Top 10 CodeIgniter interview questions with answer part -3

Topics : Codeigniter
Written on March 04, 2024

41. **What are the advantages of using CodeIgniter over other PHP frameworks?**

    - CodeIgniter has a small footprint and is lightweight, making it easy to learn and use.
    - It provides excellent performance and speed due to its efficient execution.
    - CodeIgniter offers a rich set of libraries and helpers for common tasks, reducing development time.
    - It has clear and concise documentation, making it easy to understand and work with.

42. **How do you enable and configure CSRF protection in CodeIgniter?**

    CSRF (Cross-Site Request Forgery) protection in CodeIgniter is enabled by default. You can configure CSRF settings in the `config.php` file. Here's an example:

    ```
    $config['csrf_protection'] = TRUE;
    $config['csrf_token_name'] = 'csrf_token';
    $config['csrf_cookie_name'] = 'csrf_cookie';
    $config['csrf_expire'] = 7200;
    ```

43. **Explain how to use CodeIgniter's form helper to create HTML forms.**

    CodeIgniter's Form Helper provides functions for generating HTML form elements easily. You can use functions like `form_open()`, `form_input()`, `form_dropdown()`, etc., to create form elements. Here's an example:

    ```
    echo form_open('controller/method');
    echo form_input('username', '', 'placeholder="Username"');
    echo form_password('password', '', 'placeholder="Password"');
    echo form_submit('submit', 'Login');
    echo form_close();
    ```

44. **How do you handle errors and logging in CodeIgniter?**

    Errors and logging in CodeIgniter can be configured in the `config.php` file. You can set the error logging threshold and specify the error log file path. Additionally, you can use the

`log_message()` function to log custom messages to the error log.

45. **What is HMVC architecture, and how does it differ from MVC in CodeIgniter?**

HMVC (Hierarchical Model-View-Controller) architecture is an extension of the MVC pattern that allows for better modularization and encapsulation of code. In HMVC, each module has its own MVC triad, enabling better code organization and separation of concerns compared to traditional MVC.

46. **Explain how to create custom helpers in CodeIgniter.**

You can create custom helpers in CodeIgniter by creating a PHP file in the `application/helpers` directory and defining your helper functions within it. Make sure to prefix your helper functions with a unique name to avoid conflicts with existing helpers.

47. **How do you enable and configure database query caching in CodeIgniter?**

Database query caching in CodeIgniter can be enabled by setting the `cache_on` configuration option to TRUE in the `database.php` configuration file. You can configure caching settings such as cache expiration time and cache directory in the same file.

48. **Explain how to use CodeIgniter's pagination library to paginate database results.**

CodeIgniter's Pagination library allows you to paginate database results easily. You can load the Pagination library and configure pagination settings such as the base URL, total rows, and per-page limit. Then, you can use the `create_links()` method to generate pagination links.

49. **What is HMVC, and how does it differ from MVC in CodeIgniter?**

HMVC (Hierarchical Model-View-Controller) is an architectural pattern that extends the MVC (Model-View-Controller) pattern to allow for better code organization and reusability. In HMVC, each module has its own MVC triad, allowing for hierarchical organization of code and better separation of concerns.

50. **How do you handle form validation errors in CodeIgniter?**

Form validation errors in CodeIgniter can be displayed using the `form_error()` function in the view file. This function retrieves validation errors for a specific form field and displays them next to the field. You can also display general validation errors using the `validation_errors()` function.