



Web Application Component Toolkit

Mocsnik Norbert

Harmadik Magyarországi PHP Konferencia
2005. március 12., Budapest



Mi a WACT?

- moduláris keretrendszer
- segítséget nyújt EnterprisePattern megvalósításokhoz PHP nyelvű webes alkalmazásokban
- szándékoltnan elősegíti a Refactoring és TestDrivenDesign gyakorlatok alkalmazását
- elősegíti jól struktúrált, biztonságos webalkalmazások fejlesztését
- eseményvezérelt



EnterprisePatterns

- ModelViewController
- TemplateView
- PageController
- FrontController
- ApplicationController
- TransactionScript
- RecordSet



Future Support is planned for

- DomainModel
- InterceptingFilter



Adatterületek





class DataSpace

```
var $properties = array();
```

```
function get($name)
```

```
function set($name, $value)
```



class DataSpace

```
function &getObject($name)
```

```
function getPath($path)
```

```
function &getObjectPath($path)
```

```
function setObject($name, &$value)
```

```
function setPath($path, $value)
```

```
function setObjectPath($path, &$value)
```



class DataSpace

function remove(\$name)

function removeAll()

function import(\$property_list)

function merge(\$property_list)

function &export()

function isDataSource()

function hasProperty(\$name)

function getPropertyList()



class DataSourceSupport

```
var $_properties = array();  
function addProperty($name, $value)  
function hasProperty($name)  
function getPropertyList()  
function get($name)  
function set($name, $value)  
function removeProperty($name)  
function removeAllProperties()
```



class DataSourceSupport

```
function addObjectProperty(  
    $name, &$value)
```

```
function setObject($name, &$value)
```

```
function &getObject($name)
```

```
function import($property_list)
```

```
function setMany($list)
```

```
function export()
```



class DataSourceSupport

```
function propertyChanged($name)
```

```
function propertyChange(  
    $name, &$old, &$new)
```

```
function &propertyNotFound($name)
```



DataSourceSupport leszámazottak

```
class EmptyDataSource extends  
DataSourceSupport
```

```
class ArrayDataSource extends  
DataSourceSupport
```



class DataSourceDecorator

```
var $dataSource;
```

```
function DataSourceDecorator  
(&$dataSource)
```



class DataSourceDecorator

```
function get($name)
```

```
function set($name, $value)
```

```
function &getObject($name)
```

```
function setObject($name, &$value)
```

```
function getPath($path)
```

```
function setPath($path, $value)
```



class DataSourceDecorator

function remove(\$name)

function removeAll()

function import(\$propertyList)

function merge(\$propertyList)

function &export()

function isDataSource()

function hasProperty(\$name)



class DataSourceMapper extends DataSourceDecorator

```
var $getters;  
var $sources;  
function DataSourceMapper(  
    $dataSource)  
function addProperty($name,  
    $source=NULL, $getter=NULL)  
function get($name) {  
    return $this->dataSource->{$getter}  
    ($source);  
}
```




class DictionaryDataSource

```
var $_properties = array();
```

```
function get($name)
```

```
function set($name, $value)
```



class DictionaryDataSource

```
function &getObject($name)
```

```
function getPath($path)
```

```
function &getObjectPath($path)
```

```
function setObject($name, &$value)
```

```
function setPath($path, $value)
```

```
function setObjectPath($path, &$value)
```



class DictionaryDataSource

```
function clear($name)
```

```
function clearAll()
```

```
function import($property_list)
```

```
function merge($list)
```

```
function export()
```

```
function isDataSource()
```

```
function hasProperty($name)
```

```
function getPropertyList()
```



Kérés / Válasz

HTTP Request/Response related



class Request

```
function hasParameters()  
function hasParameter($name)  
function getParameter($name)  
function isPostBack($expectedMethod)  
function hasPostProperty($name)  
function getPostProperty($name)  
function exportPostProperties()
```



class RequestFilter

(EXPERIMENTAL!)

function RequestFilter(&\$base)

function hasParameters()

function hasParameter(\$name)

function getParameter(\$name)

function isPostBack(\$expectedMethod)

function hasPostProperty(\$name)

function getPostProperty(\$name)

function exportPostProperties()



ResponseModel extends DataSpace

```
var $isValid = TRUE;  
var $errorList = NULL;  
function addError($group, $id,  
    $fieldList = NULL, $values = NULL)  
function addErrorMessage($message,  
    $fieldList = NULL)  
function isValid()  
function &getErrorList()  
function applyRule(&$rule)
```

Iterator-ok





class IteratorBase

(php5: implements Iterator)

function valid()
function key()
function ¤t()
function next()
function rewind()



class IteratorDecorator

```
var $iterator;
```

```
function valid()
```

```
function key()
```

```
function &current()
```

```
function next()
```

```
function rewind()
```



class TransformIterator extends IteratorDecorator

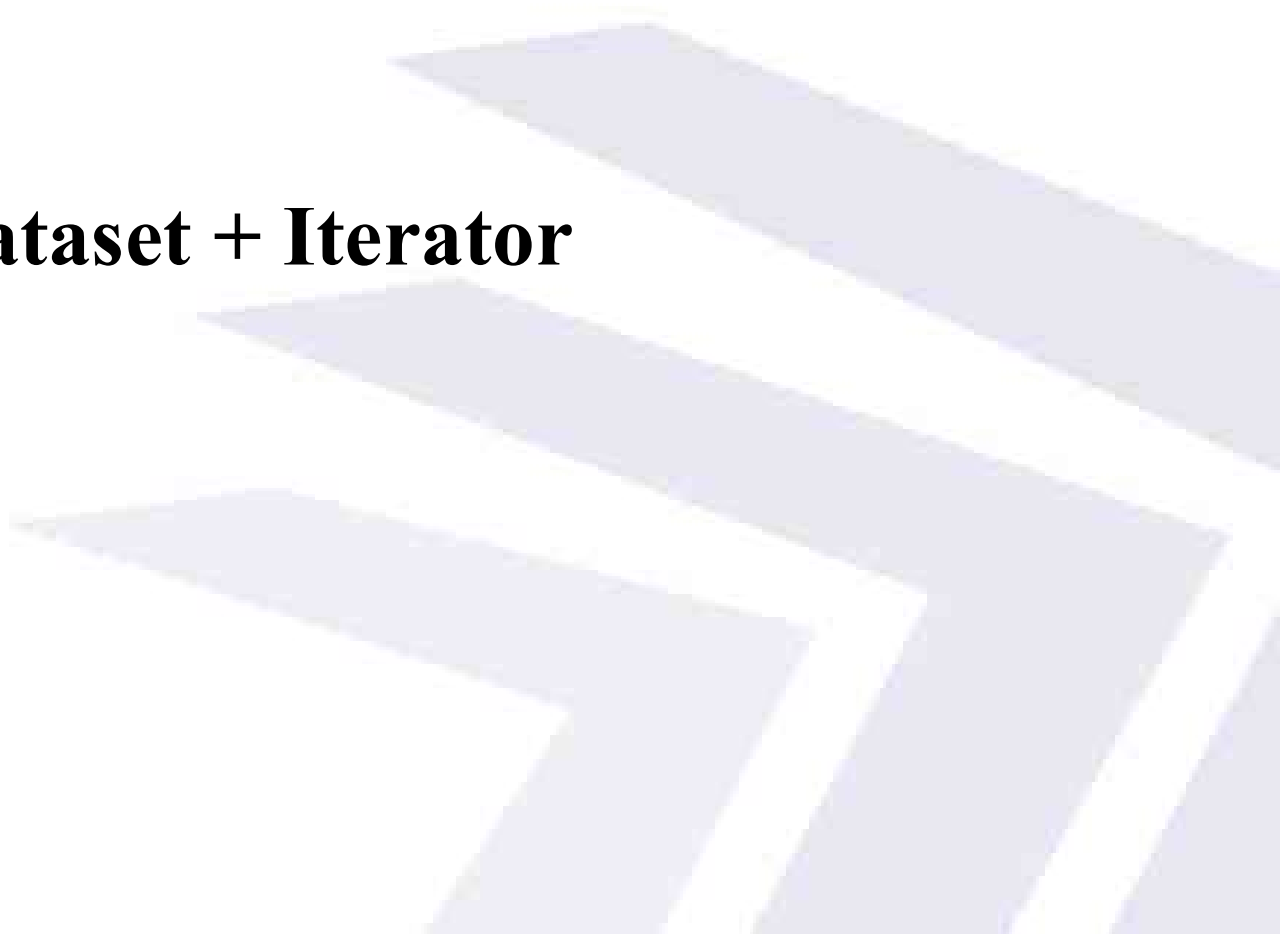
```
var $transformer;
```

```
function &current() {
```

```
    return $this->transformer->transform(  
        $this->iterator->current());
```

```
}
```

Dataset + Iterator





class ArrayDataSet extends IteratorBase

```
function importDataSetAsArray(  
    $dataSet)
```

```
function exportDataSetAsArray()
```

```
function rewind()
```

```
function next()
```



class KeyValueDataSet extends IteratorBase

```
function KeyValueDataSet(  
    $array,  
    $key_field = "key",  
    $value_field = "value")
```

```
function rewind()
```

```
function next()
```



class MutableDataSet extends ArrayDataSet

function unshiftRow(\$row)

function shiftRow()

function pushRow(\$row)

function popRow()

function insertRow(\$row)

function deleteRow()

function seekRow(\$index)



class PagedArrayDataSet extends ArrayDataSet

function getTotalRowCount()

function getRowCount()

function paginate(&\$pager)

function importDataSetAsArray(
 \$dataSet)

function rewind()

function next()



Utils

Handle, Delegate

class Handle

- LazyEvaluation of object references
- classfile.php|class
- paraméterek
- ```
$this->setDefaultView(
 new Handle('View', array(
 '/eloadasok/adatlap.html')));
```



# Delegates

- Delegate
- StaticDelegate

```
$this->registerOnLoadListener(
 new Delegate($this, 'load'));
```



# Utils

**DB modul**



# DB modul

- egységes API
- MySQL közvetlen elérése
- ADODB, MDB, MDB2, PEAR DB kompatibilitás



# class DBC

- konfiguráció automatikus betöltése
- adatbáziskapcsolat automatikus felépítése
- rövid függvények a leggyakoribb adatbázisműveletek elvégzésére



# SQL Statement (utasítás)

- `DBC::newStatement($sql, $params)`  
**SELECT \* FROM table WHERE id = :id:**
- `setNull($name), set($name, $value)`
- `setSmallInt(), setInteger(), setFloat(), setDouble(), setDecimal(), setBoolean(), setChar(), setVarChar(), setText(), setDate(), setTime(), setTimeStamp(), setBlob()`
- `import($paramList)`
- `getSQL(), execute()`



# SQL RecordSet (rekordhalmaz)

- `DBC::newRecordSet($query, $params)`

- Visszatérési érték:

`class MySQLRecordSet extends  
IteratorBase`





# PagedRecordset (lapozható adathalmaz)

- function &NewPagedRecordSet(  
\$query, &\$pager, \$params = array())
- Visszatérési érték:  
class MySQLRecordSet extends  
IteratorBase

paginate:

```
$this->pager =& $pager;
```

```
$pager->setPagedDataSet($this);
```



# get(X)ColumnAsArray (oszlop lekérdezése tömbként)

- `function getOneColumnAsArray($query, $params = array())`
- visszatérési értéke egyszerű tömb  
`$ret = array(1, 2, 3);`
- `function getTwoColumnAsArray($query, $params = array())`
- visszatérési értéke asszociatív tömb  
`$ret = array(1 => 'xxx', 2 => 'yyy');`



# FindRecord (rekordkeresés)

- function &FindRecord(  
\$query, \$params = array())
- visszatérési érték:  
MySQLRecord extends DataSpace
- getBoolean, getIntegerTimeStamp,  
getStringDate, getStringTime,  
getStringTimeStamp, getStringFixed



# DBC::GetOneValue (egyetlen érték lekérdezése)

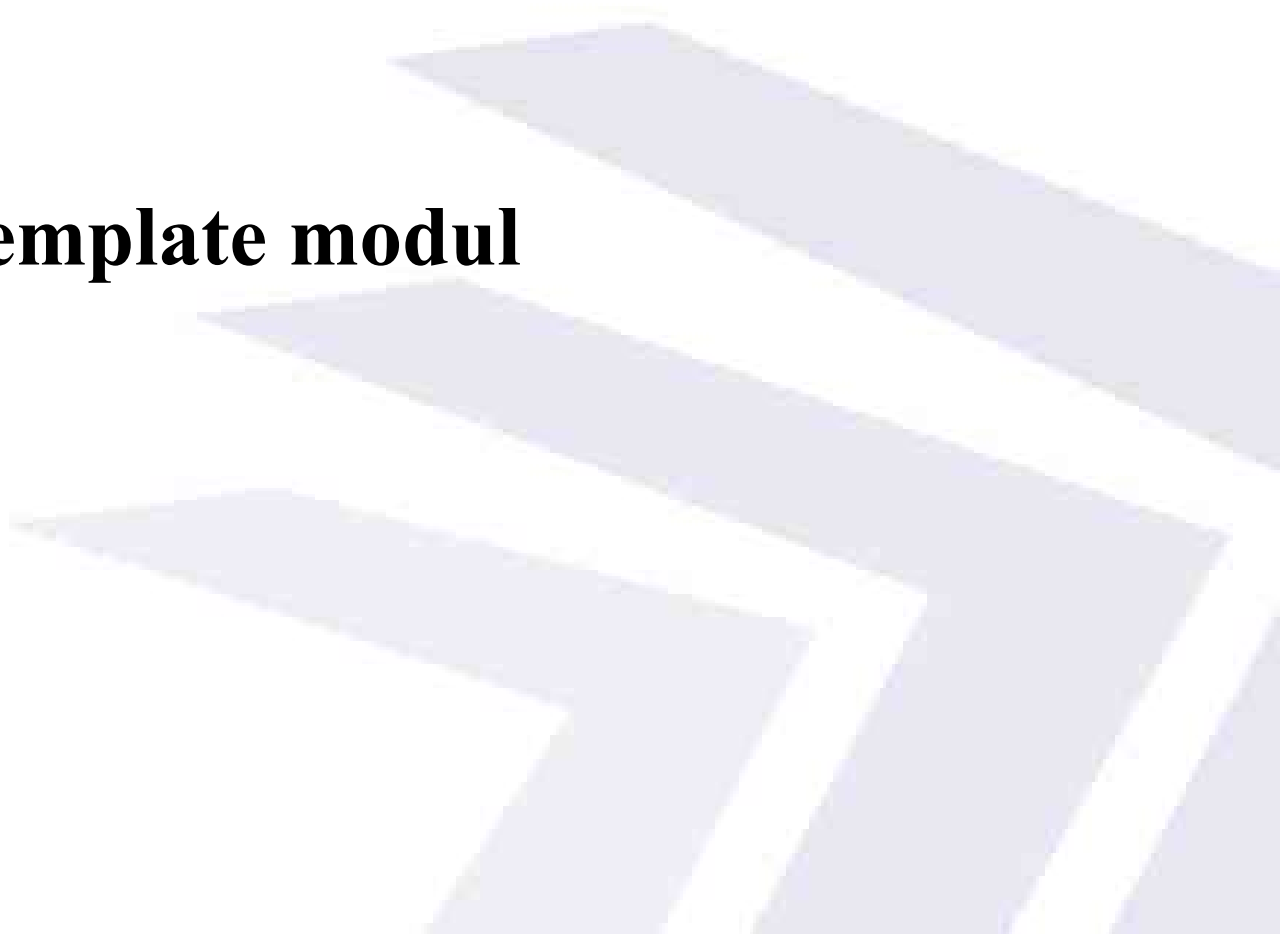
- `function getOneValue($query, $params = array())`
- visszatérési érték:  
mixed (általában string)



# DBC::execute (SQL utasítás futtatása)

- function execute(  
\$query, \$params = array())
- visszatérési értéke  
boolean (true/false)

# Template modul





# Bad Template Smells

- Duplication
- Compiling at runtime
- Custom template syntax
- Control Structures in Template
- Claims to separate code from HTML
- Exhaustive binding logic



# Good Template Smells

- Simplicity
- Validating
- Cacheable
- Self-Inspecting
- Secure
- Flexibility (non-HTML)
- Pass the Dreamweaver test





# Template View

- Push versus **Pull** Data Population
- Component APIs





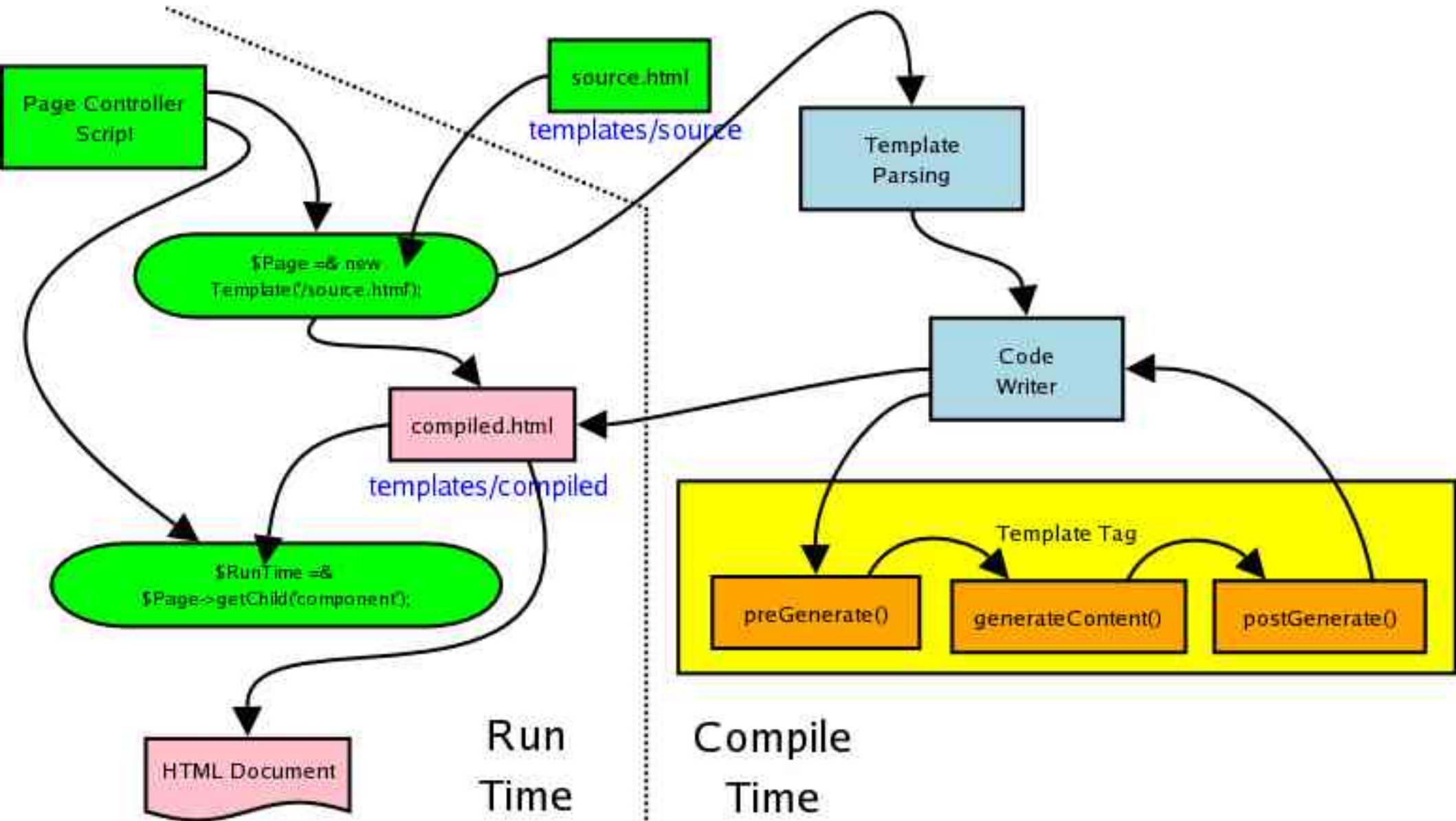
www.phpcon.hu

# Fájlsémák

- Simple
  - /templates/source
  - /templates/compiled
- Multilang
  - /templates/source/templates/
  - /templates/source/lang/en/
  - /templates/source/lang/hu/

```
define('TMPL_FILESCHEME_PATH',
FRAMEWORK_ROOT .
'template/fileschemes/simple/');
```

# Template Component Architecture





# Tag-ek (Tags)

- class CompilerComponent

subclasses:

- SilentCompilerDirectiveTag (-)
- CompilerDirectiveTag (compiletime)
- ServerComponent (run/compiletime)
- TagServerTagComponentTag  
(+ run/compiletime + DataSource)



# Kifejezések (expressions)

- `{ $PlainText }`
- `{ $#TodaysDate }`, `{ $^UserName }`,  
`{ $^^UserName }`
- Szűrők: `CapitalizeFilter`,  
`LowerCaseFilter`, `NoTagsFilter`,  
`TextFilter`, `TrimFilter`, `UpperCaseFilter`,  
`WordWrapFilter`, `DateFilter`,  
`NumberFilter`, `MathFilter`
- `{ $PlainText|wordwrap:size }`



# Komponensek (Components)

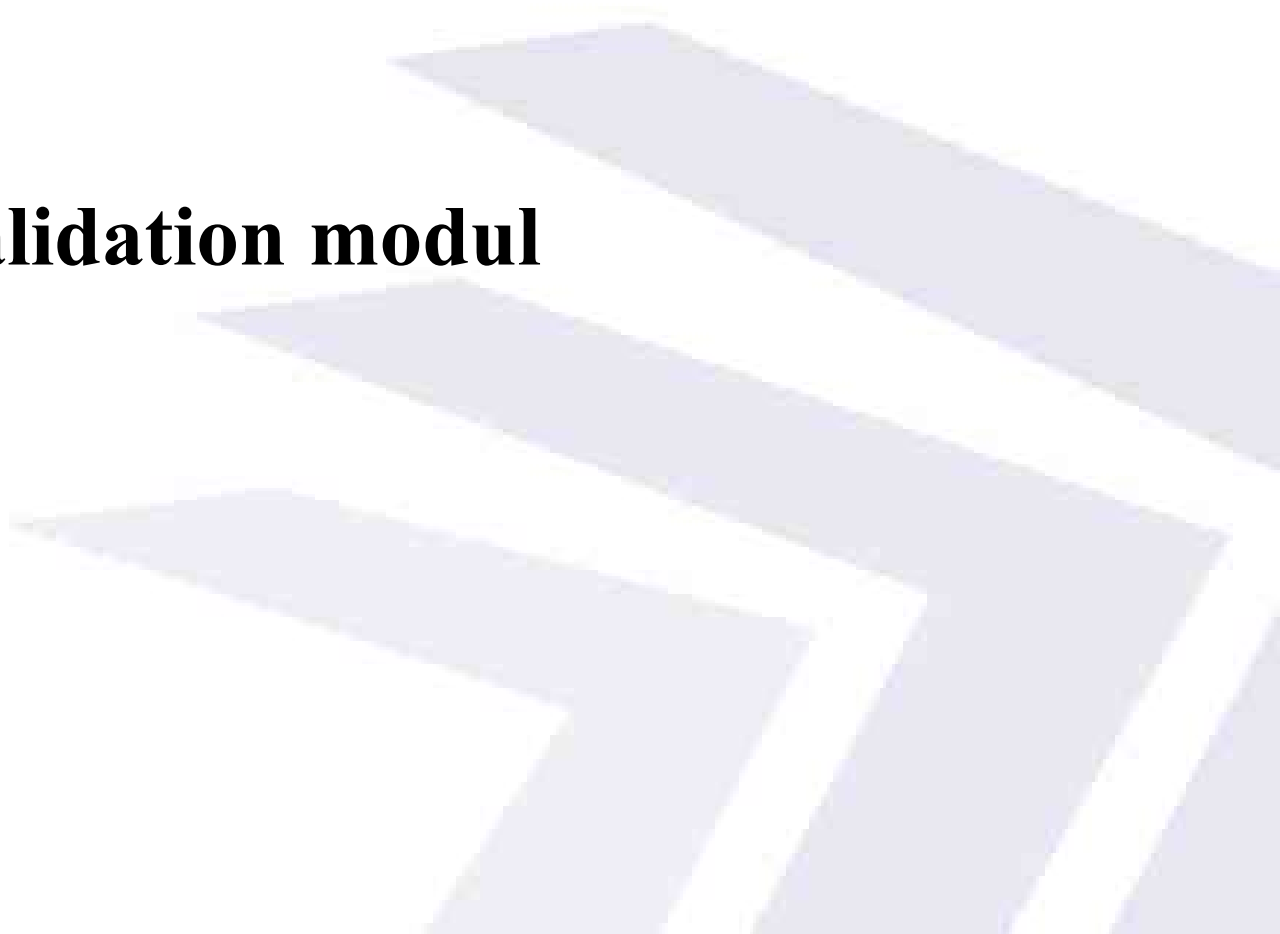
- Component
- TagComponent
  - function getClientId()
  - function setAttribute(\$attrib, \$value)
  - function getAttribute(\$attrib)
  - function hasAttribute(\$attrib)



# Widget-ek (Widgets)

- allow normal HTML to be manipulated with the TagComponent API at RunTime
- declaring widgets:  
add `runat="server"` attribute

# **Validation modul**







# class Validator

```
var $rules = array();
```

```
var $ErrorList;
```

```
var $IsValid = TRUE;
```

```
function &createErrorList()
```

```
function addRule(&$Rule)
```

```
function &getErrorList()
```

```
function IsValid($FieldName = NULL)
```

```
function validate(&$DataSource)
```



# class Rule

- function setGroup(\$Group)
- function validate(&\$DataSource, &\$ErrorList) [abstract]



# class SingleFieldRule extends Rule

```
function SingleFieldRule($fieldname)
```

```
function getField()
```

```
function Error($id, $values = NULL)
```

```
function IsValid()
```

```
function validate(&$DataSource,
 &$ErrorList)
```

```
function Check($value) [abstract]
```



class RequiredRule extends  
SingleFieldRule

```
function validate(&$DataSource,
&$ErrorList)
```





# class SizeRangeRule extends SingleFieldRule

```
var $minLength;
var $maxLength;
```

```
function Check($value)
```



# class NumericRangeRule extends SingleFieldRule

```
var $min;
```

```
var $max;
```

```
function Check($value)
```



```
class EmailRule
extends DomainRule
```

```
function Check($value)
```

```
function CheckUser($value)
```

```
function CheckDomain($value)
```



class PatternRule extends  
SingleFieldRule

function Check(\$value)







class ExcludePatternRule extends  
SingleFieldRule

function Check(\$value)





# class ErrorList extends IteratorBase

```
var $errors = array();
```

```
var $fieldRestriction;
```

```
function addError(
 $Group, $Id,
 $FieldList = NULL, $Values = NULL)
```

```
function addErrorMessage(
 $Message, $FieldList = NULL)
```



# class ErrorList extends IteratorBase

```
function rewind()
```

```
function next()
```

```
function restrictFields($fieldRestriction) {
```

```
function removeRestrictions() {
```

```
function isRestricted($fieldList) {
```



# class ErrorList extends IteratorBase

```
function setFieldNameDictionary
(&$Dictionary)
```

```
class DefaultFieldNameDictionary {
 function getFieldName($name) {
 return $name;
 }
}
```



```
class ErrorList
extends IteratorBase
```

```
function setErrorMessageDictionary
(&$Dictionary)
```

```
class ErrorMessageDictionary {
 var $ErrorMessages = array();
 function getMessage($Group, $Id) {
 return $this->ErrorMessages
 [$Group][$Id];
 }
}
```



# Egy egyszerű példa

- PHP Konferencia előadások nyilvántartása
  - adatlap megtekintése
  - adatlap szerkesztése / tárolás validációval
  - előadások listázása lapozóval
  - előadás törlése



# Sitemap

[index.php/eloadasok](http://index.php/eloadasok)

[index.php/eloadasok/uj](http://index.php/eloadasok/uj)

[index.php/eloadasok/megtekint?id=11](http://index.php/eloadasok/megtekint?id=11)

[index.php/eloadasok/szerkeszt?id=11](http://index.php/eloadasok/szerkeszt?id=11)

[index.php/eloadasok/torol?id=11](http://index.php/eloadasok/torol?id=11)

[index.php/eloadasok/listaz](http://index.php/eloadasok/listaz)



# Könyvtárszerkezet PHP kód

`/index.php`

`/lib/eloadasok/eloadasok.model.php`

`/lib/eloadasok/add.page.php`

`/lib/eloadasok/details.page.php`

`/lib/eloadasok/edit.page.php`

`/lib/eloadasok/delete.page.php`





# Könyvtárszerkezet HTML sablonok

`/templates/source/layout.html`

`/templates/source/eloadasok/details.html`

`/templates/source/eloadasok/add.html`

`/templates/source/eloadasok/edit.html`

`/templates/source/eloadasok/fields.html`

`/templates/source/eloadasok/list.html`

`/templates/compiled/*`



# /templates/source/layout.html

```
<html><head></head>
```

```
<body>
```

```
<core:placeholder id="Content">
```

```
</body>
```

```
</html>
```



# Front Controller index.php

```
$eloadasok = new
PathInfoDispatchController();
```

```
$eloadasok->addChild('uj',
new Handle(APP_LIB_ROOT .
'eloadasok/add.page.php'|
EloadasokAddPage'));
```

```
$eloadasok->addChild('szerkeszt',
new Handle(APP_LIB_ROOT .
'eloadasok/edit.page.php'|
EloadasokEditPage'));
```



# Front Controller index.php

```
$eloadasok->addChild('torol',
 new Handle(APP_LIB_ROOT .
 'eloadasok/delete.page.php |
 EloadasokDeletePage')));
```

```
$eloadasok->addChild('listaz',
 new Handle(APP_LIB_ROOT .
 'eloadasok/list.page.php |
 EloadasokListPage')));
```



# Front Controller index.php

```
$eloadasok->addChild('megtekint',
new Handle(APP_LIB_ROOT .
'eloadasok/details.page.php|
EloadasokDetailsPage'));
```

```
$eloadasok->setDefaultChild('listaz');
```



# Front Controller index.php

```
$Front = new
PathInfoDispatchController();
$Front->addChild('eloadasok',
$eloadasok);

$Front->setDefaultChild('eloadasok');

$Front->start();
```



# SQL szkript

```
CREATE TABLE 'eloadasok' (
 'id' int(11) NOT NULL auto_increment,
 'nev' varchar(255) NOT NULL default '',
 'megerositve' tinyint(4) NOT NULL default
 '0',
 'nem' set('ferfi','no') NOT NULL default '',
 'idotartam' int(11) NOT NULL default '0',
 'leiras' text NOT NULL,
 PRIMARY KEY ('id')
) ENGINE=MyISAM DEFAULT
 CHARSET=latin2 AUTO_INCREMENT=1 ;
```



# DB modul - class Eloadasok eloadasok/eloadasok.model.php

```
function &getList(&$Pager) {
 $sql = 'SELECT * FROM eloadasok';
 $sql .= ' ORDER BY nev';

 return DBC::NewPagedRecordSet(
 $sql, $Pager);
}
```





# DB modul - class Eloadasok eloadasok/eloadasok.model.php

```
function &getRecord($id) {
 $sql = 'SELECT * FROM eloadasok';
 $sql .= ' WHERE id = :id:';

 return DBC::FindRecord(
 $sql, array('id' => $id));
}
```



# DB modul - class Eloadasok eloadasok/eloadasok.model.php

```
function getFields() {
 return array(
 'nev', 'megerositve', 'nem',
 'idotartam', 'igenyek', 'leiras'
);
}
```



# DB modul - class Eloadasok eloadasok/eloadasok.model.php

```
function insert(&$data) {
 $conn =& DBC::getCurrentConnection();
 $fields = Eloadasok::getFields();

 $Builder = new MySqlInsertStatementBuilder(
 $conn, 'eloadasok', $fields, $data);

 return $Builder->insertId();
}
```



# DB modul - class Eloadasok eloadasok/eloadasok.model.php

```
function update($id, &$data) {
 $conn =& DBC::getCurrentConnection();
 $fields = Eloadasok::getFields();

 $Builder = new MySqlUpdateStatementBuilder(
 $conn, 'eloadasok', $fields, $data,
 array('id' => $id));

 return $Builder->execute();
}
```



# DB modul - class Eloadasok eloadasok/eloadasok.model.php

```
function delete($id) {
 $sql = 'DELETE FROM eloadasok'
 $sql .= ' WHERE id = :id:';

 return DBC::execute(
 $sql, array('id' => $id));
}
```



# megtekintés – 'details'

**Adatlap megjelenítése**



megtekint – class EloadasokDetailsPage  
eloadasok/details.page.php

```
class EloadasokDetailsPage extends
PageController {

 function EloadasokDetailsPage() {
 parent::PageController();
 $this->setDefaultView(
 new Handle('View', array(
 '/eloadasok/adatlap.html')));
 }
}
```



megtekint – HTML sablon

templates/source/eloadasok/details.html

```
<core:WRAP file="/layout.html" placeholder="Content">
```

```
<h3>Előadás adatlap</h3>
```

```
<table>
```

```
<tr>
```

```
<td>Előadás megnevezése</td><td>{$nev}</td>
```

```
<td>Időtartam</td><td>{$idotartam}</td>
```

```
</tr> [.....]
```

```
</table>
```

```
<html:link url="/eloadasok/szerkeszt"
 param="id">szerkesztés</html:link>
```





# megtekint – class EloadasokDetailsPage eloadasok/details.page.php

```
class EloadasokDetailsPage extends PageController {
 function EloadasokDetailsPage() {
 parent::PageController();

 $this->registerOnLoadListener(
 new Delegate($this, 'load'));

 $this->setDefaultView(
 new Handle('View', array(
 '/eloadasok/adatlap.html')));
 }
}
```



megtekint – class EloadasokDetailsPage  
eloadasok/details.page.php

```
function load(&$source, &$request,
&$responseModel) {

 $rec =& Eloadasok::getRecord(
 $request->getParameter('id'));

 $responseModel->import($rec->export());

}
```



# megtekint – class EloadasokDetailsPage eloadasok/details.page.php

```
function load(&$source, &$request, &$responseModel)
{
 $rec =& Eloadasok::getRecord(
 $request->getParameter('id'));

 if (is_null($rec)) {
 return 'notfound';
 }

 $responseModel->import($rec->export());
}
```



# megtekint – class EloadasokDetailsPage eloadasok/details.page.php

```
function EloadasokDetailsPage() {
 parent::PageController();
 $this->registerOnLoadListener(
 new Delegate($this, 'load'));
 $this->setDefaultView(
 new Handle('View', array(
 '/eloadasok/adatlap.html')));
$this->addView('notfound',
new Handle('RedirectView', array(
 '/static/display404')));
}
```



# megtekint – class EloadasokDetailsPage eloadasok/details.page.php

```
function EloadasokDetailsPage() {
 parent::PageController();

 $this->registerOnLoadListener(
 new Delegate($this, 'load'));

$this->registerOnActivateListener(
 new Delegate($this, 'guard'));

 $this->setDefaultView(
 new Handle('View', array(
 '/eloadasok/adatlap.html')));

 $this->addView('notfound',
 new Handle('RedirectView', array(
 '/static/display404')));
}
```



# megtekint – class EloadasokDetailsPage eloadasok/details.page.php

```
function guard(&$source, &$request,
&$responseModel) {
 if (!$request->hasParameter('id')) {
 return 'notfound';
 }
}
```



# törlés – 'delete'

## **Rekord törlése**



# torol – class EloadasokDeletePage eloadasok/delete.page.php

```
class EloadasokDeletePage extends PageController {
 function EloadasokDeletePage()
 parent::PageController();
 $this->registerOnActivateListener(
 new Delegate($this, 'guard'));
 $this->registerOnLoadListener(
 new Delegate($this, 'process'));
 $this->addView('failure', new Handle(
 'RedirectView', array('/static/display404')));
 $this->addView('success', new Handle(
 'RedirectView', array('/eloadasok/listaz')));
}
```





torol – class EloadasokDeletePage  
eloadasok/delete.page.php

```
function guard(&$source, &$request,
&$responseModel) {
 if (!$request->hasParameter('id')) {
 return 'notfound';
 }
}
```



torol – class EloadasokDeletePage  
eloadasok/delete.page.php

```
function process(&$Source, &$request,
&$responseModel) {
 $res = Eloadasok::delete(
 $request->getParameter('id'));

 if (!$res) {
 return 'failure';
 }

 return 'success';
}
```



# beszúrás – 'add'

**Új rekord felvitele**



# HTML sablon

templates/source/eloadasok/fields.html

```
<tr>
 <td>* <label for="nev" runat="server">
 Előadó neve</label></td>
 <td><input name="nev" type="text" id="nev"
 runat="server"></td>
</tr>
<tr>
 <td><label for="megerositve" runat="server">
 Megerősítve</label></td>
 <td><input name="megerositve" type="checkbox"
 id="megerositve" runat="server">
 </td>
</tr>
```



# HTML sablon

templates/source/eloadasok/fields.html

```
<tr>
 <td>* <label for="nem" runat="server">
 Nem</label>
 </td>
 <td>
 <input name="nem" type="radio" id="nem1"
 runat="server" value="ferfi"> férfi
 <input name="nem" type="radio" id="nem2"
 runat="server" value="no"> nő
 </td>
</tr>
```



# HTML sablon

templates/source/eloadasok/fields.html

```
<tr>
 <td>* <label for="idotartam" runat="server">
 Időtartam</label></td>
 <td><select name="idotartam" id="idotartam"
 runat="server"></select></td>
```

```
</tr>
```

```
<tr>
 <td><label for="igenyek" runat="server">
 Igények</label></td>
 <td><select name="igenyek[]" id="igenyek"
 runat="server" multiple="multiple">
 </select></td>
```

```
</tr>
```



# HTML sablon

templates/source/eloadasok/fields.html

```
<tr>
 <td>* <label for="leiras" runat="server">
 Leírás</label>
 </td>
 <td>
 <textarea name="leiras" id="leiras"
 runat="server"></textarea>
 </td>
</tr>
```



# HTML sablon

templates/source/eloadasok/add.html

```
<core:WRAP file="/layout.html" placeholder="Content">
<h3>Új előadás felvitele</h3>
<table>
 <form id="AddForm" name="AddForm"
 method="post" action="" runat="server">
 <core:include file="/eloadasok/fields.html">

 </form>
</table>
```





# uj – class EloadasokAddPage eloadasok/add.page.php

```
function EloadasokAddPage() {
 parent::PageController();

 $this->setDefaultView(
 new Handle('FormView', array(
 '/eloadasok/add.html')));
}
```



```
class EloadasokUjView
 extends FormView
```

```
function EloadasokUjView($TemplateFile) {
 parent :: FormView($TemplateFile);
}
```

```
function prepare() {
 $igenyek =& $this->Template->getChild
 ('igenyek')

 $igenyek->setChoices(
 EloadoiIgenyek::getTwoColumnAsArray();
);
}
```



# uj – class EloadasokAddPage eloadasok/add.page.php

```
function EloadasokAddPage() {
 parent::PageController();

 $this->setDefaultView(
 new Handle('EloadasokUjView', array(
 '/eloadasok/add.html')));
}
```



uj – class EloadasokAddPage  
eloadasok/add.page.php

```
$SubmitButton =& new ButtonController();
$SubmitButton->registerOnClickListener(
 new Delegate($this, 'process'));
```

```
$Form = new FormController();
$Form->addChild('submit', $SubmitButton);
$Form->setDefaultChild('submit');
```

```
$this->addChild('AddForm', $Form);
$this->setDefaultChild('AddForm');
$this->setDefaultView(
 new Handle('EloadasokUjView', array(
 '/eloadasok/add.html')));
```



uj – class EloadasokAddPage  
eloadasok/add.page.php

```
function process(&$source, &$request,
&$responseModel) {

 if ($responseModel->isValid()) {
 $id = Eloadasok::insert($responseModel);
 if (!$id) {
 return 'failure';
 }

 return 'success';
 }
}
```



# uj – class EloadasokAddPage eloadasok/add.page.php

```
$Form = new FormController(); [...]
$Form->setDefaultChild('submit');
```

```
$Form->addRule(new Handle('RequiredRule',
 array('nev')));
$Form->addRule(new Handle('SizeRangeRule',
 array('nev', 4, 32)));
$Form->addRule(new Handle('MemberRule',
 array('nem', array('ferfi', 'no'))));
$Form->addRule(new Handle('EmailRule',
 array('email')));
```



# HTML sablon

templates/source/eloadasok/fields.html

```
<tr><td>
 <errorsummary id='Summary'>
 <list:item>
 {$ErrorMessage}
 <list:separator>
</list:separator>
 </list:item>
 </errorsummary>
</td></tr>

<tr>
 <td>* <label for="nev" runat="server">
 Előadó neve</label></td>
 <td><input name="nev" type="text" id="nev"
 runat="server"></td>

</tr>
```



# HTML sablon

templates/source/eloadasok/fields.html

```
<tr>
<td>
 * <label for="nev" runat="server">
 Előadó neve</label>
</td>
<td>
 <input name="nev" type="text" id="nev"
 runat="server">
 <errorssummary id='firstNameSummary' for='nev'>
 <list:item>
 {$ErrorMessage}
 </list:item>
 </errorssummary>
</td>
</tr>
```





# szerkesztés – 'edit'

**Meglévő rekord szerkesztése**



# HTML sablon

templates/source/eloadasok/edit.html

```
<core:WRAP file="/layout.html" placeholder="Content">
 <h3>Előadás szerkesztése</h3>
 <table>
 <form id="EditForm" name="EditForm"
 method="post" action="" runat="server">
 <input type="hidden" name="id" runat="server">
 <core:include file="/eloadasok/fields.html">

 </form>
 </table>
```



# uj – class EloadasokEditPage eloadasok/edit.page.php

```
function EloadasokEditPage() {
 parent::PageController();

 $this->registerOnActivateListener(
 new Delegate($this, 'guard'));
 $this->registerOnLoadListener(
 new Delegate($this, 'load'));

 $this->setDefaultView(
 new Handle('FormView', array(
 '/eloadasok/edit.html')));
}
```



torol – class EloadasokDeletePage  
eloadasok/delete.page.php

```
function guard(&$source, &$request,
&$responseModel) {
 if (!$request->hasParameter('id')) {
 return 'notfound';
 }
}
```



# uj – class EloadasokEditPage eloadasok/edit.page.php

```
function process(&$source, &$request,
 &$responseModel) {
 if ($responseModel->isValid()) {
 $id = $responseModel->get('id');

 if (!Eloadasok::update($id, $responseModel))
 {
 return 'failure';
 }

 return 'success';
 }
}
```



# listázás - 'list'

## **Rekordok listázása**



# HTML sablon

templates/source/eloadasok/list.html

```
<list:list id='List'>
 <table>
 <tr><td>Név</td><td>Nem</td>
 <td>Időtartam</td><td>Műveletek</td></tr>
 <list:item><tr>
 <td>{ $nev }</td><td>{ $nem }</td>
 <td>{ $idotartam }</td>
 <td>
 <html:link url="/eloadasok/megtekint"
 param="id">Megtekint</html:link>
 <html:link url="/eloadasok/torol"
 param="id">Töröl</html:link>
 </td></tr></list:item>
 </table>
 </list:list>
```



# uj – class EloadasokListPage eloadasok/list.page.php

```
function EloadasokListPage() {
 parent::PageController();

 $this->setDefaultView(
 new Handle('EloadasokListView', array(
 '/eloadasok/list.html')));
}
```





```
class EloadasokListView
 extends View
```

```
function EloadasokListView($TemplateFile) {
 parent :: View($TemplateFile);
}
```

```
function prepare(&$controller, &$request,
 &$responseModel) {
```

```
 $this->Template->setChildDataSet('List',
 &Eloadasok::getList());
```

```
}
```



www.ppke.hu

# HTML sablon

templates/source/eloadasok/list.html

```
<list:list id='List'>
```

```
<page:navigator id="pagenav" items="20">
```

```
Oldal: { $PageNumber } / { $TotalPages }
```

```
<page:first>Első</page:first>
```

```
<page:prev>Előző</page:prev>
```

```
<page:list>
```

```
<page:number><page:elipses>...</page:elipses>
```

```
<page:separator> </page:separator>
```

```
</page:list>
```

```
<page:next>Következő</page:next>
```

```
<page:last>Utolsó</page:last>
```

```
</page:navigator>
```

```
<table>
```

```
[...]
```



```
class EloadasokListView
 extends View
```

```
function EloadasokListView($TemplateFile) {
 parent :: View($TemplateFile);
}
```

```
function prepare(&$controller, &$request,
 &$responseModel) {
 $pager = & $this->Template->getChild(
 'pagenav');
 $list = & Eloadasok::getList($pager);
 $this->Template->setChildDataSet('List',
 &$list);
}
```



# HTML sablon

templates/source/eloadasok/list.html

```
<list:list id='List'>
 <table>
 <tr>
 <td>
 <html:link url="/eloadasok/listaz?order=nev"
 param="id">Név</html:link>
 </td>
 <td>
 <html:link url="/eloadasok/listaz?order=nem"
 param="id">Nem</html:link>
 </td>
 </tr>
 </table>
 [...]

```



# class EloadasokListView extends View

```
function EloadasokListView($TemplateFile) {
 parent :: View($TemplateFile);
}
```

```
function prepare(&$controller, &$request,
 &$responseModel) {
 $order = $request->getParameter('order');
 $pager =& $this->Template->getChild(
 'pagenav');
 $list =& Eloadasok::getList($pager, $order);
 $this->Template->setChildDataSet('List',
 &$list);
}
```



# DB modul - class Eloadasok eloadasok/eloadasok.model.php

```
function &getList(&$Pager, $order) {
 $sql = 'SELECT * FROM eloadasok';
 if (!in_array($order, array('nev', 'nem'))) {
 $order = 'nev';
 }
 $sql .= ' ORDER BY ' . $order;

 return DBC::NewPagedRecordSet($sql,
 $Pager);
}
```



# Egy egyszerű példa

- PHP Konferencia előadások nyilvántartása
  - adatlap megtekintése
  - adatlap szerkesztése / tárolás validációval
  - előadások listázása lapozóval
  - előadás törlése



# Sitemap

[index.php/eloadasok](http://index.php/eloadasok)

[index.php/eloadasok/uj](http://index.php/eloadasok/uj)

[index.php/eloadasok/megtekint?id=11](http://index.php/eloadasok/megtekint?id=11)

[index.php/eloadasok/szerkeszt?id=11](http://index.php/eloadasok/szerkeszt?id=11)

[index.php/eloadasok/torol?id=11](http://index.php/eloadasok/torol?id=11)

[index.php/eloadasok/listaz](http://index.php/eloadasok/listaz)





# Könyvtárszerkezet PHP kód

`/index.php`

`/lib/eloadasok/eloadasok.model.php`

`/lib/eloadasok/add.page.php`

`/lib/eloadasok/details.page.php`

`/lib/eloadasok/edit.page.php`

`/lib/eloadasok/delete.page.php`



# Könyvtárszerkezet HTML sablonok

`/templates/source/layout.html`

`/templates/source/eloadasok/details.html`

`/templates/source/eloadasok/add.html`

`/templates/source/eloadasok/edit.html`

`/templates/source/eloadasok/fields.html`

`/templates/source/eloadasok/list.html`

`/templates/compiled/*`



# Linkek

- <http://wact.sourceforge.net>
- **<http://wact.sourceforge.net/examples>**
- <http://sf.net/projects/wact>
- <http://norbert.mocsnik.hu>
- [norbert@mocsnik.hu](mailto:norbert@mocsnik.hu)
- Kérdések?