



Core Concepts & Configuration

MVC Structure

Model: Represents data and business logic.
View: Handles the presentation of data to the user (HTML, etc.).
Controller: Manages requests, interacts with models, and loads views.
Directory Structure: <code>app</code> , <code>system</code> , <code>public</code> , <code>writable</code>
<code>app</code> - Application specific code (controllers, models, views, etc.)
<code>system</code> - Core CodeIgniter files (do not modify).
<code>public</code> - Contains the <code>index.php</code> file and assets (CSS, JS, images).
<code>writable</code> - Directory for logs, cache, and uploads.

Configuration (app/Config/App.php)

<code>\$baseUrl</code>	The base URL of your application (e.g., <code>http://localhost:8080</code>)
<code>\$indexPath</code>	The name of your index page (usually <code>index.php</code> or empty string to remove it.)
<code>\$encryptionKey</code>	Encryption key for secure sessions and data.
<code>\$defaultLocale</code>	The default locale for your application (e.g., <code>en</code> , <code>fr</code>).
<code>\$sessionDriver</code>	The session driver to use (<code>FileHandler</code> , <code>DatabaseHandler</code> , etc.).
<code>\$timezone</code>	The timezone for your application (e.g., <code>America/Los_Angeles</code>).

Controllers & Routing

Creating a Controller

Create a file in <code>app/Controllers/</code> (e.g., <code>MyController.php</code>)
<pre><?php namespace App\Controllers; use CodeIgniter\Controller; class MyController extends Controller { public function index() { return 'Hello, World!'; } }</pre>
Access the controller via URL (e.g., <code>http://localhost:8080/mycontroller</code>)

Routing

<code>app/Config/Routes.php</code>	Configuration file for defining routes.
<code>\$routes->get('path', 'Controller::method');</code>	Defines a GET route.
<code>\$routes->post('path', 'Controller::method');</code>	Defines a POST route.
<code>\$routes->resource('photos');</code>	Creates RESTful resource routes for a controller named 'Photos'.
Route Parameters	<code>\$routes->get('users/{:num}', 'Users::show');</code>
Auto Route (Improved)	Set <code>\$routes->setAutoRoute(true);</code> to automatically route based on controller and method names.

Request Object

Accessing the request object in a controller:
<pre>\$request = \Config\Services::request(); \$data = \$request->getPost(); // Get POST data \$data = \$request->getGet(); // Get GET data</pre>
<pre>\$userAgent = \$request->getUserAgent(); \$ipAddress = \$request->getIPAddress();</pre>

Models & Database

Creating a Model

Create a file in `app/Models/` (e.g., `UserModel.php`)

```
<?php

namespace App\Models;

use CodeIgniter\Model;

class UserModel extends Model
{
    protected $table = 'users';
    protected $primaryKey = 'id';

    protected $useAutoIncrement = true;

    protected $returnType = 'array';
    protected $useSoftDeletes = true;

    protected $allowedFields = ['name',
'email'];

    protected $useTimestamps = true;
    protected $createdField = 'created_at';
    protected $updatedField = 'updated_at';
    protected $deletedField = 'deleted_at';

    protected $validationRules = [];
    protected $validationMessages = [];
    protected $skipValidation = false;
}
```

Views & Templating

Creating a View

Create a file in `app/Views/` (e.g.,
`welcome_message.php`)

```
<!DOCTYPE html>
<html>
<head>
    <title>Welcome</title>
</head>
<body>
    <h1>Hello, <?php echo $name; ?>!</h1>
</body>
</html>
```

Database Configuration

(`app/Config/Database.php`)

```
$default Default database connection settings.
t

hostname Database server hostname (e.g.,
e localhost ).

username Database username.
e

password Database password.
d

database Database name.
e

DBDriver Database driver (e.g., MySQLi , Postgre ,
r SQLite3 ).
```

Database Queries

```
$db = \Config\Database::connect();

// Simple query
$query = $db->query('SELECT * FROM users');

foreach ($query->getResult() as $row) {
    echo $row->name;
}

// Using Query Builder
$builder = $db->table('users');
$query = $builder->select('id, name, email');
$query = $builder->get();

// Insert data
$data = [
    'name' => 'John Doe',
    'email' => 'john.doe@example.com',
];
$builder->insert($data);

// Update data
$data = ['email' => 'new.email@example.com'];
$builder->where('id', 1);
$builder->update($data);
```

Creating a View

Create a file in `app/Views/` (e.g.,
`welcome_message.php`)

```
<!DOCTYPE html>
<html>
<head>
    <title>Welcome</title>
</head>
<body>
    <h1>Hello, <?php echo $name; ?>!</h1>
</body>
</html>
```

Loading Views in Controller

```
<?php

namespace App\Controllers;

use CodeIgniter\Controller;

class Home extends Controller
{
    public function index()
    {
        $data = ['name' => 'CodeIgniter'];
        return view('welcome_message', $data);
    }
}
```

Templating with View Layouts

Create a layout file (e.g.,
`app/Views/layouts/default.php`):

```
<!DOCTYPE html>
<html>
<head>
    <title><?php echo $title; ?></title>
</head>
<body>
    <?= $this->renderSection('content') ?>
</body>
</html>
```

Create a view that extends the layout (e.g.,
`app/Views/home.php`):

```
<?= $this->extend('layouts/default') ?>

<?= $this->section('content') ?>
    <h1>Welcome Home!</h1>
<?= $this->endSection() ?>
```

In Controller:

```
public function index()
{
    $data['title'] = 'Home Page';
    return view('home', $data);
}
```