

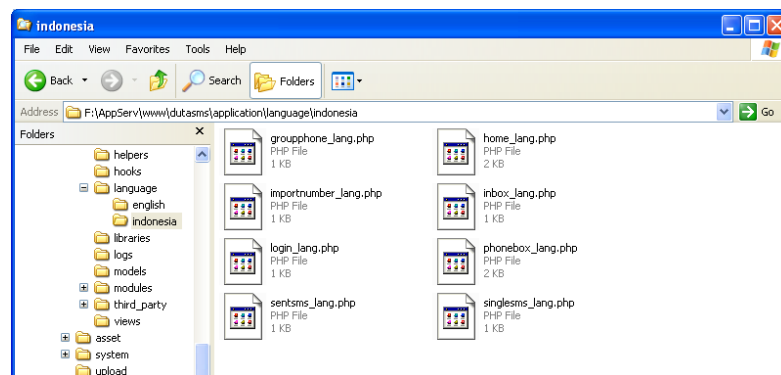
Tutorial Codeigniter : Membuat multiple language dengan Language Class

Pembaca yang budiman. Sering sekali kita melihat website dengan beberapa bahasa. Nah, kira kira bagaimana ya kalo kita membuat sendiri sebuah website dengan multiple language? Berikut akan saya tulis tutorial memanfaatkan language class milik Codeigniter untuk membuat multiple language. Monggih disimak.

1. Tentang Library Language Class

Sebetulnya sudah sangat lengkap sekali dijelaskan oleh Codeigniter lewat dokumen yang tersedia mengenai Language Class. Tapi tidak semua dari programmer paham bagaimana memanfaatkan library ini dalam membuat multiple language pada aplikasi kita.

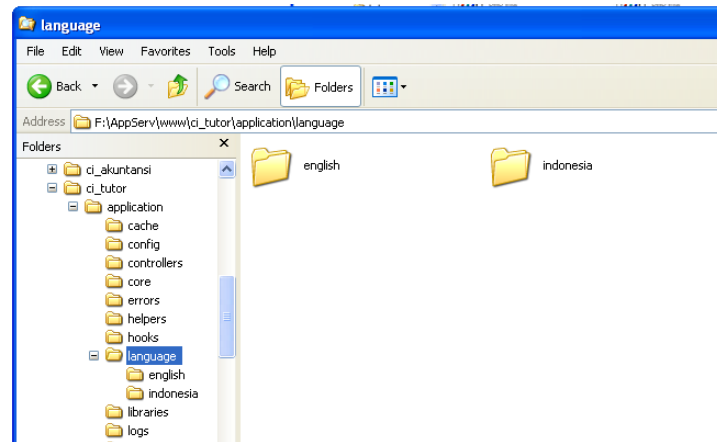
Library language class akan mengambil file yang berada pada direktori **application/language/bahasa/namafile_lang.php**, dan kemudian membaca baris teks yang ada pada file tersebut. Bahasa dimaksudkan untuk bahasa yang akan kita gunakan dan namafile_lang.php adalah kumpulan baris bahasa yang akan kita load pada aplikasi kita. Kurang lebih susunan file dalam folder language adalah sebagai berikut :



Gambar 1. Contoh susunan direktori language.

2. Membuat language file

Biar cepet dan tidak banyak basa –basi langsung saja kita membuat contoh aplikasi dengan multiple language. Disini saya akan mencoba membuat aplikasi sederhana dengan menggunakan language class. Oke pertama anda buat direktori **indonesia** pada direktori **application/language/**, seperti ini contohnya :



Gambar 2. sebuah direktori baru "indonesia"

Oke, kalo kita lihat ya, ada dua direktori disana. Default atau direktori bawaan dari Codeigniter adalah direktori "english". Nah, direktori **indonesia**, berfungsi sebagai tempat menampung file bahasa Indonesia yang akan kita gunakan dalam contoh kita kali ini. Kemudian setelah membuat folder bahasa indonesia, buat file baru selanjutnya beri nama file tersebut (misalkan ya) **latihan_lang.php** dan tuliskan source code berikut :

```
<?php

$lang['title'] = 'Latihan Bahasa';
$lang['content'] = 'Ini adalah latihan membuat multiple language
dengan Codeigniter ^_^';

?>
```

Kenapa ya harus ada `_lang.php` nya? Ya Codeigniter memberikan aturan ketika akan membuat sebuah file language, prefix `_lang` harus ada pada masing-masing file language. Dan penggunaan `$lang` merupakan aturan baku untuk menuliskan beberapa array language yang akan kita gunakan. Jelas kan? ^_^

3. Membuat Controller

Nah, setelah membuat file languagenya kita membuat controller. Disini akan dijelaskan juga bagaimana cara untuk mengambil baris text yang sudah kita buat pada file language tadi. Oke buat file baru bernama **bahasa.php**

```
<?php if(!defined('BASEPATH')) exit('Keluar dari sistem..!!!');

class Bahasa extends CI_Controller
{

    public function __construct()
    {
        parent::__construct();

        $this->load->language('latihan','indonesia');
    }

    public function index()
    {
        $data['judul'] = $this->lang->line('title');
        $data['isinya'] = $this->lang->line('content');

        $this->load->view('bahasa_v', $data);
    }
}

?>
```

4. Membuat file view

Ya, file view sangat penting karena disini lah (disini lah disinilah kita berjum lagi.. hehehe.. lagu pramuka yak... salam pramuka asolole jossssss..) kita akan menampilkan pada browser apa yang sudah kita buat. Buat file baru dan beri nama **bahasa_v.php**, berikut adalah source code nya :

```
<html>
<head>
    <title><?php echo $judul;?></title>
    <style>
    body{
        font-family:Calibri, Arial, Verdana;
    }
    </style>
</head>
<body>
    <h2><?php echo $judul;?></h2>
    <p><?php echo $isinya;?></p>
</body>
</html>
```

Kalo sudah semuanya coba eksekusi melalui browser, kalo punya saya saya akses seperti ini, http://localhost/ci_tutor/index.php/bahasa, jika tidak ada kesalahan anda akan mendapatkan tampilan berikut pada browser.



Latihan Bahasa

Ini adalah latihan membuat multiple language dengan Codeigniter ^_^

Gambar 3. Hasil pada browser.

Yups, seperti itu kurang lebih hasil dari beberapa kode tadi. Tapi kita belum puas untuk bereksperimen, kalo itu kan masih satu bahasa bagaimana kalo kita bikin tiga bahasa, apa aja ya bahasanya. Yang pertama jelas Indonesia, kedua bahasa Inggris dan yang ketiga adalah bahasa Tegal .. :D :D :D

5. Lebih lanjut dengan Language Class

Nah, biar lebih rame kita coba beberapa eksperimen lagi. Tambah combo box pada file **bahasa_v.php**, kurang lebih seperti ini kodenya :

```
<html>
<head>
  <title><?php echo $judul;?></title>
  <style>
    body{
      font-family:Calibri, Arial, Verdana;
    }
  </style>
</head>
<body>
  <form action="http://localhost/ci_tutor/index.php/bahasa/index"
name="ganti" id="ganti" method="post">
    Pilih Bahasa : <select name="bahasa"
onchange="this.form.submit()">
      <option value="indonesia">Indonesia</option>
      <option value="english">English</option>
      <option value="tegal">Bahasa Tegal</option>
    </select>
  </form>
  <h2><?php echo $judul;?></h2>
  <p><?php echo $isinya;?></p>
</body>
</html>
```

Kemudian ubah juga pada controller-nya yaitu bahasa.php, kurang lebih jadi seperti ini :

```
<?php if(!defined('BASEPATH')) exit('Keluar dari sistem..!!!');

class Bahasa extends CI_Controller
{

    public function __construct()
    {
        parent::__construct();
    }

    public function index()
    {
        $bahasa = $this->input->post('bahasa');

        $this->load->language('latihan', $bahasa);

        $data['judul'] = $this->lang->line('title');
        $data['isinya'] = $this->lang->line('content');

        $this->load->view('bahasa_v', $data);
    }
}

?>
```

Kemudian buat direktori baru, yaitu Tegal. Karena English dan Indonesia sudah ada sebelumnya. Copy file latihan_lang.php pada masing-masing folder. Dan ubah pada direktori english menjadi seperti ini :

```
<?php

$lang['title'] = 'Language Exercise';
$lang['content'] = 'This is practice using Language Class in
Codeigniter. Its so easy and simple, right?';

?>
```

Dan file latihan_lang.php yang berada pada direktori tegal, ganti menjadi :

```
<?php

$lang['title'] = 'Latihan nganggo Bahasa Jon';
$lang['content'] = 'Kyehe carane nganggo Language Class goneng
Codeigniter, gampang men ow ya.. sih ndarani angel jon... nang kene
kyeh..';

?>
```

Oke sekarang coba refresh browsernya maka kurang lebih akan menampilkan tampilan sebagai berikut :



Gambar 4. Multiple language (tiga bahasa)

6. Penjelasan singkat

Nah, ada beberapa hal yang harus menjadi perhatian yaitu :

```
$this->load->language('latihan','indonesia');
```

```
$this->lang->line('title');
```

Ya, untuk memanggil file language kita menggunakan `$this->load->language(nama_file, bahasa);` dan untuk memanggil masing-masing array pada file language kita gunakan `$this->lang->line('nama_array');` sebagai contohnya tadi kita load `$lang['title']` maka cara penggunaannya adalah `$this->lang->line('title');`

Demikian tutorial mengenai penggunaan Language Class pada Codeigniter semoga bermanfaat.



Tentang Penulis

Dimas Edu Prasada (Dimas Edubuntu Samid) / Edu, lahir di Kota Tegal, 27 Januari 1988. Bujangan yang hobi dengan music koplo ini , masih aktif sebagai praktisi IT. Menggemari “mazhab” web based application dan PHP menjadi bahasa yang paling disukainya.

Setelah sekian lama berkecimpung di dunia IT saat ini saya ingin sekali mewujudkan cita – cita yaitu memiliki sekolah gratis untuk anak-anak yang tidak mampu. Namun, saat ini masih belum bisa mewujudkannya semoga dikemudian hari dapat terwujud keinginan tersebut. Mohon doanya ya.... 😊

Saat ini penulis masih aktif mengajar di STMIK Tasimalaya serta pada LPK Mita College sebagai pengajar bahasa pemrograman. Semoga apa yang saya tulis dapat bermanfaat untuk anda semua, seperti quote yang menjadi prinsip hidup penulis, “Sebaik - baiknya orang adalah mereka yang bermanfaat untuk orang lain.”

Penulis dapat dihubungi melalui :

HP : 0857-42100-454

Email : edudimas1@gmail.com

FB : <http://facebook.com/eduaying>

Twitter : [@edu_aying](https://twitter.com/edu_aying)

Blog / URL : samidcorner.blogspot.com

DIMASEDU NET - Software House

Office Jl. Sangir No. 18 Kota Tegal