Top 10 CodeIgniter interview questions with answer part - 5

Topics : Codeigniter
Written on March 04, 2024

61. **What is CodeIgniter's Query Builder class, and how do you use it?**

    CodeIgniter's Query Builder class provides a set of methods for building SQL queries in a database-independent manner. You can use methods like `select()`, `where()`, `join()`, `insert()`, `update()`, and `delete()` to construct complex queries without writing raw SQL.

62. **Explain the purpose of the `profiler` in CodeIgniter.**

    The CodeIgniter profiler is a debugging tool that displays information about the execution of a page, including database queries, controller methods, memory usage, and loading times. It helps developers identify performance bottlenecks and optimize their code.

63. **How do you implement image manipulation in CodeIgniter?**

    Image manipulation in CodeIgniter can be done using libraries like GD2 or ImageMagick. You can resize, crop, rotate, and watermark images using these libraries. CodeIgniter provides helper functions for working with images, such as `image_lib` and `upload`.

64. **Explain how to handle AJAX requests in CodeIgniter.**

    AJAX requests in CodeIgniter are handled similarly to regular requests. You can create controller methods to process AJAX requests and return JSON or HTML responses. Additionally, you can use the `input` library to retrieve data from AJAX requests.

65. **What is the purpose of hooks in CodeIgniter, and how do you use them?**

    Hooks in CodeIgniter allow you to execute custom code at specific points during the request lifecycle. You can use hooks to modify the behavior of CodeIgniter without modifying its core files. Hooks are defined in the `hooks.php` file located in the `config` directory.

66. **How do you handle file uploads in CodeIgniter?**

    File uploads in CodeIgniter are handled using the File Uploading class. You can create a form

with the `enctype="multipart/form-data"` attribute and use the `do_upload()` method to handle file uploads in the controller. Additionally, you can set validation rules for file uploads using the Form Validation library.

67. **Explain how to implement pagination in CodeIgniter.**

    Pagination in CodeIgniter allows you to split large sets of data into multiple pages, making it easier for users to navigate through the data. You can enable pagination by loading the Pagination library and configuring it with your database query results.

68. **What are the different session drivers available in CodeIgniter?**

    CodeIgniter supports multiple session drivers, including the file, database, and cookie drivers. You can configure the session driver in the `config.php` file by setting the `sess_driver` configuration option.

69. **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.

70. **Explain the purpose of the `database.php` file in CodeIgniter.**

    The `database.php` file in CodeIgniter contains configuration settings for database connections. You can specify parameters such as database hostname, username, password, database name, and driver type. This file allows you to manage database connections and settings for different environments (e.g., development, production).