

Hochschule Darmstadt

EWA Page Templates (EWA-Seitenklassen)

Prof. Dr. Bernhard Kreling, Prof. Dr. Ralf Hahn

Version 2.0

4/5/2020 9:09:00 AM

Table of Contents

Hierarchical Index	Fehler! Textmarke nicht definiert.
Class Index	3
Page	Fehler! Textmarke nicht definiert.
Class Documentation.....	4
BlockTemplate	4
Page	6
PageTemplate	8
Index.....	10

Class Index

Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

BlockTemplate	4
Page	6
PageTemplate	8

Class Documentation

BlockTemplate

Public Member Functions

- **__construct** (\$database)
- **generateView** (\$id="")
- **processReceivedData** ()

Protected Member Functions

- **getViewData** ()

Protected Attributes

- **\$_database** = null

Detailed Description

This is a template for classes, which represent blocks (e.g. a form) within a web page. Instances of these classes are used as members of top level classes. The order of methods might correspond to the order of thinking during implementation.

Author

Bernhard Kreling, `bernhard.kreling@h-da.de`

Ralf Hahn, `ralf.hahn@h-da.de`

Constructor & Destructor Documentation

__construct (*\$database*)

Gets the reference to the DB from the calling page template. Stores the connection in member `$_database`.

Parameters

<i>\$database</i>	\$database is the reference to the DB to be used
-------------------	--------------------------------------------------

Returns

none

Member Function Documentation

generateView (*\$id* = "")

Generates an HTML block embraced by a div-tag with the submitted id. If the block contains other blocks, delegate the generation of their parts of the view to them.

Parameters

<i>\$id</i>	\$id is the unique (!!) id to be used as id in the div-tag
-------------	------------------------------------------------------------

Returns

none

getViewData () [protected]

Fetch all data that is necessary for later output. Data is stored in an easily accessible way e.g. as associative array.

Returns

none

processReceivedData ()

Processes the data that comes via GET or POST i.e. CGI. If this block is supposed to do something with submitted data do it here. If the block contains other blocks, delegate processing of the respective subsets of data to them.

Returns

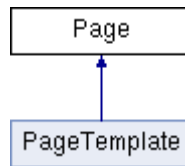
none

Member Data Documentation**\$_database = null [protected]**

Reference to the MySQLi-Database that is accessed by all operations of the class.

Page

Inheritance diagram for Page:



Public Member Functions

- `__destruct ()`

Protected Member Functions

- `__construct ()`
- `generatePageHeader ($headline="")`
- `generatePageFooter ()`
- `processReceivedData ()`

Protected Attributes

- `$_database = null`

Detailed Description

This abstract class is a common base class for all HTML-pages to be created. It manages access to the database and provides operations for outputting header and footer of a page. Specific pages have to inherit from that class. Each inherited class can use these operations for accessing the db and for creating the generic parts of a HTML-page.

Author

Bernhard Kreling, `bernhard.kreling@h-da.de`

Ralf Hahn, `ralf.hahn@h-da.de`

Constructor & Destructor Documentation

`__construct ()` [`protected`]

Connects to DB and stores the connection in member `$_database`.

Needs name of DB, user, password.

Returns

none

Reimplemented in **PageTemplate** (*p.8*).

`__destruct ()`

Closes the DB connection and cleans up

Returns

none

Reimplemented in **PageTemplate** (*p.8*).

Member Function Documentation

generatePageFooter () [protected]

Outputs the end of the HTML-file i.e. /body etc.

Returns

none

generatePageHeader (\$headline = "") [protected]

Generates the header section of the page. i.e. starting from the content type up to the body-tag. Takes care that all strings passed from outside are converted to safe HTML by htmlspecialchars.

Parameters

<i>\$headline</i>	\$headline is the text to be used as title of the page
-------------------	--------------------------------------------------------

Returns

none

processReceivedData () [protected]

Processes the data that comes via GET or POST i.e. CGI. If every page is supposed to do something with submitted data do it here. E.g. checking the settings of PHP that influence passing the parameters (e.g. magic_quotes).

Returns

none

Reimplemented in **PageTemplate** (p.9).

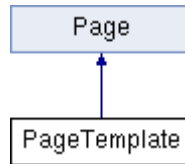
Member Data Documentation

\$_database = null [protected]

Reference to the MySQLi-Database that is accessed by all operations of the class.

PageTemplate

Inheritance diagram for PageTemplate:



Public Member Functions

- `__destruct ()`

Static Public Member Functions

- `static main ()`

Protected Member Functions

- `__construct ()`
- `getViewData ()`
- `generateView ()`
- `processReceivedData ()`

Additional Inherited Members

Detailed Description

This is a template for top level classes, which represent a complete web page and which are called directly by the user. Usually there will only be a single instance of such a class. The name of the template is supposed to be replaced by the name of the specific HTML page e.g. baker. The order of methods might correspond to the order of thinking during implementation.

Author

Bernhard Kreling, `bernhard.kreling@h-da.de`

Ralf Hahn, `ralf.hahn@h-da.de`

Constructor & Destructor Documentation

`__construct ()` [`protected`]

Instantiates members (to be defined above).

Calls the constructor of the parent i.e. page class. So the database connection is established.

Returns

none

Reimplemented from **Page** (*p.6*).

`__destruct ()`

Cleans up what ever is needed.

Calls the destructor of the parent i.e. page class. So the database connection is closed.

Returns

none

Reimplemented from **Page** (p.6).

Member Function Documentation

generateView () [protected]

First the necessary data is fetched and then the HTML is assembled for output. i.e. the header is generated, the content of the page ("view") is inserted and -if available- the content of all views contained is generated. Finally the footer is added.

Returns

none

getViewData () [protected]

Fetch all data that is necessary for later output. Data is stored in an easily accessible way e.g. as associative array.

Returns

none

static main () [static]

This main-function has the only purpose to create an instance of the class and to get all the things going. I.e. the operations of the class are called to produce the output of the HTML-file. The name "main" is no keyword for php. It is just used to indicate that function as the central starting point. To make it simpler this is a static function. That is you can simply call it without first creating an instance of the class.

Returns

none

processReceivedData () [protected]

Processes the data that comes via GET or POST i.e. CGI. If this page is supposed to do something with submitted data do it here. If the page contains blocks, delegate processing of the respective subsets of data to them.

Returns

none

Reimplemented from **Page** (p.7).

Index

- \$_database
 - BlockTemplate 6
 - Page 8
- __construct
 - BlockTemplate 5
 - Page 7
 - PageTemplate 9
- __destruct
 - Page 7
 - PageTemplate 9
- BlockTemplate 5
 - \$_database 6
 - __construct 5
 - generateView 5
 - getViewData 6
 - processReceivedData 6
- generatePageFooter
 - Page 8
- generatePageHeader
 - Page 8
- generateView
 - BlockTemplate 5
 - PageTemplate 10

- getViewData
 - BlockTemplate 6
 - PageTemplate 10
- main
 - PageTemplate 10
- Page 4, 7
 - \$_database 8
 - __construct 7
 - __destruct 7
 - generatePageFooter 8
 - generatePageHeader 8
 - processReceivedData 8
- PageTemplate 9
 - __construct 9
 - __destruct 9
 - generateView 10
 - getViewData 10
 - main 10
 - processReceivedData 10
- processReceivedData
 - BlockTemplate 6
 - Page 8
 - PageTemplate 10