

pageData

Class Page

java.lang.Object
pageData.Page

Direct Known Subclasses:

Startpage

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

Field Summary

Fields

Modifier and Type	Field and Description
static int	DEFAULT_HEIGHT
static int	DEFAULT_WIDTH

Constructor Summary

Constructors

Constructor and Description
Page (PageInformation pageInfo) Erstellt eine neue Seite mit parametrierbaren Seiteninformationen.

Method Summary

Methods

Modifier and Type	Method and Description
void	addContent (Content cont) Dies ist nur die Schnittstelle der Seite zum Model.

void	contentInstructionCompleted() Der Seite wird mitgeteilt, dass eine contentInstruction korrekt bearbeitet wurde.
Content	createContent(Content cont) Es wird ein neuer Seiteninhalt innerhalb der Seite gespeichert Der neue Inhalt wird mit einer Nummer versehen und zurückgegeben.
void	deleteContent(int number) Der übergebene Seiteninhalt wird von der Seite gelöscht.
Content[]	getContent()
Content	getContent(int number)
int	getContentInstructionSize() Diese Methode gibt die Anzahl der noch nicht in der Datenbank gespeicherten Änderungen zurück
ContentInstruction	getNextContentInstruction() Liefert die nächste ContentInstruction.
PageInformation	getPageInfo()
void	saveContent(Content cont) Der übergebene Seiteninhalt wird innerhalb der Seite gespeichert.
void	setPageInfo(PageInformation pageInfo)

Methods inherited from class java.lang.Object

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

Field Detail

DEFAULT_WIDTH

```
public static final int DEFAULT_WIDTH
```

See Also:

[Constant Field Values](#)

DEFAULT_HEIGHT

```
public static final int DEFAULT_HEIGHT
```

See Also:

[Constant Field Values](#)

Constructor Detail

Page

```
public Page(PageInformation pageInfo)
```

Erstellt eine neue Seite mit parametrierbaren Seiteninformationen. Die Nummer des ersten Seiteninhalts wird auf 0 gestetzt.

Method Detail

getPageInfo

```
public PageInformation getPageInfo()
```

setPageInfo

```
public void setPageInfo(PageInformation pageInfo)
```

getContent

```
public Content[] getContent()
```

getContent

```
public Content getContent(int number)
    throws DirectoryDoesNotExistsException
```

Throws:

`DirectoryDoesNotExistsException`

getNextContentInstruction

```
public ContentInstruction getNextContentInstruction()
```

Liefert die nächste ContentInstruction. Es sollte vorher überprüft werden, ob noch ungespeicherte Änderungen vorhanden sind. Achtung: wird die ContentInstruction korrekt bearbeitet muss zuerst "contentInstructionCompleted" aufgerufen werden, damit die nächste ContentInstruction bearbeitet werden kann. Somit wird sichergestellt, dass die Daten jederzeit konsistent sind, auch wenn es beim Speichern einer Änderung zu einem Fehler kommt.

Returns:

die nächste contentInstruction

Throws:

`java.lang.RuntimeException` - es wurden bereits alle `contentInstructions` bearbeitet

getContentInstructionSize

```
public int getContentInstructionSize()
```

Diese Methode gibt die Anzahl der noch nicht in der Datenbank gespeicherten Änderungen zurück

Returns:

Anzahl der ungespeicherten Änderungen

createContent

```
public Content createContent(Content cont)
```

Es wird ein neuer Seiteninhalt innerhalb der Seite gespeichert. Der neue Inhalt wird mit einer Nummer versehen und zurückgegeben. Der Zähler für die Nummer des nächsten Seiteninhalts wird inkrementiert. Es wird eine `contentInstruction` angelegt.

Returns:

der Seiteninhalt mit Nummer

deleteContent

```
public void deleteContent(int number)
```

Der übergebene Seiteninhalt wird von der Seite gelöscht. Falls er nicht Teil der Seite ist, wird eine Exception geworfen. Falls der Seiteninhalt noch gar nicht in der Datenbank gespeichert wurde, wird die Anweisung den Inhalt zu erstellen aus den `contentInstructions` gelöscht. Falls nicht, wird eine neue `contentInstruction` angelegt, welche dokumentiert, dass der Inhalt aus der Datenbank gelöscht werden soll.

Parameters:

`number` - die Nummer des zu löschenden Seiteninhalts

Throws:

`DirectoryDoesNotExistsException` - es existiert kein Seiteninhalt mit der angegebenen Nummer

saveContent

```
public void saveContent(Content cont)
```

Der übergebene Seiteninhalt wird innerhalb der Seite gespeichert. Es wird den `contentInstructions` ein Eintrag hinzugefügt, sodass die Änderung in der Datenbank übernommen wird, falls diese Seite das nächste mal gespeichert wird.

Parameters:

cont - der zu speichernde Inhalt

contentInstructionCompleted

```
public void contentInstructionCompleted()
```

Der Seite wird mitgeteilt, dass eine contentInstruction korrekt bearbeitet wurde. Die contentInstruction wird gelöscht und es kann die nächste abgefragt werden.

Throws:

`java.lang.RuntimeException` - Es wurden bereits alle ContentInstructions abgeschlossen

addContent

```
public void addContent(Content cont)
```

Dies ist nur die Schnittstelle der Seite zum Model. Wird die Seite aus der Datenbank geladen, ruft das Model jeden dazugehörigen Seiteninhalt ab und fügt ihn mit dieser Methode zur Seite hinzu. Dem Inhalt wird einer Nummer hinzugefügt und der Nummerzähler wird inkrementiert

Parameters:

cont - Der hinzuzufügende Inhalt

[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](#)