager dungeonManager
```

Variável usada para gerenciar a dungeon atual

layout

```
private TelaJogo.DungeonManager.DungeonLayout layout
```

O layout da dungeon atual

objetosColidiveis

```
private ArrayList<ObjetoColidivel> objetosColidiveis
```

Uma lista contendo todos os objetos colidíveis presentes na tela

pauseButton

```
private JButton pauseButton
```

Botão para pausar o jogo

batata

```
private Player batata
```

O personagem jogável do jogo (Duque Batata)

gameKeyAdapter

```
private TelaJogo.GameKeyAdapter gameKeyAdapter
```

O leitor te teclas responsável por controlar o jogador

cenouras

`private ArrayList<Projetil> cenouras`

Uma lista para gerenciar os projéteis do jogador (as cenouras)

INTERVALO_TIRO

`private static final long INTERVALO_TIRO`

O intervalo entre cada tiro do jogador (0,3 s)

See Also:

Constant Field Values

inimigos

`private ArrayList<TelaJogo.Inimigo> inimigos`

Uma lista contendo todos os inimigos presentes na tela

enemyCount

`private int enemyCount`

Guarda a quantidade de inimigos na tela

paredes

`private ArrayList<Parede> paredes`

Uma lista contendo todas as paredes presentes na tela

porta

`private Porta porta`

A porta onde o jogador deve entrar para passar para a próxima dungeon

alertas

`private ArrayList<Alert> alertas`

Uma lista contendo todas as partículas de alerta presentes na tela

pofs

`private ArrayList<Pof> pofs`

Uma lista contendo todas as partículas de pof! presentes na tela

iconPause

`private ImageIcon iconPause`

Imagem do ícone de pausa

alertImage

`private Image alertImage`

Imagens únicas

pofImage

`private Image pofImage`

Imagens únicas

paredeImg

`private Image paredeImg`

Imagens únicas

algodaoImg

`private Image algodaoImg`

Imagens únicas

backgroundImgs

`private Image[] backgroundImgs`

Arrays de imagens

portaImgs

`private Image[] portaImgs`

Arrays de imagens

batataImgs

`private Image[] batataImgs`

Arrays de imagens

cenouraImgs

`private Image[] cenouraImgs`

Arrays de imagens

slimeImgs

`private Image[] slimeImgs`

Arrays de imagens

flymeImgs

`private Image[] flymeImgs`

Arrays de imagens

pratoImgs

`private Image[] pratoImgs`

Arrays de imagens

facaImgs

`private Image[] facaImgs`

Arrays de imagens

armandibulaImgs

`private Image[] armandibulaImgs`

Arrays de imagens

morcerangoImgs

`private Image[] morcerangoImgs`

Arrays de imagens

queijoBoxerImgs

`private Image[] queijoBoxerImgs`

Arrays de imagens

bracoImgs

`private Image[] bracoImgs`

Arrays de imagens

luvaImgs

`private Image[] luvaImgs`

Arrays de imagens

chocochatoImgs

`private Image[] chocochatoImgs`

Arrays de imagens

algodogDoceImgs

`private Image[] algodogDoceImgs`

Arrays de imagens

slimeBotImgs

`private Image[] slimeBotImgs`

Arrays de imagens

laserImgs

`private Image[] laserImgs`

Arrays de imagens

gigaBotImgs

`private Image[] gigaBotImgs`

Arrays de imagens

gLaserImgs

`private Image[] gLaserImgs`

Arrays de imagens

malandranhaImgs

`private Image[] malandranhaImgs`

Arrays de imagens

alhoImgs

`private Image[] alhoImgs`

Arrays de imagens

Constructor Details

TelaJogo

```
TelaJogo(MusicPlayer musica)
  throws IOException
```

Construtor da tela de jogo

Parameters:

`musica` - Player de música compartilhado entre telas

Throws:

`IOException` - Se ocorrer um erro de I/O durante a leitura do arquivo de save

Method Details

readSaveData

```
public void readSaveData()
                  throws IOException
```

Lê o arquivo de save para determinar qual mundo deve ser carregado

Throws:

`IOException` - Se ocorrer um erro de I/O durante a leitura do arquivo de save

newSave

```
private void newSave(File saveFile)
              throws IOException
```

Cria um novo arquivo de save com o valor padrão (1).

Parameters:

`saveFile` - Arquivo a ser criado

Throws:

`IOException` - Se ocorrer um erro de I/O durante a criação do arquivo

start

```
public void start()
```

Inicia o jogo

Inicia o timer e muda o estado do jogo para RODANDO, carrega o layout inicial do mundo atual, o jogador e a porta.

Overrides:

`start` in class `TelaBase`

pauseButton

```
public void pauseButton()
```

Configura o botão de pausa. Ao ser apertado, o jogo é pausado

mostrarPausa

```
private void mostrarPausa()
```

Mostra o menu de pausa. A partir dele, o jogador pode continuar o jogo ou voltar para a tela inicial

voltarParaMenu

```
private void voltarParaMenu()
```

Carrega a tela inicial

carregarImagens

```
public void carregarImagens()
```

Carrega as imagens necessárias para a tela de jogo.

Specified by:

`carregarImagens` in class `TelaBase`

desenharTela

```
public void desenharTela(Graphics g)
```

Renderiza os elementos visuais da tela de jogo, incluindo todos os objetos colidíveis e partículas.

Specified by:

`desenharTela` in class `TelaBase`

Parameters:

g - Contexto gráfico para renderização

actionPerformed

```
public void actionPerformed(ActionEvent e)
```

Manipula eventos de ação.

Atualiza o estado de todos os objetos colidíveis e as interações entre eles, além das partículas de pof!. Solicita a repintura do componente a cada intervalo definido no timer.

Specified by:

`actionPerformed` in interface `ActionListener`

Overrides:

`actionPerformed` in class `TelaBase`

Parameters:

`e` - Evento de ação disparado

gameOver

```
private void gameOver()
```

Carrega a tela de Game Over

verificarColisaoParede

```
private void verificarColisaoParede(ObjetoColidivel entity)
```

Gerencia a colisão entre paredes e outros objetos colidíveis

Parameters:

`entity` - A entidade colidindo com a parede

carregarProximaDungeon

```
private void carregarProximaDungeon()
```

Carrega a próxima dungeon

loadBoss

```
public void loadBoss(int bossNum)
```

Carrega a tela de batalha contra um boss.

Parameters:

`bossNum` - Número identificador do boss

cleanUp

```
public void cleanUp()
```

Realiza limpeza de recursos antes da tela ser descartada.

Specified by:

`cleanUp` in class `TelaBase`

cleanImgArray

```
private void cleanImgArray(Image[] imgs)
```

Limpa um array de imagem

Parameters:

`imgs` - o array a ser limpo

cleanKeyListeners

```
private void cleanKeyListeners()
```

Limpa o leitor de teclas

resetKeyState

```
private void resetKeyState()
```

Garante que todas as teclas são liberadas

softClean

```
private void softClean()
```

Método de limpeza parcial, usado individualmente para a transição entre dungeons