

[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 WorldController

java.lang.Object  
WorldController

### All Implemented Interfaces:

PhotoController

```
public class WorldController  
extends java.lang.Object  
implements PhotoController
```

### Author:

Leela Prabhu (S1471625)

## Constructor Summary

### Constructors

#### Constructor and Description

**WorldController**(PhotoViewer viewer)

The viewer handles all GUI interactions of the application, all GUI commands from the controller pass to the viewer, it is made accessible to the controller through this function.

## Method Summary

### All Methods    Instance Methods    Concrete Methods

Modifier and Type	Method and Description
void	<b>arrowInit()</b> This initializes arrows that change the displayed image and other elements of the GUI.
void	<b>backgroundShow</b> (int x_coord) This is called whenever the pane or frame is changed, it shows the background image pane given frame.
void	<b>clickF()</b> Handle click front requests from the viewer
void	<b>clickL()</b>

	Handle click left requests from the viewer
void	<b>clickR()</b> Handle click right requests from the viewer
void	<b>forwardArrow()</b> This is called when a new pane or frame is entered.
void	<b>forwardDecision()</b> This is called when the Forward arrow is pressed.
void	<b>frameInit()</b> This function loads images and stores them into Frame instances.
void	<b>getFramesInit()</b> A frame is one entire panorama image at a location.
void	<b>hashtableInit()</b> This function initializes the Hashtable 'putOpt' that facilitates fast access to the menu items.
void	<b>Initialise()</b> Initialize the frames, maps, pickables, etc.
void	<b>mapInit()</b> This is called during intialization.
void	<b>pickableInit()</b> This loads images for Pickable items and creates instances.
void	<b>pickableShow(int x_coord)</b> This determines which pickable items should be shown given the pane and frame.
void	<b>pickUp(javaafx.event.ActionEvent e)</b> Handles pick up requests from the viewer
void	<b>pickUp(java.lang.String switchVar)</b> Handles pick up requests from the test viewer
void	<b>putDown(javaafx.event.ActionEvent e)</b> Handles put down requests from the viewer
void	<b>putDown(java.lang.String switchVar)</b> Handles put down requests from the test viewer
java.lang.String	<b>stringParse(java.lang.String comp)</b> Takes the String of the action event extracts the MenuItem responsible.

## Methods inherited from class java.lang.Object

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

## Constructor Detail

### WorldController

```
public WorldController(PhotoViewer viewer)
```

The viewer handles all GUI interactions of the application, all GUI commands from the controller pass to the viewer, it is made accessible to the controller through this function.

**Parameters:**

viewer - it implements PhotoViewer, it handles all GUI functions.

## Method Detail

### Initialise

```
public void Initialise()
```

**Description copied from interface: PhotoController**

Initialize the frames, maps, pickables, etc.

**Specified by:**

Initialise in interface PhotoController

### getFramesInit

```
public void getFramesInit()
```

A frame is one entire panorama image at a location. A pane is a portion of the frame. This function handles moving from one Frame to another based on the current pane. It is called during initialization.

### hashtableInit

```
public void hashtableInit()
```

This function initializes the Hashtable 'putOpt' that facilitates fast access to the menu items. It is called during initialize.

### frameInit

```
public void frameInit()
```

This function loads images and stores them into Frame instances. Properties of the Frame instances are also set, like image, map, frameNo and offset. This is called during initialization.

#### mapInit

```
public void mapInit()
```

This is called during initialization. This function initializes the map, that is establishes connections between Frames. Note that the positioning of Frames is based on the entry position and not absolute orientation.

#### arrowInit

```
public void arrowInit()
```

This initializes arrows that change the displayed image and other elements of the GUI.

#### pickableInit

```
public void pickableInit()
```

This loads images for Pickable items and creates instances. It also sets some properties like the name and the viewer to which GUI actions must be reported. This is called during initialize.

#### clickF

```
public void clickF()
```

#### Description copied from interface: PhotoController

Handle click front requests from the viewer

#### Specified by:

clickF in interface PhotoController

#### forwardDecision

```
public void forwardDecision()
```

This is called when the Forward arrow is pressed. It handles which Frame should be shown next.

#### forwardArrow

```
public void forwardArrow()
```

This is called when a new pane or frame is entered. It decides whether or not the Forward

arrow should be enabled.

### clickR

```
public void clickR()
```

#### Description copied from interface: PhotoController

Handle click right requests from the viewer

#### Specified by:

clickR in interface PhotoController

### clickL

```
public void clickL()
```

#### Description copied from interface: PhotoController

Handle click left requests from the viewer

#### Specified by:

clickL in interface PhotoController

### backgroundShow

```
public void backgroundShow(int x_coord)
```

This is called whenever the pane or frame is changed, it shows the background image pane given frame.

#### Parameters:

x\_coord - and int that determines which pane of the frame should be shown.

### pickableShow

```
public void pickableShow(int x_coord)
```

This determines which pickable items should be shown given the pane and frame. It not only determines the visibilty of the elements but also the

#### Parameters:

x\_coord -

### stringParse

```
public java.lang.String stringParse(java.lang.String comp)
```

Takes the String of the action event extracts the MenuItem responsible.

**Parameters:**

comp - a String dereived from the action event

**Returns:**

switchVar a String that can be used to access the Hashtable 'putOpt'

**pickUp**

```
public void pickUp(javafx.event.ActionEvent e)
```

**Description copied from interface: PhotoController**

Handles pick up requests from the viewer

**Specified by:**

pickUp in interface PhotoController

**Parameters:**

e - action event for menu selection

**pickUp**

```
public void pickUp(java.lang.String switchVar)
```

**Description copied from interface: PhotoController**

Handles pick up requests from the test viewer

**Specified by:**

pickUp in interface PhotoController

**Parameters:**

switchVar - corresponding to an event action on a menu item

**putDown**

```
public void putDown(javafx.event.ActionEvent e)
```

**Description copied from interface: PhotoController**

Handles put down requests from the viewer

**Specified by:**

putDown in interface PhotoController

**Parameters:**

e - action event for menu selection

**putDown**

```
public void putDown(java.lang.String switchVar)
```

**Description copied from interface: PhotoController**

Handles put down requests from the test viewer

**Specified by:**

putDown in interface PhotoController

**Parameters:**

switchVar - corresponding to an event action on a menu item

[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](#)