

Home (<https://www.codexworld.com>) » CodeIgniter (<https://www.codexworld.com/tutorials/codeigniter/>)
» How to Generate SEO Friendly URL in CodeIgniter

How to Generate SEO Friendly URL in CodeIgniter

By: CodexWorld | In: CodeIgniter (<https://www.codexworld.com/tutorials/codeigniter/>) |

Last Updated: Jul 18, 2017

Share

Tweet

SEO friendly URL structure is a common technique for search engine optimization. The SEO friendly URL holds the keywords which help to improve the search engine ranking. The difference between normal URL and SEO friendly URL is like the following.

- Normal URL: <http://www.example.com/posts.php?id=123>
- SEO Friendly URL: <http://www.example.com/codeigniter-tutorial-for-beginners>

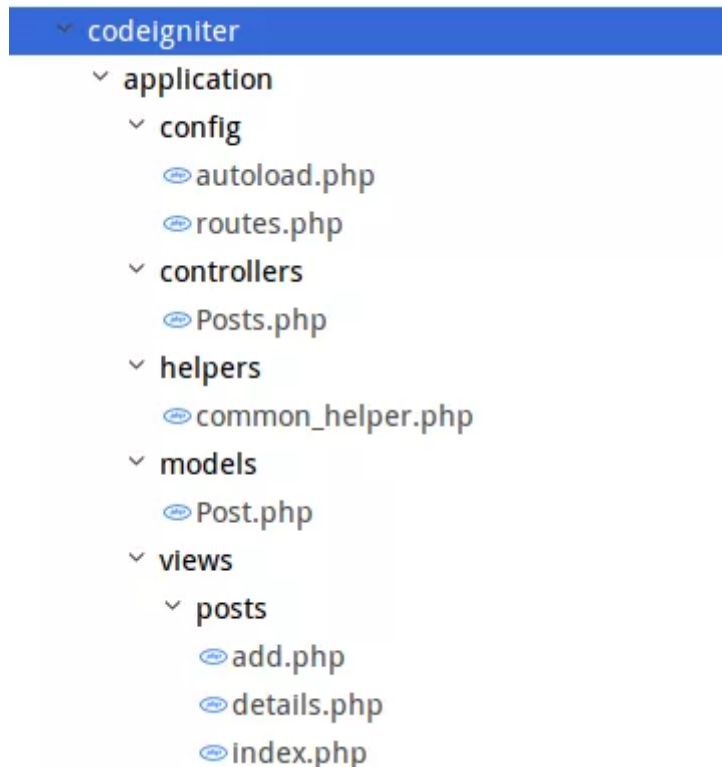
The above example clearly shows that the SEO friendly URL is more human-friendly than the normal URL. It is always a good idea to make website's URL SEO friendly for getting more traffic from search engines.

If your website is built with CodeIgniter and you want to create SEO friendly URL (<https://www.codexworld.com/generate-seo-friendly-url-from-title-string-php/>) for the dynamic contents (posts, products, pages, etc.) of the website, there is an easy way for doing that. In this tutorial, we will show you how to **generate SEO friendly URL in CodeIgniter** and store the dynamic URL in the database.

Here we will demonstrate the complete SEO friendly URL implementation process by posts or articles creation in CodeIgniter. The following functionalities will be implemented in our example script.

- Generate unique URL slug from the post title.
- Store URL slug in the database for each post.
- Get SEO URL slug from the database and specify the dynamic URL in posts listing.
- Display post content by SEO URL using CodeIgniter Routes.

Before you get started, take a look at the files structure of CodeIgniter SEO friendly URL implementation.



Database Table Creation

To store posts data a table needs to be created in the database. The following SQL creates a `posts` table with some basic fields including SEO URL field in the database.

```
CREATE TABLE `posts` (  
  `id` int(11) NOT NULL AUTO_INCREMENT,  
  `url_slug` varchar(255) COLLATE utf8_unicode_ci NOT NULL,  
  `title` varchar(255) COLLATE utf8_unicode_ci NOT NULL,  
  `content` text COLLATE utf8_unicode_ci NOT NULL,  
  `created` datetime NOT NULL,  
  `modified` datetime NOT NULL,  
  `status` tinyint(1) NOT NULL DEFAULT '1' COMMENT '1=Active, 0=Inactive',  
  PRIMARY KEY (`id`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;
```

Controller (Posts.php)

The Posts controller contains three methods, `index()`, `details()` and `add()`.

- **index()** – Gets all the posts data by the Post model and sends to the list view.
- **details()** – Gets the post data based on the `url_slug` provided in the URL and sends to the details view.
- **add()** – Generates unique SEO URL (`url_slug`) and adds the new post to the database.
 - The clean, unique and SEO friendly URL slug will be generated from the post title.
 - The CodeIgniter's `url_title()` is used to convert post title to URL slug.
 - The custom helper function `isUrlExists()` helps to check whether the URL already used in `url_slug`. If already exists, time stamp suffix will be added to the URL slug.

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');  
/**  
 * Posts Management class created by CodexWorld  
 */  
class Posts extends CI_Controller {  
  
    function __construct() {  
        parent::__construct();  
        $this->load->model('post');  
        $this->load->helper('common');  
        $this->load->library('form_validation');  
    }  
  
    public function index(){  
        $data = array();  
  
        //get the posts data  
        $data['posts'] = $this->post->getRows(array('limit'=>10));  
  
        //load the view  
        $this->load->view('posts/index', $data);  
    }  
  
    public function details($url_slug){  
        $data = array();  
  
        //get the post data  
        $data['post'] = $this->post->getRows(array('url_slug'=>$url_slug));  
  
        //load the view  
        $this->load->view('posts/details', $data);  
    }  
  
    public function add(){  
        $data = array();
```

```
$postData = array();
if($this->input->post('submitBtn')){
    $this->form_validation->set_rules('title', 'Title', 'required');
    $this->form_validation->set_rules('content', 'Content', 'required');

    $postData = array(
        'title' => strip_tags($this->input->post('title')),
        'content' => strip_tags($this->input->post('content'))
    );

    if($this->form_validation->run() == true) {
        /*
         * Generate SEO friendly URL
         */
        $title = strip_tags($this->input->post('title'));
        $titleURL = strtolower(url_title($title));
        if(isUrlExists('posts',$titleURL)){
            $titleURL = $titleURL.'-'.time();
        }
        $postData['url_slug'] = $titleURL;

        //Insert post data to database
        $insert = $this->post->insert($postData);
        if($insert){
            $postData = array();
            $data['success_msg'] = 'Post data inserted successfully.';
        } else {
            $data['error_msg'] = 'Some problems occurred, please try again.';
        }
    }
}

$data['post'] = $postData;

//load the view
```

```
        $this->load->view('posts/add', $data, false);  
    }  
}
```

Model (Post.php)

The Post model helps to fetch records from the database and insert records in the database. It contains two methods, `getRows()` and `insert()`.

- **getRows()** – Fetches data from the `posts` table and returns as an array.
- **insert()** – Insert data into the `posts` table and returns the insert ID.

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');
class Post extends CI_Model{
    /*
     * get rows from the posts table
     */
    function getRows($params = array()){
        $this->db->select('*');
        $this->db->from('posts');

        //set start and limit
        if(array_key_exists("start",$params) && array_key_exists("limit",$params)){
            $this->db->limit($params['limit'],$params['start']);
        }elseif(!array_key_exists("start",$params) && array_key_exists("limit",$params)){
            $this->db->limit($params['limit']);
        }

        if(array_key_exists("url_slug", $params)){
            $this->db->where('url_slug', $params['url_slug']);
            $query = $this->db->get();
            $result = ($query->num_rows() > 0)?$query->row_array():FALSE;
        }else{
            $query = $this->db->get();
            $result = ($query->num_rows() > 0)?$query->result_array():FALSE;
        }

        //return fetched data
        return $result;
    }

    /*
     * Insert post data to table
     */
    public function insert($data = array()) {
        if(!array_key_exists("created", $data)){
            $data['created'] = date("Y-m-d H:i:s");
        }
    }
}
```

```
}  
if(!array_key_exists("modified", $data)){  
    $data['modified'] = date("Y-m-d H:i:s");  
}  
  
//insert data  
$insert = $this->db->insert('posts', $data);  
if($insert){  
    return $this->db->insert_id();;  
}else{  
    return false;  
}  
}  
  
}
```

View (posts)

The view directory (posts/) contains three files, `index.php`, `details.php`, and `add.php`.

posts/index.php:

This view file is loaded by the `index()` method of the Posts controller. All the posts are listed in the index view with details link. The dynamic SEO URL slug is used in the hyperlink.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Generate SEO Friendly URL in CodeIgniter</title>
</head>
<body>
<div class="container">
    <h1>Post List</h1>
    <div class="row">
        <ul class="post-list">
            <?php if(!empty($posts)): foreach($posts as $post): ?>
                <li><a href="<?php echo base_url('post/'.$post['url_slug']); ?>"><?php ech
o $post['title']; ?></a></li>
            <?php endforeach; else: ?>
                <p>Post(s) not available.</p>
            <?php endif; ?>
        </ul>
    </div>
</div>
</body>
</html>
```

posts/details.php:

This view file is loaded by the `details()` method of the Posts controller. The requested post information are displayed in details view.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <title>Generate SEO Friendly URL in CodeIgniter</title>
</head>
<body>
<div class="container">
    <h1><?php echo $post['title']; ?></h1>
    <p><?php echo $post['content']; ?></p>
</div>
</body>
</html>
```

posts/add.php:

This view file is loaded by the `add()` method of the Posts controller. Initially, a form is displayed to create a new post. Once the form is submitted, the given information is inserted into the database and the status message is shown.

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Generate SEO Friendly URL in CodeIgniter</title>
</head>
<body>
<div class="container">
  <h1>Add Post</h1>
  <div class="row">
    <?php echo !empty($success_msg)?'<p>'.$success_msg.'</p>':''; ?>
    <?php echo !empty($error_msg)?'<p>'.$error_msg.'</p>':''; ?>
    <form method="post">
      <p>Post Title: <input type="text" name="title" value="<?php echo !empty($post[
'title'])?$post['title']:''; ?>"></p>
      <?php echo form_error('title','<p class="help-block error">','</p>'); ?>

      <p>Post Content: <textarea name="content"><?php echo !empty($post['content'])?
$post['content']:''; ?></textarea></p>
      <?php echo form_error('content','<p class="help-block error">','</p>'); ?>

      <input type="submit" name="submitBtn" value="Add">
    </form>
  </div>
</div>
</body>
</html>
```

Helper (common_helper.php)

The custom helper function named `isUrlExists()` is defined in this file. It helps to check whether the URL slug already exists in the posts table.

```
<?php if ( ! defined('BASEPATH')) exit('No direct script access allowed');  
/*  
 * Helper function to check whether the url slug already exists  
 */  
if(!function_exists('isUrlExists')){  
    function isUrlExists($tblName, $urlSlug){  
        if(!empty($tblName) && !empty($urlSlug)){  
            $ci = & get_instance();  
            $ci->db->from($tblName);  
            $ci->db->where('url_slug',$urlSlug);  
            $rowNum = $ci->db->count_all_results();  
            return ($rowNum>0)?true:false;  
        }else{  
            return true;  
        }  
    }  
}
```

Config

autoload.php:

Specify some required libraries and helpers in `autoload.php` to loads automatically.

```
/*
| -----
|   Auto-load Libraries
| -----
*/
$autoload['libraries'] = array('database');

/*
| -----
|   Auto-load Helper Files
| -----
*/
$autoload['helper'] = array('url', 'form');
```

routes.php:

To load the post details based on the SEO URL slug, a routing rule needs to be specified. As per the following routing rule, if URI contains SEO URL slug, the `details()` method of the Posts controller would be loaded.

```
//specify route for post details page
$route['post/(:any)'] = 'posts/details/$1';
```

Conclusion

You can follow this process to create SEO friendly URL for dynamic contents of your CodeIgniter application. Hope! this simple process helps to switch to the SEO friendly URL and make your CodeIgniter application SEO friendly. Use CodeIgniter Routes to change the URL format as per your requirement.

SEE ALSO:See Also:

How to Generate SEO Friendly URL in PHP (<https://www.codexworld.com/generate-seo-friendly-url-from-title-string-php/>)

Are you want to get implementation help, or modify or extend the functionality of this script? [Submit paid service request](#)



Download Source Code

(<https://www.codexworld.com/downloads/generate-seo-friendly-url-codeigniter/>)