

[Overview](#) [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](#)

dataProcessing

Class Controller

java.lang.Object

dataProcessing.Controller

```
public class Controller
extends java.lang.Object
```

Die Klasse Controller dient der Kapselung der Datenverwaltung und ist Teil des MVC-Design. Der Controller speichert und verwaltet die aktuell geöffnete Sitzung des Programms. Mittels des Controllers kann eine Seite geladen und modifiziert werden. Alle komplexeren Aufgaben, welcher nicht ohne Verarbeitungslogik von View zu Model weitergeleitet werden können werden vom Controller verarbeitet.

Constructor Summary

Constructors

Constructor and Description

Controller (Model model)

Es wird eine Instanz eines Controllers erstellt und der Verzeichnisbaum aktualisiert
--

Method Summary

Methods

Modifier and Type	Method and Description
Content	createContent (ContentType type, int x, int y) Auf der Aktuell geöffneten Seite wird ein Seiteninhalt erstellt, und somit auf der aktuellen Seite als ungespeicherte Änderung vermerkt.
void	deleteContent (Content c) Ein übergebener Seiteninhalt wird auf der aktuellen Seite gelöscht.
Content []	getContent () Es wird der Inhalt der aktuell geöffneten Seite als Content[] zurückgegeben
PageInformation	getCurrentPageInformation () Es werden die Seiteninformationen zur aktuellen Seite zurückgegeben.
Tree	getDataTree () Der aktuell geladene Verzeichnisbaum wird zurückgegeben.

void	loadPage (java.lang.String pageName, java.lang.String chapterName, java.lang.String bookName) Es zu einem vorgegebenen Pfad eine Seite aus dem Model geladen und als aktuelle Seite gespeichert.
void	refreshTree () Der Verzeichnisbaum wird aktualisiert.
void	saveContent (Content c) Ein übergebener Seiteninhalt wird auf der aktuellen Seite gespeichert.

Methods inherited from class java.lang.Object

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

Constructor Detail

Controller

```
public Controller(Model model)
    throws java.sql.SQLException
```

Es wird eine Instanz eines Controllers erstellt und der Verzeichnisbaum aktualisiert

Throws:

java.sql.SQLException - Datenbankfehler

Method Detail

loadPage

```
public void loadPage(java.lang.String pageName,
    java.lang.String chapterName,
    java.lang.String bookName)
    throws DirectoryDoesNotExistsException,
        java.sql.SQLException,
        java.io.IOException,
        java.lang.ClassNotFoundException
```

Es zu einem vorgegebenen Pfad eine Seite aus dem Model geladen und als aktuelle Seite gespeichert. Sollte die angeforderte Seite Startseite sein, wird eine Instanz der Startseite erstellt und in currentPage gespeichert. Die geladene Seite ist direkt nicht von der View aus ansprechbar, sondern es müssen die Inhalte oder die PageInformation über die entsprechenden Methoden abgefragt werden.

Parameters:

pageName - der Name der Seite

chapterName - der Name des Kapitels

bookName - der Name des Buches

Throws:

`DirectoryDoesNotExistsException` - Pfad nicht korrekt

`java.sql.SQLException` - Datenbankfehler

`java.lang.ClassNotFoundException` - Fehler beim deserialisieren

`java.io.IOException`

getContent

```
public Content[] getContent()  
    throws java.lang.NullPointerException
```

Es wird der Inhalt der aktuell geöffneten Seite als `Content[]` zurückgegeben

Returns:

der Inhalt der aktuell geöffneten seite

Throws:

`java.lang.NullPointerException` - es ist aktuell keine Seite geladen

createContent

```
public Content createContent(ContentType type,  
    int x,  
    int y)
```

Auf der Aktuell geöffneten Seite wird ein Seiteninhalt erstellt, und somit auf der aktuellen Seite als ungespeicherte Änderung vermerkt.

Parameters:

type - der Typ des Seiteninhalts

x - Die x-Koordinate an welcher der Inhalt erstellt werden soll

y - Die y-Koordinate an welcher der Inhalt erstellt werden soll

Returns:

der erstellte Seiteninhalt

saveContent

```
public void saveContent(Content c)  
    throws DirectoryDoesNotExistsException,  
    java.sql.SQLException,  
    java.io.IOException
```

Ein übergebener Seiteninhalt wird auf der aktuellen Seite gespeichert. Sollte die aktuell geöffnete Seite nicht

die Startseite sein wird diese gespeichert.

Parameters:

c - der Seiteninhalt

Throws:

`DirectoryDoesNotExistsException` - der Pfad der Seite ist falsch

`java.sql.SQLException` - Datenbankfehler

`java.io.IOException` - Fehler beim Serialisieren

deleteContent

```
public void deleteContent(Content c)
    throws DirectoryDoesNotExistsException,
           java.sql.SQLException,
           java.io.IOException
```

Ein übergebener Seiteninhalt wird auf der aktuellen Seite gelöscht. Sollte die aktuell geöffnete Seite nicht die Startseite sein wird diese gespeichert.

Parameters:

c - der Seiteninhalt

Throws:

`DirectoryDoesNotExistsException` - der Pfad der Seite ist falsch

`java.sql.SQLException` - Datenbankfehler

`java.io.IOException` - Fehler beim Serialisieren

refreshTree

```
public void refreshTree()
    throws java.sql.SQLException
```

Der Verzeichnisbaum wird aktualisiert. Diese Methode wird bei instanziierten des Controllers aufgerufen, und muss danach von der View aufgerufen werden, falls die Anzeige des Verzeichnisbaums im GUI aktualisiert werden soll. Der vom Model geladene Verzeichnisbaum wird in `dataTree` gespeichert und kann über die Methode `getDataTree()` abgefragt werden.

Throws:

`java.sql.SQLException` - Datenbankfehler

getDataTree

```
public Tree getDataTree()
```

Der aktuell geladene Verzeichnisbaum wird zurückgegeben. Gegebenfalls muss dieser vorher mittels der Methode `refreshTree()` aktualisiert werden.

Returns:

der Verzeichnisbaum

getCurrentPageInformation

```
public PageInformation getCurrentPageInformation()
```

Es werden die Seiteninformationen zur aktuellen Seite zurückgegeben.

Returns:

die Seiteninformationen

[Overview](#) [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](#)
