BihongoPHP User Guide

**Installation**

* [Install with bihongoCMD](#installwithbihongocmd)
* [Install manually](#installwithmanually)

**Tutorial: Create a simple blog site**

* [Create controller](#Createacontroller)
* [Set up database](#Setupdatabase)
* [Create model](#Createmodel)
* [Setup route](#Setuproute)
* [Create html view file](#Createhtmlviewfile)
* [Conclusion](#Conclusion)

**General Tropics**

* Controllers
* Views
* Models
* Helpers
* Library
* Creating Library
* Security

**Install with bihongoCMD**

BihongoCMD is package manager application for using in Bihongo Web Framework. Using bihongoCMD in bihongo Project is recommend. But without bihongoCMD you can continue using bihongo.

If you using windows OS then you can download bihongoCMD from github.

Link: <https://github.com/SojebSikder/bihongocmd/releases/download/v1.0/bihongoCMD.zip>

Note: Need Dot Net Framework 4.5 to use bihongoCMD

After download bihongoCMD first setup Environment path to use bihongoCMD anywhere on directory. If you don’t know how to setup Environment path you can google it like: how to setup environment path on windows.

Let’s assume you install bihongoCMD properly, now we install bihongoPHP using bihongoCMD. Open a folder and open Command Prompt from this folder. Now write a command:

phpcmd install:SojebSikder/bihongophp-yourprojectname

This will create our bihongo project

Note: don’t use any (-) dash sign for naming project

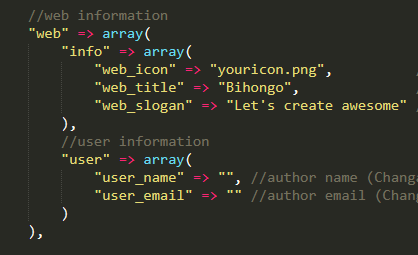
**Install manually**

To install manually go to bihongoPHP github page and download (clone) following repository:

<https://github.com/SojebSikder/bihongophp>

Extract files and put all files where you wish.

After install bihongoPHP we have to configure first. Go to config->config.php. Now look $config array and change web\_icon, web\_title, web\_slogan, user\_name, user\_email as your requirement.



Also change Base Url. Change baseUrl to your project url.



Remember don’t change anything on config.php which don’t have (changeable) comment

Now run your project. Mine is <http://localhost/yourprojectname>

**Tutorial: Create a simple blog site**

If you are a ‘Learn by doing’ type then we will create a project using bihongoPHP.

We will create a simple blog app.

Create a controller:

First we will create controller. Go to this folder app->controllers. And create a file and name it whatever you want but Controller prefix must need to have. class name and file name should be same.

Syntax: NameController.php

After creating file and add following code:

<?php

class BlogController extends Controller{

public function \_\_construct(){

parent::\_\_construct();

}

public function home(){

$this->load->view("home");

}

}

?>

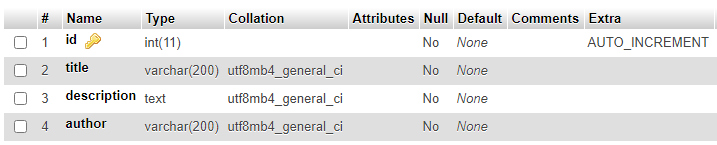
You can create controller easily with bihongoCMD. Look at this command:

phpcmd make:controller-BlogController

Note: Remember that Class name start with Capital Letter. And Controller prefix have to be exist. We can see there is a home() method.

Set up database:

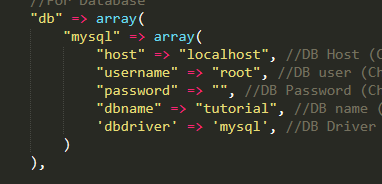
We will make simple database for our tutorial purpose. Let’s use mysql database.



Our database have 4 column id, title, description and author. Now add some values:



After creating our database we have to setup dbname on our config. Go to config/config.php change dbname, host, user, password.



Create model:

Go to this folder app->models and create a file ModelnameModel.php.

Model have Model prefix . Don’t forget to add Model prefix. Class name and file name should be same.

Syntax: NameModel

After creating file add following code:

<?php

class BlogModel extends Model{

public function \_\_construct(){

parent::\_\_construct();

}

}

?>

Again you can create model using bihongoCMD. Look at this command:

phpcmd make:model-BlogModel

**So you created your Model**. Now what we do, we have to get blog from our database. So add follow code on BlogModel.php:

<?php

class BlogModel extends Model{

public function \_\_construct(){

parent::\_\_construct();

}

public function getBlog(){

return $this->db->select("SELECT \* FROM blog");

}

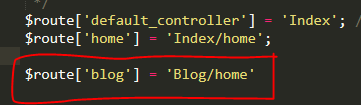
}

?>

Setup route:

Now we setup route. Go to this folder config->routes.php

And add route name as your requirements. I will add blog.



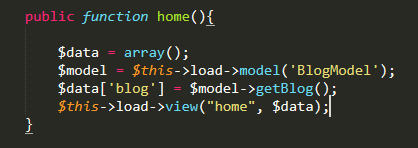
We can see Blog/home. Blog is Controller name and home is method name. If we not put method name then home() method invoke automatically. Now if you run at this url: http://localhost/yourprojectname/blog then you will see home page open.

Create html view file:

I will let html design part on you. Our html view file will keep on this folder

app->views .

Go to BlogController add following code:



We defined $data variable as array. And $model variable for model. For loading model we have to use

$this->load->model(‘BlogModel’);

This create a object for IndexModel and store in $model variable.

In $data[‘blog’] we call getBlog() method.

And by this code

$this->load->view(‘home’, $data);

will render home page on browser. Second parameter is for transfer data, so we place $data variable for transfer $data on home page. Now create a home.php file on this folder app->views. Add following code (You can design your page as your requirements) :

<!DOCTYPE html>

<html>

<head>

    <base href="<?php echo ROOT ?>">

    <meta charset="UTF-8">

    <link rel="icon" href="<?php echo ICON ?>" type="image/png" sizes="16x16">

    <meta name="description" content="">

    <meta name="keywords" content="">

    <meta name="author" content="">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <meta http-equiv="X-UA-Compatible" content="ie=edge">

    <title>Welcome to <?php echo TITLE ?></title>

</head>

<body>

<?php foreach($blog as $getBlog): ?>

    <h3><?php echo $getBlog['title'] ?></h3>

    <p><?php echo $getBlog['description'] ?></p>

    <hr>

    <?php endforeach ?>

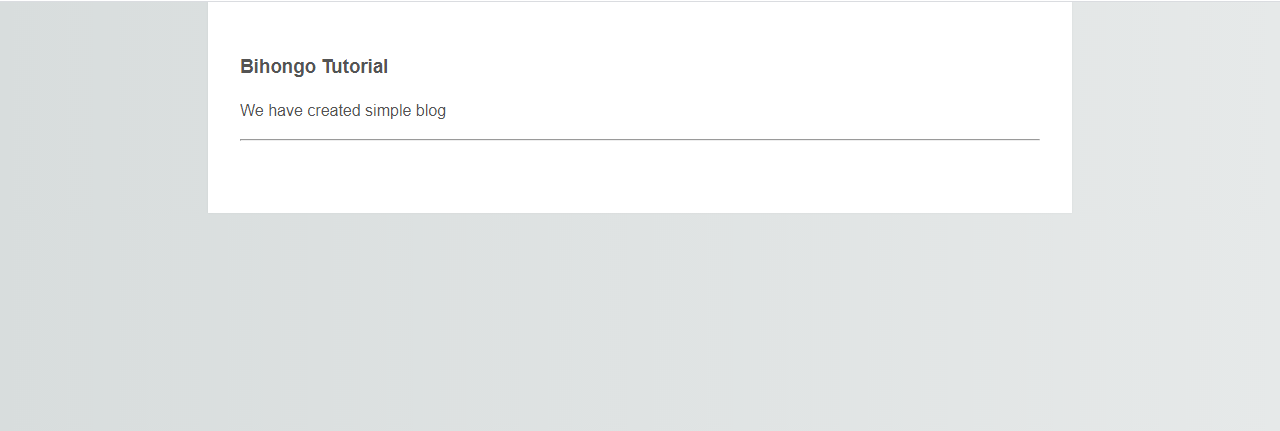
</body>

</html>

Here you can see many constant. These are predefined constant by BihongoPHP. Such as ROOT constant return app root url. Icon, TITLE were return icon url and title name.

Now run your app on this address : <http://localhost/yourprojectname/blog>

You will see this result:



Conclusion:

We done a very simple project for learning purpose. Now you have a simple knowledge about BihongoPHP Framework. Now the question is what is next?

To know more about BihongoPHP go to general tropics page. Happy Coding ☺

**General Tropics**

Controllers:

Now we will learn about controller.

Controllers are the heart of your application, as they determine how HTTP requests should be handled.

Controller is class file and used in url. BihongoPHP controller look like this:

<?php

class IndexController extends Controller{

    public function \_\_construct(){

        parent::\_\_construct();

    }

    public function home(){

        $this->load->view("home", $data);

    }

}

?>

Note: Class names must start with an uppercase letter.