CODEXWORLD

Hosting wit

Hosting + Free Dom
Bandwidth.

in.godaddy.com

HOME /   OGIN SYSTEM IN CODEIGNITER

User   gin System in
Code

By: Codex   22, 2019

Share

n CodeI...

User Login System is the most used feature in CodeIgniter application. Like the user authentication in PHP, the user login system can be easily implemented in CodeIgniter with the session. CodeIgniter framework provides many built-in libraries and helpers which helps to integrate registration and login functionality with MySQL database.

The **login system** is very useful to restrict access to your application. This feature allows only the logged-in user to access web pages of your web application. You can easily implement the user authentication system (registration and login) with the session in CodeIgniter application. In this step-by-step tutorial, we will show you how to build a **login system in CodeIgniter** with SESSION and MySQL database.

Hope you have already know the configuration process of the CodeIgniter framework. If you are new to CodeIgniter, we suggest to see this CodeIgniter beginners guide first – CodeIgniter Beginners Guide with Configuration and Setup

The following functionality will be implemented in the example **CodeIgniter Login System** script.

- The user registration form to provide the account information.
- Insert account data into the MySQL database.
- The user login form for sign in.
- Login and registration form validation.
- Existing email check during registration.
- Store logged-in user identification into session.
- Display user account details after login.
- Logout the user from their account.

Before you begin to implement user registration and login system in CodeIgniter, take a look at the files structure.

```
codeigniter_login_system/
├── application/
│   ├── controllers/
│   │   └── Users.php
│   ├── models/
```

```
    |       ├──  elements/
    |       |       ├──  header.php
    |       |       └──  footer.php
    |       └──  users/
    |               ├──  registration.php
    |               ├──  login.php
    |               └──  account.php
    └──  assets/
            └──  css/
                    └──  style.css
```

Create Database Table

To store the user account information, a table is required in the database. The following SQL creates a `users` table with some basic required fields in the MySQL database.

```sql
CREATE TABLE `users` (
 `id` int(11) NOT NULL AUTO_INCREMENT,
 `first_name` varchar(25) COLLATE utf8_unicode_ci NOT NULL,
 `last_name` varchar(25) COLLATE utf8_unicode_ci NOT NULL,
 `email` varchar(50) COLLATE utf8_unicode_ci NOT NULL,
 `password` varchar(200) COLLATE utf8_unicode_ci NOT NULL,
 `gender` enum('Male','Female') COLLATE utf8_unicode_ci NOT NULL,
 `phone` varchar(15) 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;
```

Config (autoload.php)

In this CodeIgniter Login System script, the built-in system library and helper are used. Define the frequently used libraries and helpers in the `application/config/autoload.php` file.

```php
// Auto-load Libraries
$autoload['libraries'] = array('database', 'session');

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

The Users controller handles the user registration and authentication-related operations.

- __construct() –
  - Load the Form validation library and User model.
  - Get the user's login status from SESSION.

- **index()** – Based on the login status redirects the user to the account/login page.
- **account()** –
  - Get the logged-in user's account information using `getRows()` method of User model.
  - Pass user data and load the account details view.

- **login()** –
  - Initially, load the login form view.
  - If the form is submitted,
    - Validate the posted fields data using CodeIgniter Form Validation library.
    - Check whether the user exists in the database with the provided login credentials.
    - Based on the login status redirects the user to account/login page.

- **registration()** –
  - Initially, load the registration form view.
  - If the form is submitted,
    - Validate the posted fields data using CodeIgniter Form Validation library.
    - Check whether the provided email already exists in the database using a custom callback function (`email_check`).
    - Insert the user's account information in the database using `insert()` method of the User model.

- **logout()** – Log the user out from their account.
- **email_check()** – Custom callback function used with CodeIgniter form validation library to check the existing email.

```php
<?php defined('BASEPATH') OR exit('No direct script access allowed');

class Users extends CI_Controller {

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

        // Load form validation ibrary & user model
        $this->load->library('form_validation');
```

```php
        // User login status
        $this->isUserLoggedIn = $this->session->userdata('isUserLoggedIn');
    }

    public function index(){
        if($this->isUserLoggedIn){
            redirect('users/account');
        }else{
            redirect('users/login');
        }
    }

    public function account(){
        $data = array();
        if($this->isUserLoggedIn){
            $con = array(
                'id' => $this->session->userdata('userId')
            );
            $data['user'] = $this->user->getRows($con);

            // Pass the user data and load view
            $this->load->view('elements/header', $data);
            $this->load->view('users/account', $data);
            $this->load->view('elements/footer');
        }else{
            redirect('users/login');
        }
    }

    public function login(){
        $data = array();

        // Get messages from the session
        if($this->session->userdata('success_msg')){
            $data['success_msg'] = $this->session->userdata('success_msg');
            $this->session->unset_userdata('success_msg');
        }
        if($this->session->userdata('error_msg')){
            $data['error_msg'] = $this->session->userdata('error_msg');
            $this->session->unset_userdata('error_msg');
        }

        // If login request submitted
        if($this->input->post('loginSubmit')){
```

```php
        $this->form_validation->set_rules('password', 'password', 'require
d');

        if($this->form_validation->run() == true){
            $con = array(
                'returnType' => 'single',
                'conditions' => array(
                    'email'=> $this->input->post('email'),
                    'password' => md5($this->input->post('password')),
                    'status' => 1
                )
            );
            $checkLogin = $this->user->getRows($con);
            if($checkLogin){
                $this->session->set_userdata('isUserLoggedIn', TRUE);
                $this->session->set_userdata('userId', $checkLogin['id']);
                redirect('users/account/');
            }else{
                $data['error_msg'] = 'Wrong email or password, please tr
y again.';
            }
        }else{
            $data['error_msg'] = 'Please fill all the mandatory fields.';
        }
    }

    // Load view
    $this->load->view('elements/header', $data);
    $this->load->view('users/login', $data);
    $this->load->view('elements/footer');
}

public function registration(){
    $data = $userData = array();

    // If registration request is submitted
    if($this->input->post('signupSubmit')){
        $this->form_validation->set_rules('first_name', 'First Name', 'req
uired');
        $this->form_validation->set_rules('last_name', 'Last Name', 'requi
red');
        $this->form_validation->set_rules('email', 'Email', 'require
d_email|callback_email_check');
        $this->form_validation->set_rules('password', 'password', 'require
```

```php
rd', 'required|matches[password]');

        $userData = array(
            'first_name' => strip_tags($this->input->post('first_name')),
            'last_name' => strip_tags($this->input->post('last_name')),
            'email' => strip_tags($this->input->post('email')),
            'password' => md5($this->input->post('password')),
            'gender' => $this->input->post('gender'),
            'phone' => strip_tags($this->input->post('phone'))
        );

        if($this->form_validation->run() == true){
            $insert = $this->user->insert($userData);
            if($insert){
                $this->session->set_userdata('success_msg', 'Your account registration has been successful. Please login to your account.');
                redirect('users/login');
            }else{
                $data['error_msg'] = 'Some problems occured, please try again.';
            }
        }else{
            $data['error_msg'] = 'Please fill all the mandatory fields.';
        }
    }

    // Posted data
    $data['user'] = $userData;

    // Load view
    $this->load->view('elements/header', $data);
    $this->load->view('users/registration', $data);
    $this->load->view('elements/footer');
}

public function logout(){
    $this->session->unset_userdata('isUserLoggedIn');
    $this->session->unset_userdata('userId');
    $this->session->sess_destroy();
    redirect('users/login/');
}

// Existing email check during validation
```

```php
            'returnType' => 'count',
            'conditions' => array(
                'email' => $str
            )
        );
        $checkEmail = $this->user->getRows($con);
        if($checkEmail > 0){
            $this->form_validation->set_message('email_check', 'The given emai
l already exists.');
            return FALSE;
        }else{
            return TRUE;
        }
    }
}
```

Model (User.php)

The User model handles the database related operations (fetch and insert).

- **getRows()** – Fetch the user data from the database based on the conditions.

- **insert()** – Insert the user account data in the database.

```php
<?php defined('BASEPATH') OR exit('No direct script access allowed');

class User extends CI_Model{
    function __construct() {
        // Set table name
        $this->table = 'users';
    }

    /*
     * Fetch user data from the database
     * @param array filter data based on the passed parameters
     */
    function getRows($params = array()){
        $this->db->select('*');
        $this->db->from($this->table);

        if(array_key_exists("conditions", $params)){
            foreach($params['conditions'] as $key => $val){
                $this->db->where($key, $val);
            }
```

☰ **CODEXWORLD**   🔍

```php
        if(array_key_exists("returnType",$params) && $params['returnType'] =
= 'count'){
            $result = $this->db->count_all_results();
        }else{
            if(array_key_exists("id", $params) || $params['returnType'] == 'si
ngle'){
                if(!empty($params['id'])){
                    $this->db->where('id', $params['id']);
                }
                $query = $this->db->get();
                $result = $query->row_array();
            }else{
                $this->db->order_by('id', 'desc');
                if(array_key_exists("start",$params) && array_key_exists("limi
t",$params)){
                    $this->db->limit($params['limit'],$params['start']);
                }elseif(!array_key_exists("start",$params) && array_key_exists
("limit",$params)){
                    $this->db->limit($params['limit']);
                }

                $query = $this->db->get();
                $result = ($query->num_rows() > 0)?$query->result_array():FALS
E;
            }
        }

        // Return fetched data
        return $result;
    }

    /*
     * Insert user data into the database
     * @param $data data to be inserted
     */
    public function insert($data = array()) {
        if(!empty($data)){
            // Add created and modified date if not included
            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");
            }
```

⌃

☰ **CODEXWORLD**   🔍

```php
            $insert = $this->db->insert($this->table, $data);

            // Return the status
            return $insert?$this->db->insert_id():false;
        }
        return false;
    }
}
```

View

`1. elements/`

This directory holds the element parts of the web pages.

`1.1. elements/header.php`

This file holds the header part of the web pages.

```php
<!DOCTYPE html>
<html lang="en">
<head>
<title>CodeIgniter User Login System by CodexWorld</title>
<meta charset="utf-8">
<meta name="viewport" content="width=device-width, initial-scale=1">

<link rel="stylesheet" href="//fonts.googleapis.com/css?family=Roboto:400,100,
300,500,700,900"          type="text/css" media="all">

<!-- Stylesheet file -->
<link href="<?php echo base_url('assets/css/style.css'); ?>" rel='stylesheet'
type='text/css' />
</head>
<body>

<h1>CODEIGNITER USER LOGIN SYSTEM BY CODEXWORLD</h1>
```

`1.2. elements/footer.php`

This file holds the footer part of the web pages.

```php
</body>
</html>
```

`2. users/`

This directory holds the view files of the user login system.

☰ **CODEXWORLD**   🔍

~~This view file is used to display the registration form.~~

- This file contains form HTML to collect the user account information. Once the user submits the form, it is submitted to the **registration()** method of Users controller.

```php
<div class="container">
    <h2>Create a New Account</h2>

    <!-- Status message -->
    <?php
        if(!empty($success_msg)){
            echo '<p class="status-msg success">'.$success_msg.'</p>';
        }elseif(!empty($error_msg)){
            echo '<p class="status-msg error">'.$error_msg.'</p>';
        }
    ?>

    <!-- Registration form -->
    <div class="regisFrm">
        <form action="" method="post">
            <div class="form-group">
                <input type="text" name="first_name" placeholder="FIRST NAME"
value="<?php echo !empty($user['first_name'])?$user['first_name']:''; ?>" requ
ired>
                <?php echo form_error('first_name','<p class="help-bloc
k">','</p>'); ?>
            </div>
            <div class="form-group">
                <input type="text" name="last_name" placeholder="LAST NAME" va
lue="<?php echo !empty($user['last_name'])?$user['last_name']:''; ?>" required
>
                <?php echo form_error('last_name','<p class="help-block">','</
p>'); ?>
            </div>
            <div class="form-group">
                <input type="email" name="email" placeholder="EMAIL" value="<?
php echo !empty($user['email'])?$user['email']:''; ?>" required>
                <?php echo form_error('email','<p class="help-block">','</p
>'); ?>
            </div>
            <div class="form-group">
                <input type="password" name="password" placeholder="PASSWORD"
required>
                <?php echo form_error('password','<p class="help-block">','</p
>'); ?>
```

⚡

```php
                <input type="password" name="conf_password" placeholder="CONFI
RM PASSWORD" required>
                <?php echo form_error('conf_password','<p class="help-bloc
k">','</p>'); ?>
            </div>
            <div class="form-group">
                <label>Gender: </label>
                <?php
                if(!empty($user['gender']) && $user['gender'] == 'Female'){
                    $fcheck = 'checked="checked"';
                    $mcheck = '';
                }else{
                    $mcheck = 'checked="checked"';
                    $fcheck = '';
                }
                ?>
                <div class="radio">
                    <label>
                        <input type="radio" name="gender" value="Male" <?php e
cho $mcheck; ?>>
                                        Male
                    </label>
                    <label>
                        <input type="radio" name="gender" value="Female" <?ph
p echo $fcheck; ?>>
                        Female
                    </label>
                </div>
            </div>
            <div class="form-group">
                <input type="text" name="phone" placeholder="PHONE NUMBER" val
ue="<?php echo !empty($user['phone'])?$user['phone']:''; ?>">
                <?php echo form_error('phone','<p class="help-block">','</p
>'); ?>
            </div>
            <div class="send-button">
                <input type="submit" name="signupSubmit" value="CREATE ACCOUN
T">
            </div>
        </form>
        <p>Already have an account? <a href="<?php echo base_url('users/logi
n'); ?>">Login here</a></p>
    </div>
</div>
```

- This view file is used to display the login form.

- This file contains login form HTML to authenticate the user with their account. Once the user submits the form, it is submitted to the **login()** method of Users controller.

```php
<div class="container">
    <h2>Login to Your Account</h2>

    <!-- Status message -->
    <?php
        if(!empty($success_msg)){
            echo '<p class="status-msg success">'.$success_msg.'</p>';
        }elseif(!empty($error_msg)){
            echo '<p class="status-msg error">'.$error_msg.'</p>';
        }
    ?>

    <!-- Login form -->
    <div class="regisFrm">
        <form action="" method="post">
            <div class="form-group">
                <input type="email" name="email" placeholder="EMAIL" required
="">
                <?php echo form_error('email','<p class="help-block">','</p
>'); ?>
            </div>
            <div class="form-group">
                <input type="password" name="password" placeholder="PASSWORD"
required="">
                <?php echo form_error('password','<p class="help-block">','</p
>'); ?>
            </div>
            <div class="send-button">
                <input type="submit" name="loginSubmit" value="LOGIN">
            </div>
        </form>
        <p>Don't have an account? <a href="<?php echo base_url('users/registra
tion'); ?>">Register</a></p>
    </div>
</div>
```

2.3. users/account.php

This view displays the account details of the logged-in user.

```
    <h2>Welcome <?php echo $user['first_name']; ?></h2>
    <a href="<?php echo base_url('users/logout'); ?>" class="logout">Logout</a
>

    <div class="regisFrm">
        <p><b>Name: </b><?php echo $user['first_name'].' '.$user['last_nam
e']; ?></p>
        <p><b>Email: </b><?php echo $user['email']; ?></p>
        <p><b>Phone: </b><?php echo $user['phone']; ?></p>
        <p><b>Gender: </b><?php echo $user['gender']; ?></p>
    </div>
</div>
```

> SEE ALSO: CodeIgniter CRUD Operations with Search and Pagination

Conclusion

This simple user authentication script helps you to implement the **login system in CodeIgniter**. The login script can be used for many purposes where user authentication is needed. You can easily enhance the functionality of our CodeIgniter login system script as per your needs.

codeigniter   login   session

Are you want to get implementation help, or modify or enhance the functionality of this script? _Click Here to Submit Service Request_

If you have any questions about this script, submit it to our QA community - Ask Question

⬇ Download Source Code

≡ CODEXWORLD

RELATED TUTORIALS

How to Generate SEO Friendly URL in CodeIgniter

Shopping Cart Implementation in CodeIgniter

Login with Google Account in CodeIgniter

41 COMMENTS

Kamlesah Said...

hi i need to the converter in any doc file convert to pdf in upload time and header footer watermark add this content in Codeigniter

December 30, 2020 at 10:38 AM

John Said...

Nice Post,
I have just read this post . It is very useful and it helps me a lot. Keep it up good works .
Thanks

June 18, 2020 at 11:20 AM

Tom Said...

great tutorial. I hope there is another login tutorial for CI4 soon. thanks

June 3, 2020 at 9:27 AM

Fatih Said...

very useful. Thanks for inspiration.

November 6, 2019 at 7:42 PM

≡ **CODEXWORLD**   🔍

so thank you and visit again....

October 18, 2019 at 10:19 AM

Reets Said...

if status = 0 then how to show the message for this condition?
Can you please Guide me?
basically, I want to check if a user status is 0 (0 = Deactivated by admin) or Status (1 =
Activated by default).
Please Guide me, Sir.

October 29, 2018 at 1:42 PM

Khalid Said...

Really good work! but I have a question, I want to store email or something else in
session instead of 'id', Please guide me how should I do that???

July 10, 2018 at 11:43 AM

Paras Said...

How to do either email or username and password in login

May 21, 2018 at 1:59 PM

Sana Said...

where to find the downloading source file ? is it the sourse code is free or i have to pay
for it ?
many thanks

April 26, 2018 at 12:10 AM

> **CodexWorld Said...**
>
> Click the "Download Source Code" button at the end of the tutorial.
>
> April 26, 2018 at 5:39 PM

Sana Said...

give me the css file please anyone
where should i get this links as you suggested to other people above comments
The source code contains all the required files including CSS, please download it from
the above link.
thanks

April 24, 2018 at 1:50 PM

> **CodexWorld Said...**

⟪

Marc Said...

Registration is working correctly and when I go to login it doesn't give me any error messages, but doesn't take me to the account page. I added the code:

if($this->session->userdata('isUserLoggedIn')){

redirect('users/account');

}

to see if that would help, but it didn't. Any suggestion on why this is?

March 6, 2018 at 12:35 AM

> **CodexWorld Said...**
>
> Place the following code at the beginning of the `login()` method. It will redirect the user to the account page if already logged in.
>
> ```
> if($this->session->userdata('isUserLoggedIn')){
>     redirect('users/account/');
> }
> ```
>
> March 14, 2018 at 6:30 PM

Ajith Said...

add Email verification

February 21, 2018 at 6:23 AM

Jolina Rose Said...

Good day there! I just wanna ask if there's an admin page? Thank you.

December 3, 2017 at 2:55 PM

> **CodexWorld Said...**
>
> No, there is no admin page for this script. If you want to develop an admin section of this script, send us your requirement at service@codexworld.com. Otherwise, you can buy the CodeIgniter admin panel project from our premium script section – https://www.codexworld.com/scripts/codeigniter-admin-panel/
>
> December 3, 2017 at 7:53 PM

Gaurav Said...

may i know where you use the action="" im see the tutorial but action is blank how you redirect to the registration controler

November 15, 2017 at 10:22 AM

This view is loaded by the registration() method, so, by default (action=''), the form will be submitted to the registration() method of Users controller.

November 15, 2017 at 5:56 PM

Arif Said...

give me css file please any one

July 21, 2017 at 2:43 PM

> **CodexWorld Said...**
>
> The source code contains all the required files including CSS, please download it from the above link.
>
> July 23, 2017 at 4:19 PM

Saintlight Said...

I have gotten it pls.. Thanks.

July 12, 2017 at 4:59 PM

Saintlight Said...

Pls ave downloaded the source file but don't really know where to place it. Should I place each file in its supposed folder? Like the model folder into the application model folder likewise the rest. Or are they to be left in same folder as it came.. If so pls which particular folder. Is the the codeigniter folder or application folder or which pls. Urgent response and thanks..

July 12, 2017 at 2:46 AM

> **CodexWorld Said...**
>
> Yes, place each file in its supposed directory.
>
> July 12, 2017 at 4:10 PM

Gary Said...

Thanks for sharing. Would want to know if i were to add roles to this login system, what are the things i need to add?

July 10, 2017 at 9:28 AM

Eric Rosenberg Said...

Hello! Thanks for the great guide here. I have everything working at the url /projectroot/users/registration. I tried moving the php files out of the users direc into the root for views but that gives me a 404. How can I have the registration pag accessible at /projectroot/registration?

≡ **CODEXWORLD**   🔍

CodexWorld Said...

You don't need to move the `users` directory into the root of the views. Just add the following line of code in `config/routes.php` file.

```
$route['registration'] = 'users/registration';
```

May 23, 2017 at 7:08 PM

Lokesh Said...

hey i am lokesh could you explain isUserLoggedIn predefined or inbuilt function of codeigniter...thanks

April 20, 2017 at 12:06 PM

CodexWorld Said...

`isUserLoggedIn` is a session variable.

April 20, 2017 at 5:37 PM

Bhavin Said...

Good Tutorial with detail explanation....

April 4, 2017 at 11:42 AM

Sya Said...

hey I am a new in using CodeIgniter. I followed all the steps but when i try to run it said error 404.help me

March 12, 2017 at 10:33 AM

CodexWorld Said...

To solve this issue, use the .htaccess to rewrite URL in your CodeIgniter. Go through this guide to remove index.php from URLs in CodeIgniter – http://www.codexworld.com/how-to/remove-index-php-from-url-codeigniter-htaccess/

March 13, 2017 at 3:06 PM

Kevin Fonseca Said...

Hello, I just want to thank you for doing such an amazing tutorial, keep up the good work.

February 25, 2017 at 9:39 AM

February 16, 2017 at 9:00 AM

Vkdewangga Said...

add this code may useful if user has logged

public function login(){

$data = array();

if($this->session->userdata('isUserLoggedIn')){

redirect('users/account/');

} else {

.....

$this->load->view('users/login', $data);

}

}

January 16, 2017 at 12:08 PM

CodexWorld Said...

You can make it simple by adding the following code in the `login()` function.

```
if($this->session->userdata('isUserLoggedIn')){
    redirect('users/account');
}
```

January 17, 2017 at 4:45 PM

Floyd Said...

I'm a noob. Can you tell me all the files that need to be edited?

January 12, 2017 at 11:20 AM

CodexWorld Said...

@Floyd You'll get the complete instruction in the downloadable source file. Please follow the instructions given in the `ReadMe.txt` file.

January 12, 2017 at 5:24 PM

Hetal Said...

do i have to change anything in routes.php file??

January 10, 2017 at 3:47 PM

CodexWorld Said...

No, you don't need to change anything in routes.php file.

CODEXWORLD

Dinesh Krishnan Said...

Thanks for sharing. It was useful for my current development. Keep up the good work

December 30, 2016 at 11:36 AM

LEAVE A REPLY

Comment*

Your Name*

Your Email*

Your Website

I'm not a robot

reCAPTCHA
Privacy - Terms

Post Comment

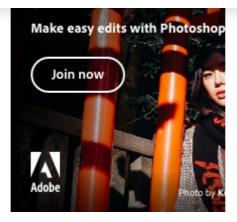## SUBSCRIBE FOR FREE NEWSLETTER

Join our 1M+ subscribers and get the latest tutorials and resources, straight to your inbox.

Enter your email...

SUBSCRIBE

TRENDING TUTORIALS

Convert HTML to PDF using JavaScript

Upload and Store Image File in Database using PHP and MySQL

Multi-select Dropdown List with Checkbox using jQuery

Store and Retrieve Image from MySQL Database using PHP

Login with Google Account using PHP

Login with Facebook using PHP

Dynamic Dependent Select Box Dropdown using jQuery, Ajax and PHP

Add Remove Input Fields Dynamically using jQuery

TOPICS   all topics

PHP	CodeIgniter
WordPress	JavaScript
GoogleMap	HTML&CSS
Web Development	Drupal
Bootstrap	CakePHP
Laravel	PayPal

LATEST HOW TO GUIDES

How to Save Output of a PHP File in an HTML File

How to change Page Orientation to Landscape in JavaScript window.print()

How to Hide DIV Element After Some Time Interval using jQuery

How to Get the Current Year to Display Dynamic Year in Copyright using JavaScript

How to Validate File Size while Uploading using JavaScript or jQuery

CodexWorld is the most popular Programming & Web Development website aiming to provide the best online resources for web application developers and designers.

ABOUT US

◉ About CodexWorld

◉ Write For Us

CONTACT

◉ Contact Us

◉ Advertise

TERMS

EXTRA LINKS

CODEXWORLD

Privacy Policy

Online Tools

Terms & Condition

Privacy Policy