Java Awesome Alphabet

Mark Jaleel Mike Vivek Levi

:-)

Package
edu.bu.cs673.AwesomeAlphabet.controller

edu.bu.cs673.AwesomeAlphabet.controller
Class AlphabetPageController

```
java.lang.Object
  |
  +-edu.bu.cs673.AwesomeAlphabet.controller.PageController
      |
      +-edu.bu.cs673.AwesomeAlphabet.controller.AlphabetPageController
```

public class **AlphabetPageController**
extends PageController

This class defines the Alphabet Page Controller.

Field Summary

private	m_alphabet
private	m_view

Fields inherited from class edu.bu.cs673.AwesomeAlphabet.controller.PageController

m_pageObserver

Constructor Summary

public	AlphabetPageController(IPageObserver pageObserver, AlphabetPageView view, Alphabet alphabet) Class constructor.

Method Summary

java.util.Iterator	GetLetterIterator() Gets a Letter iterator from the Alphabet model.
boolean	GoToLetterPage(Letter cLetter) Causes the Letter Page to be shown.
boolean	GoToTitlePage() Causes the Title Page to be shown.
boolean	PlayAlphabetSong() Plays the alphabet song.

Methods inherited from class edu.bu.cs673.AwesomeAlphabet.controller.PageController

GoToPage

Methods inherited from class java.lang.Object

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

Fields

m_view

```
private edu.bu.cs673.AwesomeAlphabet.view.AlphabetPageView m_view
```

m_alphabet

```
private edu.bu.cs673.AwesomeAlphabet.model.Alphabet m_alphabet
```

Constructors

AlphabetPageController

```
public AlphabetPageController(IPageObserver pageObserver,
                              AlphabetPageView view,
                              Alphabet alphabet)
```

Class constructor.

Parameters:

`pageObserver` - A page observer reference so that page transitions may be requested. For example, this may refer to the main window.

`view` - The view.

`alphabet` - The Alphabet model.

Methods

GetLetterIterator

```
public java.util.Iterator GetLetterIterator()
```

Gets a Letter iterator from the Alphabet model.

Returns:

Letter Iterator.

PlayAlphabetSong

```
public boolean PlayAlphabetSong()
```

Plays the alphabet song.

Returns:

True if song was played successfully.

GoToTitlePage

```
public boolean GoToTitlePage()
```

Causes the Title Page to be shown.

Returns:

True if Title Page is able to be shown.

GoToLetterPage

```
public boolean GoToLetterPage(Letter cLetter)
```

Causes the Letter Page to be shown.

Parameters:

`cLetter` - The letter to be shown on the Letter Page.

Returns:

True if Letter Page is able to be shown.

edu.bu.cs673.AwesomeAlphabet.controller
Class ButtonHandler

```
java.lang.Object
  |
  +-edu.bu.cs673.AwesomeAlphabet.controller.ButtonHandler
```

All Implemented Interfaces:
java.awt.event.ActionListener

public class **ButtonHandler**
extends java.lang.Object
implements java.awt.event.ActionListener

This Class handles user actions on the buttons on the GUI

Field Summary

static	`log`
private	`method`
private	`pv`

Constructor Summary

public	`ButtonHandler(PageView pv, java.lang.String method)` Constructor for ButtonHandler.

Method Summary

void	`actionPerformed(java.awt.event.ActionEvent ae)` Method actionPerformed.

Methods inherited from class `java.lang.Object`

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

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

`actionPerformed`

Fields

pv

`private edu.bu.cs673.AwesomeAlphabet.view.PageView **pv**`

method

```
private java.lang.reflect.Method method
```

log

```
static org.apache.log4j.Logger log
```

Constructors

ButtonHandler

```
public ButtonHandler(PageView pv,
                     java.lang.String method)
```

Constructor for ButtonHandler.

Parameters:

`pv` - PageView
`method` - String

Methods

actionPerformed

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

Method actionPerformed.

Parameters:

`ae` - ActionEvent

See Also:

`ActionListener.actionPerformed(java.awt.event.ActionEvent)`

edu.bu.cs673.AwesomeAlphabet.controller
Class LabelClickHandler

```
java.lang.Object
  |
  +-edu.bu.cs673.AwesomeAlphabet.controller.LabelClickHandler
```

All Implemented Interfaces:

java.awt.event.MouseListener

public class **LabelClickHandler**
extends java.lang.Object
implements java.awt.event.MouseListener

Field Summary

static	log
private	method
private	pv

Constructor Summary

public	LabelClickHandler(PageView pv, java.lang.String method) Constructor for LabelClickHandler.

Method Summary

void	mouseClicked(java.awt.event.MouseEvent e) Method mouseClicked.
void	mouseEntered(java.awt.event.MouseEvent e) Method mouseEntered.
void	mouseExited(java.awt.event.MouseEvent e) Method mouseExited.
void	mousePressed(java.awt.event.MouseEvent e) Method mousePressed.
void	mouseReleased(java.awt.event.MouseEvent e) Method mouseReleased.

Methods inherited from class `java.lang.Object`

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

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

`mouseClicked, mouseEntered, mouseExited, mousePressed, mouseReleased`

Fields

pv

```
private edu.bu.cs673.AwesomeAlphabet.view.PageView pv
```

method

```
private java.lang.reflect.Method method
```

log

```
static org.apache.log4j.Logger log
```

Constructors

LabelClickHandler

```
public LabelClickHandler(PageView pv,
                         java.lang.String method)
```

Constructor for LabelClickHandler.

Parameters:

`pv` - PageView
`method` - String

Methods

mouseClicked

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

Method mouseClicked.

Parameters:

`e` - MouseEvent

See Also:

`MouseListener.mouseClicked(java.awt.event.MouseEvent)`

mouseEntered

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

Method mouseEntered.

Parameters:

`e` - MouseEvent

(continued from last page)

See Also:
`MouseListener.mouseEntered(java.awt.event.MouseEvent)`

mouseExited

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

Method mouseExited.

Parameters:
`e` - MouseEvent

See Also:
`MouseListener.mouseExited(java.awt.event.MouseEvent)`

mousePressed

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

Method mousePressed.

Parameters:
`e` - MouseEvent

See Also:
`MouseListener.mousePressed(java.awt.event.MouseEvent)`

mouseReleased

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

Method mouseReleased.

Parameters:
`e` - MouseEvent

See Also:
`MouseListener.mouseReleased(java.awt.event.MouseEvent)`

edu.bu.cs673.AwesomeAlphabet.controller
Class LetterPageController

```
java.lang.Object
  |
  +-edu.bu.cs673.AwesomeAlphabet.controller.PageController
      |
      +-edu.bu.cs673.AwesomeAlphabet.controller.LetterPageController
```

public class **LetterPageController**
extends PageController

This class defines the Letter Page Controller.

Field Summary

private	m_alphabet
private	m_view

Fields inherited from class edu.bu.cs673.AwesomeAlphabet.controller.PageController

m_pageObserver

Constructor Summary

public	LetterPageController(IPageObserver pageObserver, LetterPageView view, Alphabet alphabet) Class constructor.

Method Summary

char	GetLetterAsChar() This method gets the current letter from the model.
void	GetNextExample() This method causes the controller to update the current Letter model so that the next word example is shown.
boolean	GetNextLetter() This method causes the controller to update the Alphabet model so that the next letter is displayed.
java.awt.image.BufferedImage	GetPicture() This method gets the current word picture from the Letter model.
boolean	GetPreviousLetter() This method causes the controller to update the Alphabet model so that the previous letter is displayed.
java.lang.String	GetWord() This method gets the current word from the Letter model.
void	GoToAlphabetPage() Causes the Alphabet Page to be shown.

`void`	`GoToTitlePage()` Causes the Title Page to be shown.
`void`	`LetterClicked()` This method is used to notify the controller that the image of the letter was clicked.
`void`	`ObserveCurrentLetter()`
`void`	`PictureClicked()` This method is used to notify the controller that the word's picture was clicked.
`void`	`WordClicked()` This method is used to notify the controller that the word was clicked.

Methods inherited from class `edu.bu.cs673.AwesomeAlphabet.controller.PageController`

`GoToPage`

Methods inherited from class `java.lang.Object`

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

Fields

m_view

```
private edu.bu.cs673.AwesomeAlphabet.view.LetterPageView m_view
```

m_alphabet

```
private edu.bu.cs673.AwesomeAlphabet.model.Alphabet m_alphabet
```

Constructors

LetterPageController

```
public LetterPageController(IPageObserver pageObserver,
                            LetterPageView view,
                            Alphabet alphabet)
```

Class constructor.

Parameters:

`pageObserver` - A page observer reference so that page transitions may be requested. For example, this may refer to the main window.

`view` - The view.

`alphabet` - The Alphabet model.

Methods

(continued from last page)

LetterClicked

`public void LetterClicked()`

This method is used to notify the controller that the image of the letter was clicked. This will cause a phonetic letter sound to be played.

PictureClicked

`public void PictureClicked()`

This method is used to notify the controller that the word's picture was clicked. This will cause the word sound to be played.

WordClicked

`public void WordClicked()`

This method is used to notify the controller that the word was clicked. This will cause the word sound to be played.

GetNextLetter

`public boolean GetNextLetter()`

This method causes the controller to update the Alphabet model so that the next letter is displayed.

Returns:
True if next letter can be displayed.

GetPreviousLetter

`public boolean GetPreviousLetter()`

This method causes the controller to update the Alphabet model so that the previous letter is displayed.

Returns:
True if previous letter can be displayed.

GoToAlphabetPage

`public void GoToAlphabetPage()`

Causes the Alphabet Page to be shown.

GoToTitlePage

`public void GoToTitlePage()`

Causes the Title Page to be shown.

GetNextExample

`public void GetNextExample()`

This method causes the controller to update the current Letter model so that the next word example is shown.

(continued from last page)

GetLetterAsChar

```
public char GetLetterAsChar()
```

This method gets the current letter from the model.

Returns:
Current letter as a lower-case char.

GetPicture

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

This method gets the current word picture from the Letter model.

Returns:
Word picture as a BufferedImage.

GetWord

```
public java.lang.String GetWord()
```

This method gets the current word from the Letter model.

Returns:
Letter word as a String.

ObserveCurrentLetter

```
public void ObserveCurrentLetter()
```

edu.bu.cs673.AwesomeAlphabet.controller
Class PageController

```
java.lang.Object
  |
  +-edu.bu.cs673.AwesomeAlphabet.controller.PageController
```

Direct Known Subclasses:
AlphabetPageController, LetterPageController, TitlePageController

public abstract class **PageController**
extends java.lang.Object

This class defines the Page Controller. It is the abstract parent class of all page controller classes, including AlphabetPageController, LetterPageController, etc. It manages a Page Observer so that page transitions may be requested.

Field Summary

private	m_pageObserver

Constructor Summary

public	PageController(IPageObserver pageObserver) Class constructor.

Method Summary

boolean	GoToPage(PageName page) Commands a page to be shown.

Methods inherited from class `java.lang.Object`

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

Fields

m_pageObserver

`private edu.bu.cs673.AwesomeAlphabet.model.IPageObserver **m_pageObserver**`

Constructors

PageController

`public **PageController**(IPageObserver pageObserver)`

Class constructor.

(continued from last page)

Parameters:

`pageObserver` - A page observer reference so that page transitions may be requested. For example, this may refer to the main window.

Methods

GoToPage

```
protected boolean GoToPage(PageName page)
```

Commands a page to be shown.

Parameters:

`page` - The name of the page.

Returns:

True if page is able to be shown.

edu.bu.cs673.AwesomeAlphabet.controller
Class TitlePageController

```
java.lang.Object
  |
  +-edu.bu.cs673.AwesomeAlphabet.controller.PageController
      |
      +-edu.bu.cs673.AwesomeAlphabet.controller.TitlePageController
```

public class **TitlePageController**
extends PageController

This class defines the Title Page Controller.

Field Summary

private	`m_view`

Fields inherited from class `edu.bu.cs673.AwesomeAlphabet.controller.PageController`

`m_pageObserver`

Constructor Summary

public	`TitlePageController(IPageObserver pageObserver, TitlePageView view)` Class constructor.

Method Summary

boolean	`Start()` This method is used to notify the controller that the Start button was clicked.

Methods inherited from class `edu.bu.cs673.AwesomeAlphabet.controller.PageController`

`GoToPage`

Methods inherited from class `java.lang.Object`

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

Fields

m_view

```
private edu.bu.cs673.AwesomeAlphabet.view.TitlePageView m_view
```

Constructors

(continued from last page)

TitlePageController

```
public TitlePageController(IPageObserver pageObserver,
                           TitlePageView view)
```

Class constructor.

Parameters:

`pageObserver` - A page observer reference so that page transitions may be requested. For example, this may refer to the main window.
`view` - The view.

Methods

Start

```
public boolean Start()
```

This method is used to notify the controller that the Start button was clicked. This will cause the Alphabet Page to be shown.

Returns:

True if Alphabet Page is able to be shown.

Package edu.bu.cs673.AwesomeAlphabet.main

edu.bu.cs673.AwesomeAlphabet.main
Class AwesomeAlphabetApp

```
java.lang.Object
  |
  +-edu.bu.cs673.AwesomeAlphabet.main.AwesomeAlphabetApp
```

public class **AwesomeAlphabetApp**
extends java.lang.Object

This class contains the application's main() method.

Field Summary

static	log

Constructor Summary

public	AwesomeAlphabetApp()

Method Summary

static void	main(java.lang.String[] args) Main entry point into the application.

Methods inherited from class java.lang.Object

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

Fields

log

`static org.apache.log4j.Logger log`

Constructors

AwesomeAlphabetApp

`public AwesomeAlphabetApp()`

Methods

(continued from last page)

main

```
public static void main(java.lang.String[] args)
```

Main entry point into the application. It is responsible for creating the models, views, controllers, and main window. In addition, it processes the resource file and causes the Title Page to be shown.

Parameters:

`args` - Application arguments.

Throws:

`Exception`

Package edu.bu.cs673.AwesomeAlphabet.model

edu.bu.cs673.AwesomeAlphabet.model
Class Alphabet

```
java.lang.Object
   |
   +-java.util.Observable
      |
      +-edu.bu.cs673.AwesomeAlphabet.model.Alphabet
```

public class **Alphabet**
extends java.util.Observable

The class defines the Alphabet model. It creates and maintains references to 26 Letter objects; one for each letter of the English alphabet. It also maintains the current letter selection and has methods for getting the Letter object, changing the letter selection, and loading resources.

Field Summary

static	`log`
private	`m_alphabetsong`
public	`m_iCurLetterIndex`
private	`m_letters`

Fields inherited from class `java.util.Observable`

`changed`, `obs`

Constructor Summary

public	`Alphabet()` Class constructor.

Method Summary

`Letter`	`GetCurrentLetter()` Returns the selected letter.
`java.util.Iterator`	`GetIterator()` Gets an iterator to the list of Letter objects.
`int`	`GetLetterIndex(char c)` Gets the array index of the specified letter.
`Letter`	`GoToNextLetter()` Changes letter selection to next letter.
`Letter`	`GoToPreviousLetter()` Changes letter selection to previous letter.

`void`	`LoadResources(java.util.Properties prop)` Loads word, picture, and sound resources into Letter objects.
`void`	`PlayAlphabetSong()`
`Letter`	`SetCurrentLetter(Letter letter)` Sets the letter selection.

Methods inherited from class `java.util.Observable`
`addObserver, clearChanged, countObservers, deleteObserver, deleteObservers, hasChanged, notifyObservers, notifyObservers, setChanged`

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

Fields

m_letters

```
private edu.bu.cs673.AwesomeAlphabet.model.Letter m_letters
```

m_iCurLetterIndex

```
public int m_iCurLetterIndex
```

m_alphabetsong

```
private edu.bu.cs673.AwesomeAlphabet.model.GameSound m_alphabetsong
```

log

```
static org.apache.log4j.Logger log
```

Constructors

Alphabet

```
public Alphabet()
```

Class constructor. Responsible for creating the Letter objects.

Methods

(continued from last page)

GetLetterIndex

`private int GetLetterIndex(char c)`

Gets the array index of the specified letter.

Parameters:

`c` - The letter, represented as a char.

Returns:

The index of the letter in the array or -1 if the letter is invalid.

GetIterator

`public java.util.Iterator GetIterator()`

Gets an iterator to the list of Letter objects.

Returns:

An iterator to the list of Letter objects.

SetCurrentLetter

`public Letter SetCurrentLetter(Letter letter)`

Sets the letter selection.

Parameters:

`letter` - The letter to select.

Returns:

A reference to the newly selected Letter object or null if the selection was invalid.

GetCurrentLetter

`public Letter GetCurrentLetter()`

Returns the selected letter.

Returns:

The selected letter.

GoToPreviousLetter

`public Letter GoToPreviousLetter()`

Changes letter selection to previous letter.

Returns:

The previous letter object or null if there are no previous letters.

GoToNextLetter

`public Letter GoToNextLetter()`

Changes letter selection to next letter.

Returns:

(continued from last page)

The next letter object or null if there are no more letters.

LoadResources

```
public void LoadResources(java.util.Properties prop)
```

Loads word, picture, and sound resources into Letter objects.

Parameters:

`prop` - The property list containing resource information.

PlayAlphabetSong

```
public void PlayAlphabetSong()
```

edu.bu.cs673.AwesomeAlphabet.model Class GameImage

```
java.lang.Object
  |
  +-edu.bu.cs673.AwesomeAlphabet.model.GameImage
```

public class **GameImage**
extends java.lang.Object

Field Summary

private static	cl
static	log

Constructor Summary

public	GameImage()

Method Summary

static java.awt.Image	getImage(java.lang.String filename) Constructor.

Methods inherited from class `java.lang.Object`

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

Fields

cl

`private static java.lang.ClassLoader` **`cl`**

log

`static org.apache.log4j.Logger` **`log`**

Constructors

GameImage

`public` **`GameImage`**`()`

(continued from last page)

Methods

getImage

```
public static java.awt.Image getImage(java.lang.String filename)
```

Constructor. This prepends the resource directory to the image's filename.

Parameters:

`filename` - the name of the file containing the graphics

Returns:

an Image containing the graphic read from the jar or the filesystem.

edu.bu.cs673.AwesomeAlphabet.model
Class GameSound

```
java.lang.Object
  |
  +-edu.bu.cs673.AwesomeAlphabet.model.GameSound
```

public class **GameSound**
extends java.lang.Object

Field Summary

private static	cl
static	log
private final	queue
private	soundfilepath

Constructor Summary

public	GameSound(java.lang.String soundfilepath) Constructor.

Method Summary

void	PlaySound() Plays the sound for this object.

Methods inherited from class `java.lang.Object`

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

Fields

log

`static org.apache.log4j.Logger` **`log`**

cl

`private static java.lang.ClassLoader` **`cl`**

soundfilepath

```
private java.lang.String soundfilepath
```

queue

```
private final java.util.concurrent.BlockingQueue queue
```

Constructors

GameSound

```
public GameSound(java.lang.String soundfilepath)
```

Constructor. This prepends the directory to the sound's filename

Parameters:

`soundfilepath` - sound's filename

Methods

PlaySound

```
public void PlaySound()
```

Plays the sound for this object. Does not return until the sound has finished playing. This is limited to short-duration sounds only (2 seconds or less).

edu.bu.cs673.AwesomeAlphabet.model
Interface IPageObserver

All Known Implementing Classes:
MainWindow

public interface **IPageObserver**
extends

This interface defines a Page Observer. It is used to to help manage page transitions without requiring classes to have knowledge of the user interface.

Method Summary

`boolean`	`GoToPage(java.lang.String sPageName)` Commands a page to be shown.

Methods

GoToPage

```
public boolean GoToPage(java.lang.String sPageName)
```

Commands a page to be shown.

Parameters:
`sPageName` - The name of the page.

Returns:
True if page is able to be shown.

edu.bu.cs673.AwesomeAlphabet.model
Class Letter

```
java.lang.Object
  |
  +-java.util.Observable
      |
      +-edu.bu.cs673.AwesomeAlphabet.model.Letter
```

public class **Letter**
extends java.util.Observable

This class defines the Letter model. It represents a letter in the English alphabet and contains a list words that start with the letter, along with associated pictures and sounds. An index is maintained for the word list to keep track of which word was most recently shown.

Field Summary

private	index
private	m_cLetter
private	m_LetterSound
private	m_PhonicSound
private	m_wps

Fields inherited from class `java.util.Observable`

changed, obs

Constructor Summary

public	`Letter(char cLetter)` Class constructor.

Method Summary

void	`addLetterSoundResource(java.lang.String soundName)`
void	`addPhonicSoundResource(java.lang.String soundName)`
void	`addResource(java.lang.String imageName, java.lang.String soundName, java.lang.String wordText)` Adds a new WordPictureSound object to the word list.
javax.swing.Icon	`getIcon(int width, int height)` Gets the image associated with the current word.
char	`GetLetterAsChar()` Gets the lower-case letter that this object represents.

char	`GetUppercaseLetter()` Gets the upper-case letter that this object represents.
java.lang.String	`getWord()` Gets the text of the current word.
WordPictureSound	`getWPSData(int index)` Gets the WordPictureSound object stored at the provided list index.
void	`nextExample()` Advances the word list index and notifies observers that the word example has changed.
void	`playSound()` Plays the sound associated with the current word.
void	`playSoundLetter()`
void	`playSoundPhonic()`

Methods inherited from class `java.util.Observable`

```
addObserver, clearChanged, countObservers, deleteObserver, deleteObservers,
hasChanged, notifyObservers, notifyObservers, setChanged
```

Methods inherited from class `java.lang.Object`

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

Fields

m_cLetter

```
private char m_cLetter
```

m_wps

```
private java.util.List m_wps
```

m_LetterSound

```
private edu.bu.cs673.AwesomeAlphabet.model.GameSound m_LetterSound
```

m_PhonicSound

```
private edu.bu.cs673.AwesomeAlphabet.model.GameSound m_PhonicSound
```

index

```
private int index
```

Constructors

Letter

```
public Letter(char cLetter)
```

Class constructor.

Parameters:
`cLetter` - The letter that this object will represent.

Methods

GetLetterAsChar

```
public char GetLetterAsChar()
```

Gets the lower-case letter that this object represents.

Returns:
The letter as a char.

GetUppercaseLetter

```
public char GetUppercaseLetter()
```

Gets the upper-case letter that this object represents.

Returns:
The letter as a char.

addResource

```
public void addResource(java.lang.String imageName,
         java.lang.String soundName,
         java.lang.String wordText)
```

Adds a new WordPictureSound object to the word list.

Parameters:
`imageName` - The filename of the word image.
`soundName` - The filename of the word sound.
`wordText` - The text of the word.

addLetterSoundResource

```
public void addLetterSoundResource(java.lang.String soundName)
```

(continued from last page)

addPhonicSoundResource

```
public void addPhonicSoundResource(java.lang.String soundName)
```

getWPSData

```
private WordPictureSound getWPSData(int index)
```

Gets the WordPictureSound object stored at the provided list index.

Parameters:

`index` - The index into the list.

Returns:

The WordPictureSound object or null if the index is invalid.

getWord

```
public java.lang.String getWord()
```

Gets the text of the current word.

Returns:

The text of the current word as a String.

getIcon

```
public javax.swing.Icon getIcon(int width,
         int height)
```

Gets the image associated with the current word.

Returns:

The image associated with the current word.

nextExample

```
public void nextExample()
```

Advances the word list index and notifies observers that the word example has changed.

playSound

```
public void playSound()
```

Plays the sound associated with the current word.

playSoundLetter

```
public void playSoundLetter()
```

playSoundPhonic

```
public void playSoundPhonic()
```

(continued from last page)

edu.bu.cs673.AwesomeAlphabet.model
Class MainWindow

```
java.lang.Object
   |
   +-edu.bu.cs673.AwesomeAlphabet.model.MainWindow
```

All Implemented Interfaces:

IPageObserver

public class **MainWindow**
extends java.lang.Object
implements IPageObserver

Field Summary

private	m_cl
private	m_curView
private	m_frame
private	m_pageHash

Constructor Summary

public	MainWindow()

Method Summary

boolean	GoToPage(java.lang.String sPageName)
void	registerPage(PageView page)
void	Show()

Methods inherited from class `java.lang.Object`

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

Methods inherited from interface `edu.bu.cs673.AwesomeAlphabet.model.IPageObserver`

GoToPage

Fields

(continued from last page)

m_frame

```
private javax.swing.JFrame m_frame
```

m_curView

```
private javax.swing.JPanel m_curView
```

m_cl

```
private java.awt.CardLayout m_cl
```

m_pageHash

```
private java.util.Hashtable m_pageHash
```

Constructors

MainWindow

```
public MainWindow()
```

Methods

registerPage

```
public void registerPage(PageView page)
```

Show

```
public void Show()
```

GoToPage

```
public boolean GoToPage(java.lang.String sPageName)
```

edu.bu.cs673.AwesomeAlphabet.model
Class PageName

```
java.lang.Object
  |
  +-java.lang.Enum
      |
      +-edu.bu.cs673.AwesomeAlphabet.model.PageName
```

All Implemented Interfaces:
java.io.Serializable, java.lang.Comparable

---

public final class **PageName**
extends java.lang.Enum

This enumeration defines the page names for each view. To get the name as a String, use the toString() method.

Field Summary

public static final	AlphabetPage
public static final	LetterPage
private final	stringValue
public static final	TitlePage

Fields inherited from class `java.lang.Enum`

name, ordinal

Constructor Summary

private	PageName(java.lang.String s)

Method Summary

java.lang.String	toString()
static PageName	valueOf(java.lang.String name)
static PageName[]	values()

Methods inherited from class `java.lang.Enum`

clone, compareTo, equals, finalize, getDeclaringClass, hashCode, name, ordinal, readObject, readObjectNoData, toString, valueOf

Methods inherited from class `java.lang.Object`

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

Methods inherited from interface `java.lang.Comparable`

```
compareTo
```

Fields

TitlePage

`public static final edu.bu.cs673.AwesomeAlphabet.model.PageName` **`TitlePage`**

AlphabetPage

`public static final edu.bu.cs673.AwesomeAlphabet.model.PageName` **`AlphabetPage`**

LetterPage

`public static final edu.bu.cs673.AwesomeAlphabet.model.PageName` **`LetterPage`**

stringValue

`private final java.lang.String` **`stringValue`**

Constructors

PageName

`private` **`PageName`**`(java.lang.String s)`

Methods

values

`public static PageName[]` **`values`**`()`

valueOf

`public static PageName` **`valueOf`**`(java.lang.String name)`

toString

public java.lang.String **toString**()

edu.bu.cs673.AwesomeAlphabet.model
Class Version

```
java.lang.Object
  |
  +-edu.bu.cs673.AwesomeAlphabet.model.Version
```

public class **Version**
extends java.lang.Object

This class contains the application version number and related information.

Field Summary

public static final	sAppName Value: **Awesome Alphabet**
public static final	sVersionNum Value: **0.0.0.0**

Constructor Summary

public	Version()

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

Fields

sAppName

public static final java.lang.String **sAppName**

Constant value: **Awesome Alphabet**

sVersionNum

public static final java.lang.String **sVersionNum**

Constant value: **0.0.0.0**

Constructors

Version

public **Version**()

(continued from last page)

edu.bu.cs673.AwesomeAlphabet.model
Class WordPictureSound

```
java.lang.Object
   |
   +-edu.bu.cs673.AwesomeAlphabet.model.WordPictureSound
```

public class **WordPictureSound**
extends java.lang.Object

Field Summary

private	m_image
private	m_sound
private	m_word

Constructor Summary

public	WordPictureSound(java.lang.String word, java.lang.String imageFile, java.lang.String soundFile)

Method Summary

javax.swing.ImageIcon	GetWordImage(int width, int height)
java.lang.String	GetWordString()
void	PlaySound()

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

Fields

m_word

`private java.lang.String **m_word**`

m_sound

`private edu.bu.cs673.AwesomeAlphabet.model.GameSound **m_sound**`

(continued from last page)

m_image

```
private java.awt.Image m_image
```

Constructors

WordPictureSound

```
public WordPictureSound(java.lang.String word,
                        java.lang.String imageFile,
                        java.lang.String soundFile)
```

Methods

PlaySound

```
public void PlaySound()
```

GetWordString

```
public java.lang.String GetWordString()
```

GetWordImage

```
public javax.swing.ImageIcon GetWordImage(int width,
         int height)
```

Package edu.bu.cs673.AwesomeAlphabet.view

edu.bu.cs673.AwesomeAlphabet.view
Class AAJPanel

```
java.lang.Object
  |
  +-java.awt.Component
      |
      +-java.awt.Container
          |
          +-javax.swing.JComponent
              |
              +-javax.swing.JPanel
                  |
                  +-edu.bu.cs673.AwesomeAlphabet.view.AAJPanel
```

All Implemented Interfaces:
java.io.Serializable, java.awt.MenuContainer, java.awt.image.ImageObserver, javax.swing.TransferHandler.HasGetTransferHandler, java.io.Serializable, javax.accessibility.Accessible

public class **AAJPanel**
extends javax.swing.JPanel

This class defines a customized JPanel. It has the ability to add a background image.

Field Summary

	m_backgroundImage
	m_originalBackgroundImage

Fields inherited from class `javax.swing.JPanel`

uiClassID

Fields inherited from class `javax.swing.JComponent`

aaTextInfo, accessibleContext, actionMap, ACTIONMAP_CREATED, alignmentX, alignmentY, ANCESTOR_INPUTMAP_CREATED, ANCESTOR_NOTIFIER_KEY, ANCESTOR_USING_BUFFER, ancestorInputMap, autoscrolls, AUTOSCROLLS_SET, border, clientProperties, COMPLETELY_OBSCURED, componentObtainingGraphicsFrom, componentObtainingGraphicsFromLock, CREATED_DOUBLE_BUFFER, DEBUG_GRAPHICS_LOADED, defaultLocale, flags, FOCUS_INPUTMAP_CREATED, FOCUS_TRAVERSAL_KEYS_BACKWARD_SET, FOCUS_TRAVERSAL_KEYS_FORWARD_SET, focusController, focusInputMap, INHERITS_POPUP_MENU, INPUT_VERIFIER_KEY, INPUT_VERIFIER_SOURCE_KEY, inputVerifier, IS_DOUBLE_BUFFERED, IS_OPAQUE, IS_PAINTING_TILE, IS_PRINTING, IS_PRINTING_ALL, IS_REPAINTING, isAlignmentXSet, isAlignmentYSet, KEY_EVENTS_ENABLED, KEYBOARD_BINDINGS_KEY, listenerList, managingFocusBackwardTraversalKeys, managingFocusForwardTraversalKeys, NEXT_FOCUS, NOT_OBSCURED, OPAQUE_SET, paintingChild, PARTIALLY_OBSCURED, popupMenu, readObjectCallbacks, REQUEST_FOCUS_DISABLED, RESERVED_1, RESERVED_2, RESERVED_3, RESERVED_4, RESERVED_5, RESERVED_6, tempRectangles, TOOL_TIP_TEXT_KEY, TRANSFER_HANDLER_KEY, ui, uiClassID, UNDEFINED_CONDITION, verifyInputWhenFocusTarget, vetoableChangeSupport, WHEN_ANCESTOR_OF_FOCUSED_COMPONENT, WHEN_FOCUSED, WHEN_IN_FOCUSED_WINDOW, WHEN_IN_FOCUSED_WINDOW_BINDINGS, WIF_INPUTMAP_CREATED, windowInputMap, WRITE_OBJ_COUNTER_FIRST, WRITE_OBJ_COUNTER_LAST

Fields inherited from class `java.awt.Container` component, containerListener, containerSerializedDataVersion, dbg, descendantsCount, dispatcher, EMPTY_ARRAY, focusCycleRoot, focusTraversalPolicy, focusTraversalPolicyProvider, INCLUDE_SELF, layoutMgr, listeningBoundsChildren, listeningChildren, mixingLog, modalAppContext, modalComp, numOfHWComponents, numOfLWComponents, preserveBackgroundColor, printing, printingThreads, SEARCH_HEAVYWEIGHTS, serialPersistentFields, serialVersionUID

Fields inherited from class `java.awt.Component`

acc, accessibleContext, actionListenerK, adjustmentListenerK, appContext, background, backgroundEraseDisabled, BOTTOM_ALIGNMENT, boundsOp, bufferStrategy, CENTER_ALIGNMENT, changeSupport, coalesceEventsParams, coalesceMap, coalescingEnabled, componentListener, componentListenerK, componentOrientation, componentSerializedDataVersion, compoundShape, containerListenerK, cursor, dbg, dropTarget, enabled, eventCache, eventMask, FOCUS_TRAVERSABLE_DEFAULT, FOCUS_TRAVERSABLE_SET, FOCUS_TRAVERSABLE_UNKNOWN, focusable, focusListener, focusListenerK, focusLog, focusTraversalKeyPropertyNames, focusTraversalKeys, focusTraversalKeysEnabled, font, foreground, graphicsConfig, height, hierarchyBoundsListener, hierarchyBoundsListenerK, hierarchyListener, hierarchyListenerK, ignoreRepaint, incRate, inputMethodListener, inputMethodListenerK, isAddNotifyComplete, isFocusTraversableOverridden, isInc, isPacked, itemListenerK, keyListener, keyListenerK, LEFT_ALIGNMENT, locale, LOCK, log, maxSize, maxSizeSet, minSize, minSizeSet, mixingCutoutRegion, mixingLog, mouseListener, mouseListenerK, mouseMotionListener, mouseMotionListenerK, mouseWheelListener, mouseWheelListenerK, name, nameExplicitlySet, nativeInLightFixer, newEventsOnly, objectLock, ownedWindowK, parent, peer, peerFont, popups, prefSize, prefSizeSet, privateKey, requestFocusController, RIGHT_ALIGNMENT, serialVersionUID, textListenerK, TOP_ALIGNMENT, valid, visible, width, windowClosingException, windowFocusListenerK, windowListenerK, windowStateListenerK, x, y

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

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

Constructor Summary

`public`	`AAJPanel()` Class constructor.

Method Summary

`void`	`paint(java.awt.Graphics g)` Overridden paint method.
`void`	`SetBackgroundImage(java.awt.Image image)` Sets the background image.
`void`	`SetBackgroundImage(java.lang.String filename)` Sets the background image.

Methods inherited from class `javax.swing.JPanel`

getAccessibleContext, getUI, getUIClassID, paramString, setUI, updateUI, writeObject

Methods inherited from class `javax.swing.JComponent` `_paintImmediately, addAncestorListener, addNotify, addVetoableChangeListener, adjustPaintFlags, alwaysOnTop, checkIfChildObscuredBySibling, clientPropertyChanged, componentInputMapChanged, computeVisibleRect, computeVisibleRect, compWriteObjectNotify, contains, createToolTip, deregisterNextFocusableComponent, disable, dndDone, dropLocationForPoint, enable, fetchRectangle, firePropertyChange, firePropertyChange, firePropertyChange, fireVetoableChange, getAccessibleContext, getActionForKeyStroke, getActionMap, getActionMap, getAlignmentX, getAlignmentY, getAncestorListeners, getAncestorNotifier, getAutoscrolls, getBaseline, getBaselineResizeBehavior, getBorder, getBounds, getClientProperties, getClientProperty, getComponentGraphics, getComponentPopupMenu, getConditionForKeyStroke, getCreatedDoubleBuffer, getDebugGraphicsOptions, getDefaultLocale, getFlag, getFontMetrics, getGraphics, getGraphicsInvoked, getHeight, getInheritsPopupMenu, getInputMap, getInputMap, getInputMap, getInputVerifier, getInsets, getInsets, getListeners, getLocation, getManagingFocusBackwardTraversalKeys, getManagingFocusForwardTraversalKeys, getMaximumSize, getMinimumSize, getNextFocusableComponent, getObscuredState, getPopupLocation, getPreferredSize, getRegisteredKeyStrokes, getRootPane, getSize, getToolTipLocation, getToolTipText, getToolTipText, getTopLevelAncestor, getTransferHandler, getUIClassID, getVerifyInputWhenFocusTarget, getVetoableChangeListeners, getVisibleRect, getWidth, getWriteObjCounter, getX, getY, grabFocus, isComponentObtainingGraphicsFrom, isDoubleBuffered, isLightweightComponent, isManagingFocus, isOpaque, isOptimizedDrawingEnabled, isPainting, isPaintingForPrint, isPaintingOrigin, isPaintingTile, isRequestFocusEnabled, isValidateRoot, paint, paintBorder, paintChildren, paintComponent, paintForceDoubleBuffered, paintImmediately, paintImmediately, paintToOffscreen, paramString, print, printAll, printBorder, printChildren, printComponent, processComponentKeyEvent, processKeyBinding, processKeyBindings, processKeyBindingsForAllComponents, processKeyEvent, processMouseEvent, processMouseMotionEvent, putClientProperty, readObject, rectangleIsObscured, recycleRectangle, registerKeyboardAction, registerKeyboardAction, registerNextFocusableComponent, registerNextFocusableComponent, registerWithKeyboardManager, registerWithKeyboardManager, removeAncestorListener, removeNotify, removeVetoableChangeListener, repaint, repaint, requestDefaultFocus, requestFocus, requestFocus, requestFocusInWindow, requestFocusInWindow, resetKeyboardActions, reshape, revalidate, safelyGetGraphics, safelyGetGraphics, scrollRectToVisible, setActionMap, setAlignmentX, setAlignmentY, setAutoscrolls, setBackground, setBorder, setComponentPopupMenu, setCreatedDoubleBuffer, setDebugGraphicsOptions, setDefaultLocale, setDoubleBuffered, setDropLocation, setEnabled, setFlag, setFocusTraversalKeys, setFont, setForeground, setInheritsPopupMenu, setInputMap, setInputVerifier, setMaximumSize, setMinimumSize, setNextFocusableComponent, setOpaque, setPaintingChild, setPreferredSize, setRequestFocusEnabled, setToolTipText, setTransferHandler, setUI, setUIProperty, setVerifyInputWhenFocusTarget, setVisible, setWriteObjCounter, shouldDebugGraphics, superProcessMouseMotionEvent, unregisterKeyboardAction, unregisterWithKeyboardManager, unregisterWithKeyboardManager, update, updateUI, writeObject`

Methods inherited from class `java.awt.Container` add, add, add, add, add, addContainerListener, addDelicately, addImpl, addNotify, addPropertyChangeListener, addPropertyChangeListener, adjustDecendantsOnParent, adjustDescendants, adjustListeningChildren, applyComponentOrientation, areFocusTraversalKeysSet, canContainFocusOwner, checkAdding, checkAddToSelf, checkGD, checkNotAWindow, clearCurrentFocusCycleRootOnHide, clearMostRecentFocusOwnerOnHide, containsFocus, countComponents, countHierarchyMembers, createChildHierarchyEvents, createHierarchyEvents, decreaseComponentCount, deliverEvent, dispatchEventImpl, dispatchEventToSelf, doLayout, eventEnabled, findComponentAt, findComponentAt, findComponentAt, findComponentAtImpl, findTraversalRoot, getAccessibleAt, getAccessibleChild, getAccessibleChildrenCount, getAlignmentX, getAlignmentY, getBottommostComponentIndex, getComponent, getComponentAt, getComponentAt, getComponentCount, getComponents_NoClientCode, getComponents, getComponentZOrder, getContainerListeners, getDropTargetEventTarget, getFocusTraversalKeys, getFocusTraversalPolicy, getHeavyweightContainer, getInsets, getLayout, getListeners, getListenersCount, getMaximumSize, getMinimumSize, getMouseEventTarget, getMouseEventTarget, getMouseEventTargetImpl, getMousePosition, getOpaqueShape, getPreferredSize, getTopmostComponentIndex, hasHeavyweightDescendants, hasLightweightDescendants, increaseComponentCount, initializeFocusTraversalKeys, initIDs, insets, invalidate, invalidateTree, isAncestorOf, isFocusCycleRoot, isFocusCycleRoot, isFocusTraversalPolicyProvider, isFocusTraversalPolicySet, isParentOf, isRemoveNotifyNeeded, isSameOrAncestorOf, layout, lightweightPaint, lightweightPrint, list, list, locate, minimumSize, mixOnReshaping, mixOnShowing, mixOnValidating, mixOnZOrderChanging, nextFocusHelper, numListening, paint, paintComponents, paintHeavyweightComponents, paramString, postProcessKeyEvent, postsOldMouseEvents, preferredSize, preProcessKeyEvent, print, printComponents, printHeavyweightComponents, processContainerEvent, processEvent, proxyEnableEvents, readObject, recursiveApplyCurrentShape, recursiveApplyCurrentShape, recursiveApplyCurrentShape, recursiveSubtractAndApplyShape, recursiveSubtractAndApplyShape, recursiveSubtractAndApplyShape, remove, remove, removeAll, removeContainerListener, removeDelicately, removeNotify, reparentChild, reparentTraverse, setComponentZOrder, setFocusCycleRoot, setFocusTraversalKeys, setFocusTraversalPolicy, setFocusTraversalPolicyProvider, setFont, setLayout, startLWModal, stopLWModal, transferFocusBackward, transferFocusDownCycle, update, validate, validateTree, writeObject

Methods inherited from class java.awt.Component

```
action, add, addComponentListener, addFocusListener, addHierarchyBoundsListener,
addHierarchyListener, addInputMethodListener, addKeyListener, addMouseListener,
addMouseMotionListener, addMouseWheelListener, addNotify, addPropertyChangeListener,
addPropertyChangeListener, adjustListeningChildrenOnParent,
applyComponentOrientation, applyCompoundShape, applyCurrentShape,
applyCurrentShapeBelowMe, areBoundsValid, areFocusTraversalKeysSet,
areInputMethodsEnabled, autoProcessMouseWheel, autoTransferFocus, bounds,
calculateCurrentShape, canBeFocusOwner, checkCoalescing, checkGD, checkImage,
checkImage, checkTreeLock, checkWindowClosingException,
clearCurrentFocusCycleRootOnHide, clearMostRecentFocusOwnerOnHide, coalesceEvents,
constructComponentName, contains, contains, containsFocus, countHierarchyMembers,
createBufferStrategy, createBufferStrategy, createHierarchyEvents, createImage,
createImage, createVolatileImage, createVolatileImage, deliverEvent, disable,
disableEvents, dispatchEvent, dispatchEventImpl, dispatchMouseWheelToAncestor,
doAutoTransfer, doLayout, doSwingSerialization, enable, enable, enableEvents,
enableInputMethods, eventEnabled, eventTypeEnabled, findUnderMouseInWindow,
firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange,
firePropertyChange, firePropertyChange, firePropertyChange, firePropertyChange,
firePropertyChange, getAccessControlContext, getAccessibleContext,
getAccessibleIndexInParent, getAccessibleStateSet, getAlignmentX, getAlignmentY,
getAppliedShape, getBackBuffer, getBackground, getBaseline,
getBaselineResizeBehavior, getBounds, getBounds, getBoundsOp, getBufferStrategy,
getColorModel, getComponentAt, getComponentAt, getComponentListeners,
getComponentOrientation, getContainer, getContainingWindow, getContainingWindow,
getCursor, getDropTarget, getFocusCycleRootAncestor, getFocusListeners,
getFocusTraversalKeys_NoIDCheck, getFocusTraversalKeys, getFocusTraversalKeysEnabled,
getFont_NoClientCode, getFont, getFontMetrics, getForeground,
getGraphics_NoClientCode, getGraphics, getGraphicsConfiguration_NoClientCode,
getGraphicsConfiguration, getHeight, getHierarchyBoundsListeners,
getHierarchyListeners, getIgnoreRepaint, getInputContext, getInputMethodListeners,
getInputMethodRequests, getInsets_NoClientCode, getKeyListeners, getListeners,
getLocale, getLocation, getLocation, getLocationOnScreen_NoTreeLock,
getLocationOnScreen, getLocationOnWindow, getMaximumSize, getMinimumSize,
getMouseListeners, getMouseMotionListeners, getMousePosition, getMouseWheelListeners,
getName, getNativeContainer, getNormalShape, getObjectLock, getOpaqueShape,
getParent_NoClientCode, getParent, getPeer, getPreferredSize,
getPropertyChangeListeners, getPropertyChangeListeners, getSiblingIndexAbove,
getSiblingIndexBelow, getSize, getSize, getToolkit, getToolkitImpl, getTreeLock,
getWidth, getX, getY, gotFocus, handleEvent, hasFocus, hide, imageUpdate,
initializeFocusTraversalKeys, initIDs, inside, invalidate, invalidateIfValid,
isBackgroundSet, isCoalesceEventsOverriden, isCoalescingEnabled, isCursorSet,
isDisplayable, isDoubleBuffered, isEnabled, isEnabledImpl, isFocusable,
isFocusCycleRoot, isFocusOwner, isFocusTraversable, isFocusTraversableOverridden,
isFontSet, isForegroundSet, isInstanceOf, isLightweight, isMaximumSizeSet,
isMinimumSizeSet, isMixingNeeded, isNonOpaqueForMixing, isOpaque, isPreferredSizeSet,
isRecursivelyVisible, isRequestFocusAccepted, isSameOrAncestorOf, isShowing, isValid,
isVisible_NoClientCode, isVisible, keyDown, keyUp, layout, lightweightPaint,
lightweightPrint, list, list, list, list, list, locate, location, lostFocus,
minimumSize, mixOnHiding, mixOnReshaping, mixOnShowing, mixOnValidating,
mixOnZOrderChanging, mouseDown, mouseDrag, mouseEnter, mouseExit, mouseMove, mouseUp,
move, nextFocus, nextFocusHelper, notifyNewBounds, numListening, paint, paintAll,
paintHeavyweightComponents, paramString, pointRelativeToComponent, postEvent,
postNextFocusHelper, postsOldMouseEvents, preferredSize, preNextFocusHelper,
prepareImage, prepareImage, print, printAll, printHeavyweightComponents,
processComponentEvent, processEvent, processFocusEvent, processHierarchyBoundsEvent,
processHierarchyEvent, processInputMethodEvent, processKeyEvent, processMouseEvent,
processMouseMotionEvent, processMouseWheelEvent, readObject, remove,
removeComponentListener, removeFocusListener, removeHierarchyBoundsListener,
removeHierarchyListener, removeInputMethodListener, removeKeyListener,
removeMouseListener, removeMouseMotionListener, removeMouseWheelListener,
removeNotify, removePropertyChangeListener, removePropertyChangeListener, repaint,
repaint, repaint, repaint, repaintParentIfNeeded, requestFocus, requestFocus,
requestFocus, requestFocus, requestFocusHelper, requestFocusHelper,
requestFocusInWindow, requestFocusInWindow, requestFocusInWindow,
requestFocusInWindow, resetGC, reshape, reshapeNativePeer, resize, resize,
setBackground, setBounds, setBounds, setBoundsOp, setComponentOrientation, setCursor,
setDropTarget, setEnabled, setFocusable, setFocusTraversalKeys_NoIDCheck,
setFocusTraversalKeys, setFocusTraversalKeysEnabled, setFont, setForeground,
setGCFromPeer, setIgnoreRepaint, setLocale, setLocation, setLocation, setMaximumSize,
setMinimumSize, setName, setPreferredSize, setRequestFocusController, setSize,
setSize, setVisible, show, show, size, subtractAndApplyShape,
subtractAndApplyShapeBelowMe, toString, transferFocus, transferFocusBackward,
transferFocusUpCycle, update, updateCursorImmediately, validate, writeObject
```

Methods inherited from class `java.lang.Object`

```
clone, equals, finalize, getClass, hashCode, notify, notifyAll, registerNatives,
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.swing.TransferHandler.HasGetTransferHandler`

```
getTransferHandler
```

Methods inherited from interface `javax.accessibility.Accessible`

```
getAccessibleContext
```

Fields

m_originalBackgroundImage

```
java.awt.Image m_originalBackgroundImage
```

m_backgroundImage

```
java.awt.Image m_backgroundImage
```

Constructors

(continued from last page)

AAJPanel

`public **AAJPanel**()`

Class constructor.

Methods

SetBackgroundImage

`public void **SetBackgroundImage**(java.lang.String filename)`

Sets the background image.

Parameters:
`filename` - The filename of the image.

SetBackgroundImage

`public void **SetBackgroundImage**(java.awt.Image image)`

Sets the background image.

Parameters:
`image` - The image.

paint

`public void **paint**(java.awt.Graphics g)`

Overridden paint method. Paints the background image.

edu.bu.cs673.AwesomeAlphabet.view
Class AlphabetPageView

```
java.lang.Object
  |
  +-edu.bu.cs673.AwesomeAlphabet.view.PageView
      |
      +-edu.bu.cs673.AwesomeAlphabet.view.AlphabetPageView
```

All Implemented Interfaces:
java.util.Observer

public class **AlphabetPageView**
extends PageView

This class defines the Alphabet Page View. From this view, the user can play the alphabet song, go to the Title Page, or click on a letter button to go to a specific Letter Page.

Field Summary

private	letterPanel
private	m_controller

Fields inherited from class edu.bu.cs673.AwesomeAlphabet.view.PageView

border, letterFont, m_panel, m_sPageName, NAV_BUTTON_ALPHABET_PAGE, NAV_BUTTON_ALPHABET_SONG, NAV_BUTTON_HOME, NAV_BUTTON_NEXT_EXAMPLE, NAV_BUTTON_NEXT_LETTER, NAV_BUTTON_PREV_LETTER, NAV_BUTTON_START, NAV_BUTTON_TITLE_PAGE, wordFont

Constructor Summary

public	AlphabetPageView(java.lang.String sPageName) Class constructor.

Method Summary

void	activated()
void	OnLetterButtonClick(Letter cLetter) Called when a letter button is clicked and causes the Letter Page to be shown.
void	OnPlayAlphabetSongButtonClick() Called when the Alphabet Song button is clicked and causes the alphabet song to be played.
void	OnTitlePageButtonClick() Called when the Title Page button is clicked and causes the Title Page to be shown.
void	SetController(AlphabetPageController controller) Sets the controller associated with this view and creates the letter buttons.

void	`update(java.util.Observable o, java.lang.Object arg)` Causes the view to be updated when the model changes.

Methods inherited from class `edu.bu.cs673.AwesomeAlphabet.view.PageView`
`activated, getButtonImage, getPageName, getPagePanel`

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

Methods inherited from interface `java.util.Observer`
`update`

Fields

m_controller

```
private edu.bu.cs673.AwesomeAlphabet.controller.AlphabetPageController m_controller
```

letterPanel

```
private javax.swing.JPanel letterPanel
```

Constructors

AlphabetPageView

```
public AlphabetPageView(java.lang.String sPageName)
```

Class constructor.

Parameters:
`sPageName` - The page name associated with this view.

Methods

SetController

```
public void SetController(AlphabetPageController controller)
```

Sets the controller associated with this view and creates the letter buttons.

Parameters:
`controller` - The controller.

(continued from last page)

update

```
public void update(java.util.Observable o,
         java.lang.Object arg)
```

Causes the view to be updated when the model changes. Currently this method does nothing. It is required for the observer pattern.

OnLetterButtonClick

```
public void OnLetterButtonClick(Letter cLetter)
```

Called when a letter button is clicked and causes the Letter Page to be shown.

Parameters:

`cLetter` - The letter to show.

OnTitlePageButtonClick

```
public void OnTitlePageButtonClick()
```

Called when the Title Page button is clicked and causes the Title Page to be shown.

OnPlayAlphabetSongButtonClick

```
public void OnPlayAlphabetSongButtonClick()
```

Called when the Alphabet Song button is clicked and causes the alphabet song to be played.

activated

```
public void activated()
```

edu.bu.cs673.AwesomeAlphabet.view
Class LetterPageView

```
java.lang.Object
   |
   +-edu.bu.cs673.AwesomeAlphabet.view.PageView
      |
      +-edu.bu.cs673.AwesomeAlphabet.view.LetterPageView
```

All Implemented Interfaces:
java.util.Observer

public class **LetterPageView**
extends PageView

This class defines the Letter Page View. From this view, the user is able to see the current letter; see words and pictures associated with the letter; and play letter and word sounds. The user is also able to go to the Title and Alphabet pages and see the previous and next letters.

Field Summary

	`m_controller`
	`m_currentLetter`
	`m_image`
	`m_lowercase`
	`m_uppercase`
	`m_word`

Fields inherited from class `edu.bu.cs673.AwesomeAlphabet.view.PageView`

`border`, `letterFont`, `m_panel`, `m_sPageName`, `NAV_BUTTON_ALPHABET_PAGE`, `NAV_BUTTON_ALPHABET_SONG`, `NAV_BUTTON_HOME`, `NAV_BUTTON_NEXT_EXAMPLE`, `NAV_BUTTON_NEXT_LETTER`, `NAV_BUTTON_PREV_LETTER`, `NAV_BUTTON_START`, `NAV_BUTTON_TITLE_PAGE`, `wordFont`

Constructor Summary

`public`	`LetterPageView(java.lang.String sPageName)` Class constructor.

Method Summary

`void`	`activated()` Lets the controller know that this view has become active.
`void`	`OnAlphabetPageButtonClick()` Called when the Alphabet Page button is clicked and causes the Alphabet Page to be shown.

void	OnGetNextExampleButtonClick() Called when the Get Next Example button is clicked and causes the next word and picture to be shown for the current letter.
void	OnLetterClick() Called when the letter text/image is clicked and causes the controller to play a phonetic letter sound.
void	OnNextLetterButtonClick() Called when the Next Letter button is clicked and causes the next letter to be displayed (if available).
void	OnPictureClick() Called when the word's picture is clicked and causes the controller to play the word's sound.
void	OnPreviousLetterButtonClick() Called when the Previous Letter button is clicked and causes the previous letter to be displayed (if available).
void	OnTitlePageButtonClick() Called when the Title Page button is clicked and causes the Title Page to be shown.
void	OnWordClick() Called when the word is clicked and causes the controller to play the word's sound.
void	SetController(LetterPageController controller) Sets the controller associated with this view.
void	update(java.util.Observable o, java.lang.Object arg) Causes the view to be updated when the model changes.

Methods inherited from class edu.bu.cs673.AwesomeAlphabet.view.PageView

activated, getButtonImage, getPageName, getPagePanel

Methods inherited from class java.lang.Object

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

Methods inherited from interface java.util.Observer

update

Fields

m_controller

`edu.bu.cs673.AwesomeAlphabet.controller.LetterPageController` **`m_controller`**

m_uppercase

`javax.swing.JButton` **`m_uppercase`**

m_lowercase

```
javax.swing.JButton m_lowercase
```

m_image

```
javax.swing.JButton m_image
```

m_word

```
javax.swing.JLabel m_word
```

m_currentLetter

```
edu.bu.cs673.AwesomeAlphabet.model.Letter m_currentLetter
```

Constructors

LetterPageView

```
public LetterPageView(java.lang.String sPageName)
```

Class constructor.

Parameters:

`sPageName` - The page name associated with this view.

Methods

SetController

```
public void SetController(LetterPageController controller)
```

Sets the controller associated with this view.

Parameters:

`controller` - The controller.

update

```
public void update(java.util.Observable o,
         java.lang.Object arg)
```

Causes the view to be updated when the model changes. It is required for the observer pattern.

OnLetterClick

```
public void OnLetterClick()
```

(continued from last page)

Called when the letter text/image is clicked and causes the controller to play a phonetic letter sound.

OnPictureClick

```
public void OnPictureClick()
```

Called when the word's picture is clicked and causes the controller to play the word's sound.

OnWordClick

```
public void OnWordClick()
```

Called when the word is clicked and causes the controller to play the word's sound.

OnNextLetterButtonClick

```
public void OnNextLetterButtonClick()
```

Called when the Next Letter button is clicked and causes the next letter to be displayed (if available).

OnPreviousLetterButtonClick

```
public void OnPreviousLetterButtonClick()
```

Called when the Previous Letter button is clicked and causes the previous letter to be displayed (if available).

OnAlphabetPageButtonClick

```
public void OnAlphabetPageButtonClick()
```

Called when the Alphabet Page button is clicked and causes the Alphabet Page to be shown.

OnTitlePageButtonClick

```
public void OnTitlePageButtonClick()
```

Called when the Title Page button is clicked and causes the Title Page to be shown.

OnGetNextExampleButtonClick

```
public void OnGetNextExampleButtonClick()
```

Called when the Get Next Example button is clicked and causes the next word and picture to be shown for the current letter.

activated

```
public void activated()
```

Lets the controller know that this view has become active.

edu.bu.cs673.AwesomeAlphabet.view
Class PageView

```
java.lang.Object
  |
  +-edu.bu.cs673.AwesomeAlphabet.view.PageView
```

All Implemented Interfaces:

java.util.Observer

Direct Known Subclasses:

AlphabetPageView, LetterPageView, TitlePageView

---

public abstract class **PageView**
extends java.lang.Object
implements java.util.Observer

This class defines the Page View. It is the abstract parent class of all page view classes, including AlphabetPageView, LetterPageView, etc. It manages the page name and panel.

Field Summary

protected static	border
protected static	letterFont
protected	m_panel
private	m_sPageName
protected static final	NAV_BUTTON_ALPHABET_PAGE Value: **Graphics/AlphabetPage.png**
protected static final	NAV_BUTTON_ALPHABET_SONG Value: **Graphics/Music.png**
protected static final	NAV_BUTTON_HOME Value: **Graphics/Home.png**
protected static final	NAV_BUTTON_NEXT_EXAMPLE Value: **Graphics/NextExample.png**
protected static final	NAV_BUTTON_NEXT_LETTER Value: **Graphics/NextArrow.png**
protected static final	NAV_BUTTON_PREV_LETTER Value: **Graphics/PreviousArrow.png**

protected static final	NAV_BUTTON_START Value: **Graphics/StartButton.png**
protected static final	NAV_BUTTON_TITLE_PAGE Value: **Graphics/Home.png**
protected static	wordFont

Constructor Summary

public	PageView(java.lang.String sPageName) Class constructor.

Method Summary

abstract void	activated()
javax.swing.JButton	getButtonImage(java.lang.String filename, java.lang.String defaultName) Gets a button image from the graphics code.
java.lang.String	getPageName() Gets the page name.
javax.swing.JPanel	getPagePanel() Gets the panel.

Methods inherited from class `java.lang.Object`

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

Methods inherited from interface `java.util.Observer`

update

Fields

m_sPageName

`private java.lang.String` **`m_sPageName`**

m_panel

`protected edu.bu.cs673.AwesomeAlphabet.view.AAJPanel` **`m_panel`**

letterFont

`protected static java.awt.Font` **`letterFont`**

(continued from last page)

wordFont

```
protected static java.awt.Font wordFont
```

border

```
protected static javax.swing.border.Border border
```

NAV_BUTTON_TITLE_PAGE

```
protected static final java.lang.String NAV_BUTTON_TITLE_PAGE
```

Constant value: `Graphics/Home.png`

NAV_BUTTON_ALPHABET_SONG

```
protected static final java.lang.String NAV_BUTTON_ALPHABET_SONG
```

Constant value: `Graphics/Music.png`

NAV_BUTTON_HOME

```
protected static final java.lang.String NAV_BUTTON_HOME
```

Constant value: `Graphics/Home.png`

NAV_BUTTON_NEXT_LETTER

```
protected static final java.lang.String NAV_BUTTON_NEXT_LETTER
```

Constant value: `Graphics/NextArrow.png`

NAV_BUTTON_PREV_LETTER

```
protected static final java.lang.String NAV_BUTTON_PREV_LETTER
```

Constant value: `Graphics/PreviousArrow.png`

NAV_BUTTON_NEXT_EXAMPLE

```
protected static final java.lang.String NAV_BUTTON_NEXT_EXAMPLE
```

Constant value: `Graphics/NextExample.png`

NAV_BUTTON_ALPHABET_PAGE

```
protected static final java.lang.String NAV_BUTTON_ALPHABET_PAGE
```

(continued from last page)

Constant value: **`Graphics/AlphabetPage.png`**

NAV_BUTTON_START

```
protected static final java.lang.String NAV_BUTTON_START
```

Constant value: **`Graphics/StartButton.png`**

Constructors

PageView

```
public PageView(java.lang.String sPageName)
```

Class constructor.

Parameters:

`sPageName` - The name of the page associated with this view.

Methods

getPageName

```
public java.lang.String getPageName()
```

Gets the page name.

Returns:

The page name as a String.

getPagePanel

```
public javax.swing.JPanel getPagePanel()
```

Gets the panel.

Returns:

The panel as a JPanel.

getButtonImage

```
public javax.swing.JButton getButtonImage(java.lang.String filename,
         java.lang.String defaultName)
```

Gets a button image from the graphics code. If the image could not be loaded, then it creates text button using the default text.

Parameters:

`filename` - the graphics file to read

`defaultName` - the name of the text button used as a fallback

Returns:

a button ready for displaying

(continued from last page)

activated

```
public abstract void activated()
```

edu.bu.cs673.AwesomeAlphabet.view
Class TitlePageView

```
java.lang.Object
  |
  +-edu.bu.cs673.AwesomeAlphabet.view.PageView
      |
      +-edu.bu.cs673.AwesomeAlphabet.view.TitlePageView
```

All Implemented Interfaces:
java.awt.event.ActionListener, java.util.Observer

public class **TitlePageView**
extends PageView
implements java.util.Observer, java.awt.event.ActionListener

This class defines the Title Page View. From this view, the user is able to see the application's name and version and go to the Alphabet Page.

Field Summary

private	`m_controller`

Fields inherited from class `edu.bu.cs673.AwesomeAlphabet.view.PageView`

`border`, `letterFont`, `m_panel`, `m_sPageName`, `NAV_BUTTON_ALPHABET_PAGE`, `NAV_BUTTON_ALPHABET_SONG`, `NAV_BUTTON_HOME`, `NAV_BUTTON_NEXT_EXAMPLE`, `NAV_BUTTON_NEXT_LETTER`, `NAV_BUTTON_PREV_LETTER`, `NAV_BUTTON_START`, `NAV_BUTTON_TITLE_PAGE`, `wordFont`

Constructor Summary

public	`TitlePageView(java.lang.String sPageName)` Class constructor.

Method Summary

void	`actionPerformed(java.awt.event.ActionEvent ae)` Handles button click events.
void	`activated()`
void	`OnStartButtonClick()` Called when the Start Button is clicked and causes the Alphabet Page to be shown.
void	`SetController(TitlePageController controller)` Sets the controller associated with this view.
void	`update(java.util.Observable o, java.lang.Object arg)` Causes the view to be updated when the model changes.

Methods inherited from class `edu.bu.cs673.AwesomeAlphabet.view.PageView`

`activated`, `getButtonImage`, `getPageName`, `getPagePanel`

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

Methods inherited from interface `java.util.Observer`
`update`

Methods inherited from interface `java.awt.event.ActionListener`
`actionPerformed`

Fields

m_controller

```
private edu.bu.cs673.AwesomeAlphabet.controller.TitlePageController m_controller
```

Constructors

TitlePageView

```
public TitlePageView(java.lang.String sPageName)
```

Class constructor.

Parameters:
`sPageName` - The page name associated with this view.

Methods

actionPerformed

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

Handles button click events.

SetController

```
public void SetController(TitlePageController controller)
```

Sets the controller associated with this view.

Parameters:
`controller` - The controller.

update

```
public void update(java.util.Observable o,
        java.lang.Object arg)
```

(continued from last page)

Causes the view to be updated when the model changes. Currently this method does nothing. It is required for the observer pattern.

OnStartButtonClick

```
protected void OnStartButtonClick()
```

Called when the Start Button is clicked and causes the Alphabet Page to be shown.

activated

```
public void activated()
```

Index