



# WEB разработка

MVC

Codeigniter

ВЪВЕДЕНИЕ

# Съдържание

- MVC
  - Codeigniter framework
  - Controllers
  - Параметри в контролера
  - View
  - controller->view
  - [CI helpers](#)
  - [PHP style guide](#)

# MVC

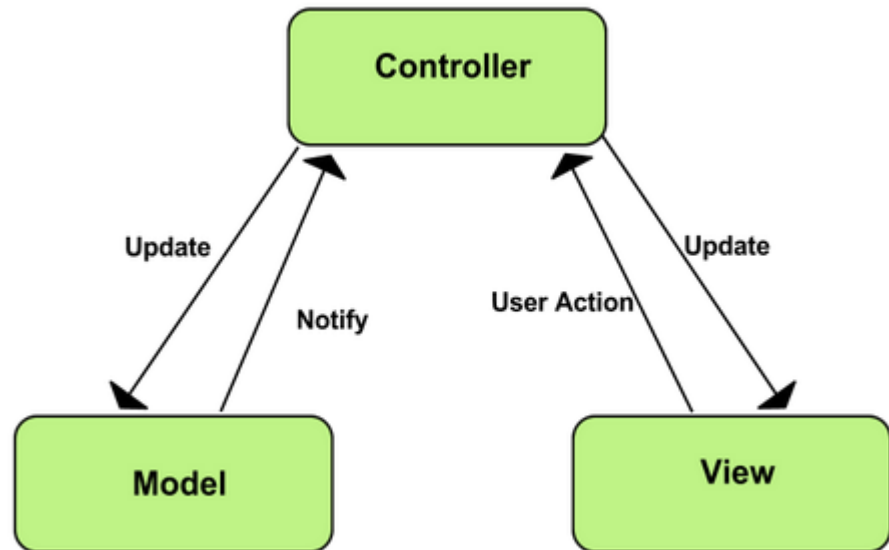
# MVC

MVC - архитектурен шаблон в програмирането, при който бизнес логиката, графичния интерфейс и данните са разделени.

**M**odel - отговаря за данните /БД/

**V**iew - представянето на данните  
/графичния интерфейс/

**C**ontroller - контролира бизнес  
логиката на програмата /кое,  
кога и как ще бъде  
представено и т. н./



# Codeigniter framework

# Codeigniter

## Framework - ?

### Codeigniter

- [Документация](#)
- [Кратко представяне](#)
- [Download CI](#)
- [Installing CI](#)

# CONTROLLERS

# controller

- extends CI\_Controller
- Методите
- uri:

**project/index.php/class1/method1**

**project/index.php/class3/method2**

**project/index.php/class4/method2**



# Controller - 2

Задача: Отпечатайте на екрана: CI is cool! 2-date if/else logic based on even/odd

```
class First extends CI_Controller{
    public function print1()
    {
        echo "CI is cool!";
    }
}
```

-><http://localhost/demo1/index.php/first/print1>

[CI tutorial - static pages](#)

# Параметри в контролера

# Параметри в контролера

*Задача: Да се отпечата на екрана “CI is \$noun!”, където \$noun се въвежда през url.*

```
public function print2($param)
{
    echo “CI is $param!”;
}
```

-><http://localhost/demo1/index.php/First/print2/strange>

**VIEWS**

# Views

- Служи за представяне на информация на потребителя
- .php файл, който съдържа основно HTML & CSS/php код, генериращ HTML/

view-файловете се зареждат чрез методи в контролера ->

```
$this->load->view('view_name');
```

# Views - 2

*Задача: Отпечатайте “My first VIEW”.*

Controllers/First.php

```
public function print_view()  
{  
    return $this->load->view('view1');  
}
```

Views/view1.php

```
<h2>  
    <?php echo "This is my first CI view!"; ?>  
</h2>
```

**Подаване на данни от  
Controller към View**

# Подаване на данни от C към V

C подават данни към V чрез асоциативен масив.

```
$this->load->view('test1', $data); /зарежда view-файл test1.php/,
```

```
$data['data_key1'] = .....
```

```
$data['data_key2'] = .....
```

```
*****
```



# Подаване на данни от С към V - 2

*Controllers/Second.php*

```
public function method()
```

```
{
```

```
    $data['title'] = 'Awsome weather in Vratsa!';
```

```
    $data['content'] = "Great sunday weather in Vratsa fot the end of oct.!!!";
```

```
    $this->load->view('news', $data);
```

```
}
```

*Views/news.php*

```
<?php
```

```
    echo "<h1>$title </h1>";
```

```
    echo "<p>$content </p>";
```

```
    echo 'footer';
```

```
?>
```

# CI Helpers

# CI Helpers

## *Задача*

1. Дефинирайте в контролера функция `load_demo()`, която зарежда view - `demo_links.php`. Във `demo_links.php` да има линк, който зарежда отново `load_demo()` от контролера, която отново ни връща в/зарежда `demo_links.php`.

# CI Helpers - 2

## Задача 2

1. Прочетете [CI Helpers](#).
2. Направете необходимите корекции, така че url да се зареждат коректно.

# CI Helpers - 3

## Задача 3

1. Отпечатайте снимки, описание и заглавие, под формата на таблица в `demo_pictures.php view`.
2. Направете бутон-брояч. При натискане на бутона чилото, отпечатано над него се увеличава с 1.
  - a. При натискане на бутона числото се увеличава с число по ваш избор, различно от 1.
  - b. Добавете бутон, при натискане на който, числото намалява с 1 /или с число по избор/.
3. При влизане в сайта, потребителят има възможност да избере - да прочете информация или да въвежда информация. /линкове към съответни `view` файлове/. Информацията е отпечатана под формата на няколко блока – заглавие и съдържание. Ако избере да въвежда информация - линка го отвежда на страница с форма за въвеждане. /Полетата да са поне две, по ваш избор/.