

Chương 02

Thao tác với Controller và View

1. *Làm việc với controller*
2. *Thiết lập view và truyền dữ liệu*
3. *Làm chủ 5 cấp độ layout*
4. *Partial view*
5. *Xây dựng project nhiều module theo mô hình chuẩn Phalcon*

Chương 02

Thao tác với Controller và View

- 1. Làm việc với controller*
- 2. Thiết lập view và truyền dữ liệu*
- 3. Làm chủ 5 cấp độ layout*
- 4. Partial view*
- 5. Xây dựng project nhiều module theo mô hình chuẩn Phalcon*

Làm việc với Controller

- Tạo controller và action
- Truyền tham số cho action thông qua đường dẫn url
- Tìm hiểu phương thức __construct và initialize

Truyền tham số cho Action

Truyền một tham số

```
public function indexAction($name){  
    echo 'hello ' . $name;  
}
```

<http://localhost/singlephalcon/control/index/zendvn>

Truyền tham số cho Action

Truyền nhiều tham số

```
public function index2Action($name, $age, $color){  
    echo 'fullname is' . $name;  
    echo '<br>your age is : ' . $age  
    echo '<br>your color is : ' . $color  
}
```

<http://localhost/singlephalcon/control/index/peter/20/green>

Phương thức __construct và initialize

__construct

- Là hàm khởi tạo cho đối tượng của một class
- Được thực thi đầu tiên trong một class
- Không hỗ trợ trong controller

Initialize

- Chức năng tương tự như phương thức __construct

Chương 02

Thao tác với Controller và View

1. *Làm việc với controller*
2. *Thiết lập view và truyền dữ liệu*
3. *Làm chủ 5 cấp độ layout*
4. *Partial view*
5. *Xây dựng project nhiều module theo mô hình chuẩn Phalcon*

Làm việc với View

- Khai báo các thư mục chứa các tập tin view
- Cách thức tạo và đặt tên cho tập tin view
- Truyền tham số từ controller ra ngoài view
- Thay đổi tập tin view mặc định cho action

Làm việc với View

- Mô hình phân cấp layout trong phalcon
- Thiết lập level render cho view
- Disable các cấp layout cho view
- Sử dụng partial trong view

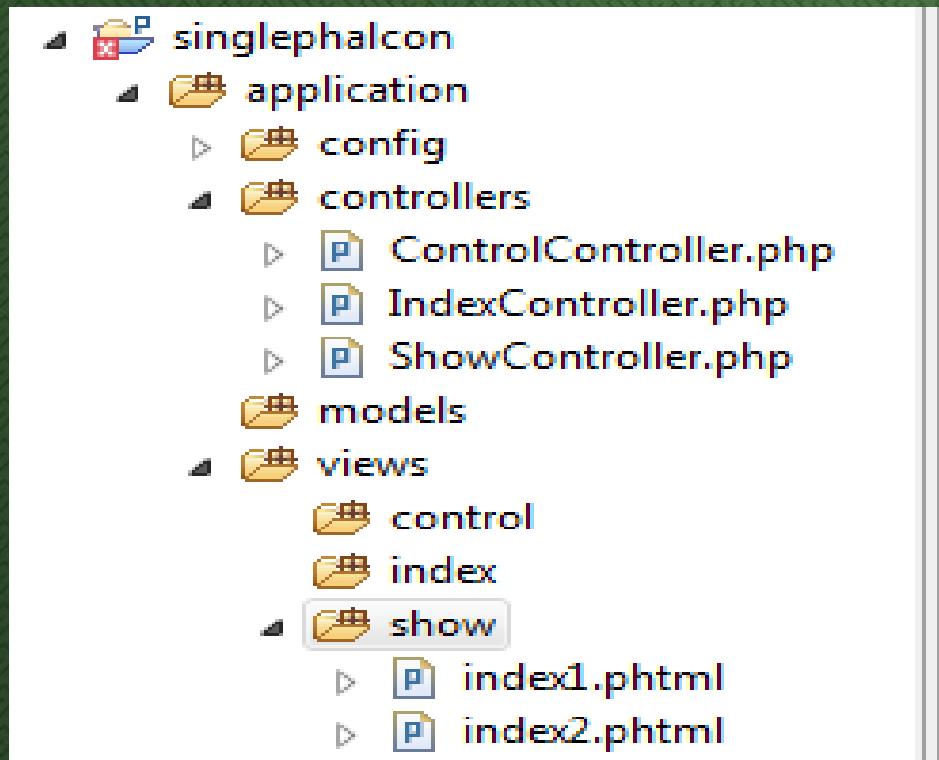
Khai báo thư mục chứa các tập tin View

```
$di = new Phalcon\Di\FactoryDefault();
$di->set('view', function (){
    $view = new \Phalcon\Mvc\View();
    $view->setViewsDir('application/views');
    return $view;
});
```

Cách thức tạo và đặt tên cho tập tin View

Vị trí : views/controller_name

Tên : Trùng với tên của action



```
class showController extends \Phalcon\Mvc\Controller {

    public function index1Action(){

    }

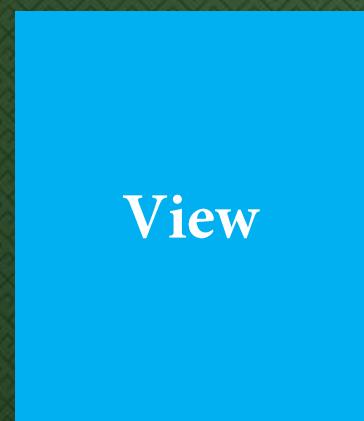
    public function index2Action(){

    }
}
```

Truyền tham số từ Controller ra ngoài View



```
public function index2Action()
{
    $this->view->setVar($key, $value);
}
```

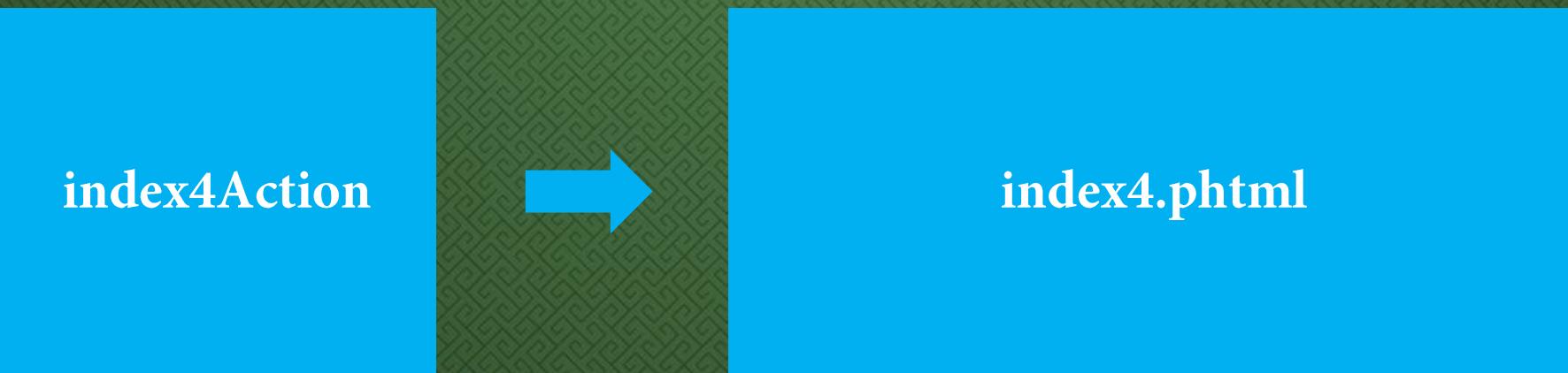


```
public function index2Action()
{
    $this->view->key = 'value';
}
```

A screenshot of a code editor showing two snippets of PHP code. The top snippet is identical to the one above. The bottom snippet shows a line of code: "echo \$key;" with line numbers 3 and 4 visible on the left.

```
3 echo $key;
4
```

Thay đổi tập tin view mặc định cho Action



Thay đổi tập tin view mặc định cho Action

index4Action



views/temp/pick.phtml

```
public function index4Action()
{
    $this->view->pick('temp/pick');
}
```

Chương 02

Thao tác với Controller và View

1. *Làm việc với controller*
2. *Thiết lập view và truyền dữ liệu*
- 3. *Làm chủ 5 cấp độ layout***
4. *Partial view*
5. *Xây dựng project nhiều module theo mô hình chuẩn Phalcon*

Mô hình phân cấp layout trong Phalcon

Loại layout	Vị trí	Đặt tên	Mô tả
1.Main layout	views	Index.phtml	Phần giao diện dùng chung cho các controller
2.Controller layout	Views/layouts	Controller_name.phtml layout1Controller – layout1.phtml	Phần giao diện dùng chung cho các action của cùng 1 controller
3.Action layout	Views/controller_name	action_name.phtml	Phần giao diện dùng riêng cho mỗi action

```
echo $this->getContent();
```

Main Layout

Main layout

Main layout

Main layout

Layout1Controller
index1
index2

Layout2Controller
index3
index4

Layout3Controller
Index5
index6

Controller Layout



Layout1Controller
index1
index2



Layout2Controller
index3
index4

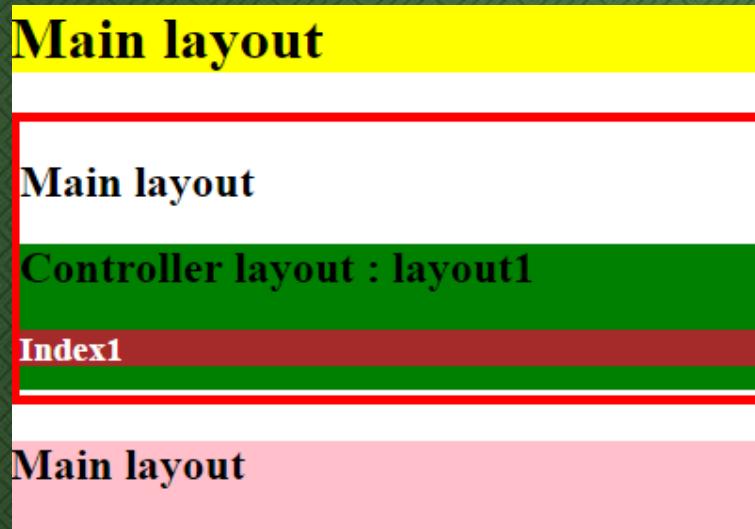
Controller Layout

Main layout

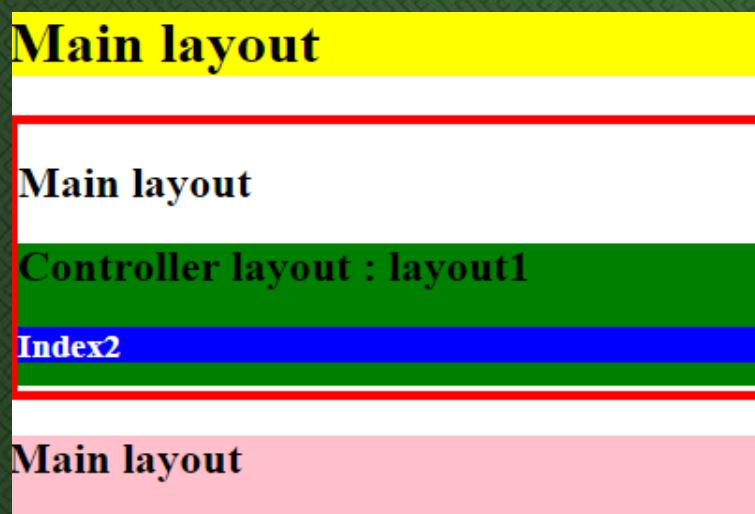
echo \$this->getContent()

Controller layout

Action Layout



Layout1Controller
index1



Layout1Controller
Index2

Action Layout

Main layout

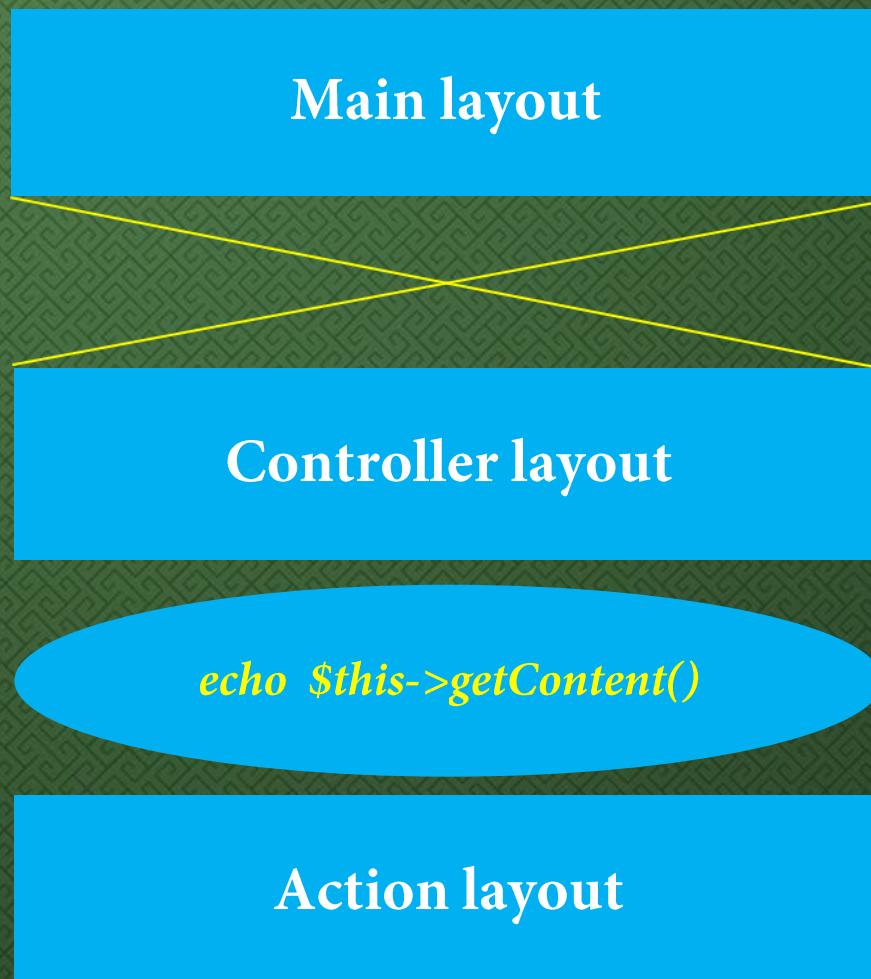
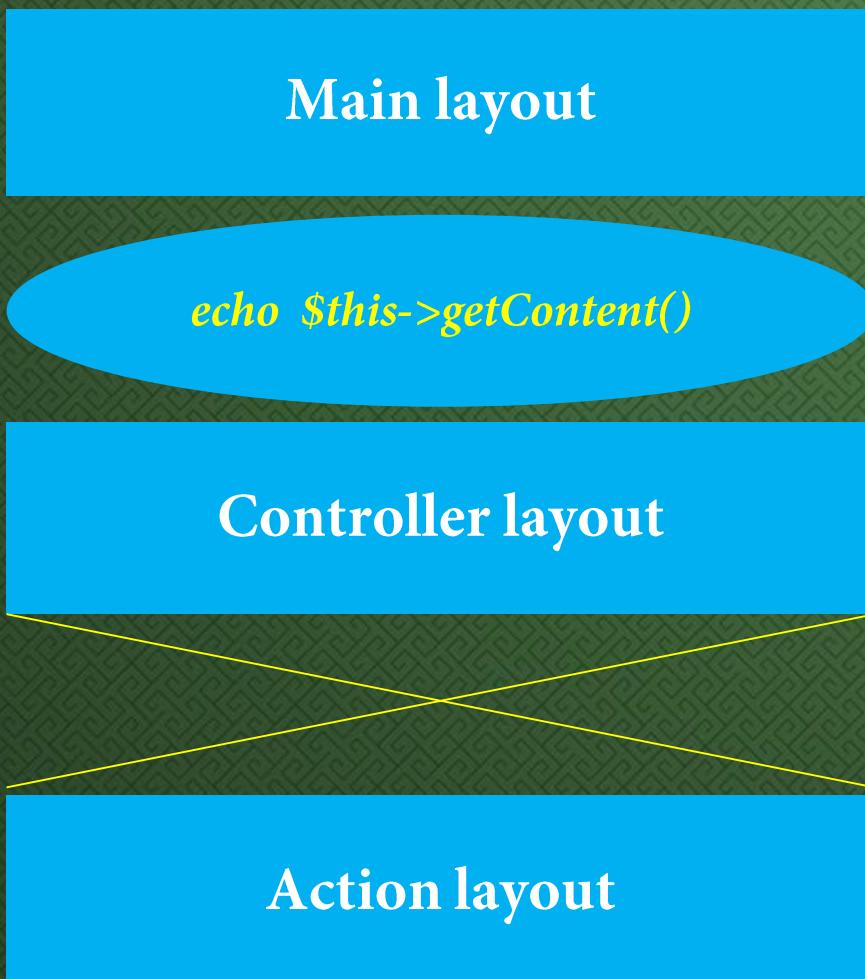
echo \$this->getContent()

Controller layout

echo \$this->getContent()

Action layout

Action Layout



Mô hình phân cấp layout trong Phalcon

Loại	Vị trí	Đặt tên	Cú pháp
Main layout			
After template	View/layouts	Tùy ý	\$this->view->setTemplateAfter('name');
Controller layout	View/layouts		
Before template	View/layouts	Tùy ý	\$this->view>setTemplateBefore('name');
Action layout			

After – Before Template

Mô hình đầy đủ

Main layout – index.phtml

echo \$this->getContent()

After template – after.phtml

echo \$this->getContent()

Controller layout-layout1.phtml

echo \$this->getContent()

Before template-before.phtml

echo \$this->getContent()

Action layout-index1.phtml

Layout1Controller
index1

After – Before Template

Mô hình đầy đủ

Main layout – index.phtml

echo \$this->getContent()

After template – after.phtml

echo \$this->getContent()

Controller layout-layout1.phtml

echo \$this->getContent()

Before template-before.phtml

echo \$this->getContent()

Action layout-index1.phtml

Mô hình Khuyết

After template – after.phtml

echo \$this->getContent()

Controller layout-layout1.phtml

echo \$this->getContent()

Before template-before.phtml

echo \$this->getContent()

Action layout-index1.phtml

After – Before Template

Mô hình đầy đủ

Main layout – index.phtml

echo \$this->getContent()

After template – after.phtml

echo \$this->getContent()

Controller layout-layout1.phtml

echo \$this->getContent()

Before template-before.phtml

echo \$this->getContent()

Action layout-index1.phtml

Mô hình Khuyết

After template – after.phtml

echo \$this->getContent()

Before template-before.phtml

echo \$this->getContent()

Action layout-index1.phtml

After – Before Template

Mô hình đầy đủ

Main layout – index.phtml

```
echo $this->getContent()
```

After template – after.phtml

```
echo $this->getContent()
```

Controller layout-layout1.phtml

```
echo $this->getContent()
```

Before template-before.phtml

```
echo $this->getContent()
```

Action layout-index1.phtml

Mô hình Khuyết

After template – after.phtml

```
echo $this->getContent()
```

Controller layout-layout1.phtml

Before template-before.phtml

```
echo $this->getContent()
```

Action layout-index1.phtml

Thiết lập level render cho View

Mô hình đầy đủ

Main layout – index.phtml

```
echo $this->getContent()
```

After template – after.phtml

```
echo $this->getContent()
```

Controller layout-layout1.phtml

```
echo $this->getContent()
```

Before template-before.phtml

```
echo $this->getContent()
```

Action layout-index1.phtml

Mô hình Khuyết

After template – after.phtml

```
echo $this->getContent()
```

Controller layout-layout1.phtml

```
echo $this->getContent()
```

Before template-before.phtml

```
echo $this->getContent()
```

Action layout-index1.phtml

Thiết lập level render cho View

```
$this->view->setRenderLevel(View::LEVEL_MAIN_LAYOUT);
$this->view->setRenderLevel(View::LEVEL_AFTER_TEMPLATE);
$this->view->setRenderLevel(View::LEVEL_LAYOUT);
$this->view->setRenderLevel(View::LEVEL_BEFORE_TEMPLATE);
$this->view->setRenderLevel(View::LEVEL_ACTION_VIEW);
$this->view->setRenderLevel(View::LEVEL_NO_RENDER);
```

Disable các layout cho View

Phương thức thực hiện

```
$this->view->disableLevel(View::LEVEL_MAIN_LAYOUT)
```

```
$this->view->disableLevel(View::LEVEL_AFTER_TEMPLATE)
```

```
$this->view->disableLevel(View::LEVEL_LAYOUT)
```

```
$this->view->disableLevel(View::LEVEL_BEFORE_TEMPLATE)
```

```
$this->view->disableLevel(View::LEVEL_ACTION_VIEW)
```

Chương 02

Thao tác với Controller và View

1. *Làm việc với controller*
2. *Thiết lập view và truyền dữ liệu*
3. *Làm chủ 5 cấp độ layout*
- 4. *Partial view***
5. *Xây dựng project nhiều module theo mô hình chuẩn Phalcon*

Sử dụng partial trong View

Partial là gì ?

```
<div>
    <h1>Content of partial</h1>
</div>
```

```
<div>
    <h1>Content of view</h1>
</div>

<div>
    <h1>Content of partial</h1>
</div>
```

Sử dụng partial trong View

Khi nào sử dụng Partial ?

```
<div>
    <h1>Content of view 1</h1>
</div>
```

```
<div>
    <h1>Content of view 2</h1>
</div>
```

```
<div>
    <h1>Content of view 3</h1>
</div>
```

Sử dụng Partial trong View

Vị trí	Đặt tên	Cú pháp
views	name.phtml	echo \$this->partial('name',[]);
Views/folder_name	name.phtml	echo \$this->partial('foler_name/name',[]);

Ví dụ với Partial

```
<div>
  <h3 style="color:blue;">Khóa học lập trình PHP</h3>
  <p>PHP đã và đang trở thành ngôn ngữ lập trình phổ biến hiện nay</p>
</div>
```

```
<div>
  <h3 style="color:red">Khóa học lập trình jQuery</h3>
  <p>
    Nếu nói đến lập trình web,
    chúng ta nghĩ ngay đến một ngôn ngữ lập trình không thể thiếu
  </p>
</div>
```

```
<div>
  <h3 style="color:red; text-decoration:underline;">Khóa học lập trình Joomla 2,5</h3>
  <p>Joomla CMS là một hệ thống quản lý dữ liệu được đánh giá rất cao hiện nay</p>
</div>
```