PHP CAKEPHP Interview Questions

What is CakePHP?

CakePHP is a model–view–controller (MVC) based PHP web framework that provides rapid development of web applications and APIs.CakePHP is inspired by Ruby on Rails framework and distributed under the MIT License.

Q1. What is CakePHP ?

CakePHP is an **open-source** free web framework written in PHP scripting Language for rapid web development. **CakePHP** follows the **model–view–controller (MVC)** approach and modeled after the concepts of **Ruby on Rails**, and distributed under the MIT License.

Q2. How to get current URL in CakePHP ?

In **Cakephp 2.x** you can get current url in view by using **$this->here;** or **Router::url($this->here, true);**

In **Cakephp 3.x** you can get current url in view by using **$this->Url->build(null, true);**

Also Read **Best Core PHP Interview Questions**
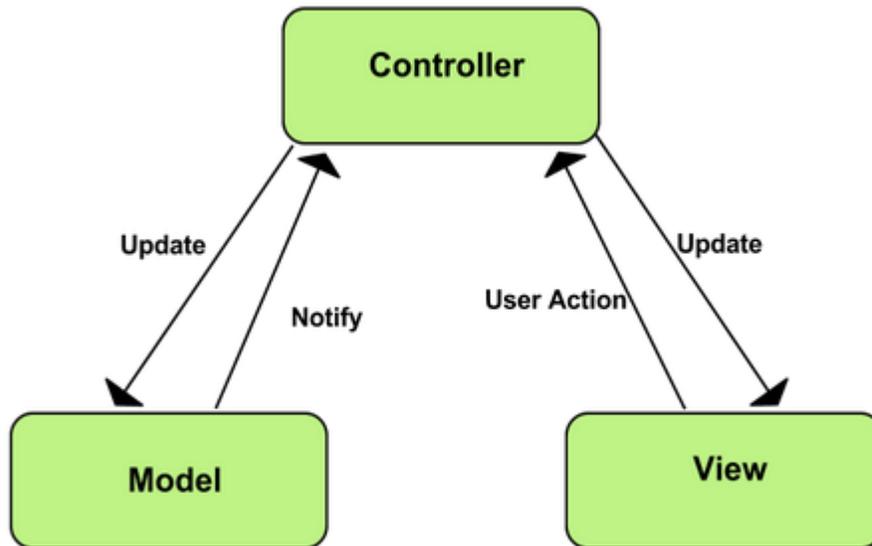
Q3. What is MVC in CakePHP?

CakePHP works on MVC structure.

MVC stands for the model view controller.MVC is not a design pattern, it is an architectural pattern that describes a way to structure our application and explains the responsibilities and interactions of each part in that structure:

Model: Wraps up data and logic of CakePHP.
View: Handles output and presentation to User.
Controller: Manipulates the data from models and generates or pass it to views.

Caption

image Source:https://developer.chrome.com/apps/app_frameworks

Q4. What are Hooks in CakePHP?

CakePHP **hooks** are callback functions that are called before or after a model operation.We define these functions in our Model classes.

Below is the list of some hooks or callback functions provided by CakePHP.

- beforeFind
- afterFind
- beforeValidate
- afterValidate
- beforeSave
- afterSave
- beforeDelete
- afterDelete
- onError

Q5. List some features of CakePHP framework ?

Top features of CakePHP framework

- MVC Architecture
- Zero configuration

- Inbuilt validation
- ACL Functionality and Security
- CRUD scaffolding
- Easily extendable with plug-ins
- Quick and flexible

You may also Read **PHP 7 interview questions**

Q6. What is the name of CakePHP database configuration file ?

database.php.default file is used for database configuration in CakePHP. It is located in **/app/config/** directory of CakePHP

Q7. What are sessions in PHP . How to read, write and delete session in cakephp ?

Session in PHP

PHP Sessions allows you to identify unique users across requests and store persistent data for specific users against a unique **session ID**. This can be used to persist state information between page requests. Session IDs are normally sent to the browser via session cookies and the ID is used to retrieve existing session data.

Reading, Writing and Deleting session in Cakephp 3.x

You can access the session data any place you have access to a request object. This means the session is accessible from:

- Controllers
- Views
- Helpers
- Cells
- Components

In addition to the basic session object, you can also use the Cake\View\Helper\SessionHelper to interact with the session in your views. A basic example of session usage would be:

```
$name = $this->request->session()->read('User.name');
// If you are accessing the session multiple times,
// you will probably want a local variable.
$session = $this->request->session();
$name = $session->read('User.name');
```

Reading, Writing and Deleting Session Data

Session::read($key) function is used to read specific session data in CakePHP.

Session::write($key, $value) function is used to write session data in CakePHP.

Session::delete($key) function is used to delete specific session data in CakePHP.

Sample Usage

```
//Reading a session
$session->read('Config.language');
//Writing a session
$session->write('Config.language', 'en');
//Deleting a session
$session->delete('Some.value');
```

Q8. How to use Pagination in CakePHP ?

Pagination in CakePHP

In **CakePHP** controller **Pagination** component is used to building paginated queries. In order to generate pagination links & buttons in view **PaginatorHelper** is used

Below are sample pagination usage code in CakePHP

Pagination in Controller

```
class PostsController extends AppController
{
    public $paginate = [
        'limit' => 25,
        'order' => [
            'Posts.id' => 'desc'
        ]
    ];
    public function initialize()
    {
        parent::initialize();
        $this->loadComponent('Paginator');
        $this->loadHelper('Paginator', ['templates' => 'MyPlugin.paginator-templa
    }
    public function display(){
      $this->set('posts', $this->paginate());
    }
}
```

Pagination in CakePHP Views

```
<?php
$paginator = $this->Paginator;

if($posts){
```

```php
        //creating our table
        echo "<table>";

            // our table header, we can sort the data user the paginator sort() metho
            echo "<tr>";

                // in the sort method, there first parameter is the same as the colum
                // the second parameter is the header label we want to display in the
                echo "<th>" . $paginator->sort('id', 'ID') . "</th>";
                echo "<th>" . $paginator->sort('title', 'Title') . "</th>";
                echo "<th>" . $paginator->sort('published_on', 'Published on') . "</t

            echo "</tr>";

            // loop through the user's records
            foreach( $posts as $post ){
                echo "<tr>";
                    echo "<td>{$post['Post']['id']} </td>";
                    echo "<td>{$post['Post']['title']} </td>";
                    echo "<td>{$post['Post']['published_on']} </td>";

                echo "</tr>";
            }

        echo "</table>";

        // pagination section
        echo "<div class='paging'>";

            // the 'first' page button
            echo $paginator->first("First");

            // 'prev' page button,
            // we can check using the paginator hasPrev() method if there's a previou
            // save with the 'next' page button
            if($paginator->hasPrev()){
                echo $paginator->prev("Prev");
            }

            // the 'number' page buttons
            echo $paginator->numbers(array('modulus' => 2));

            // for the 'next' button
            if($paginator->hasNext()){
                echo $paginator->next("Next");
            }

            // the 'last' page button
            echo $paginator->last("Last");

        echo "</div>";

    }

// tell the user there's no records found
else{
    echo "No posts found.";
```

```
}
?>
```

Q9. [List different type of Cache CakePHP Supports ?](#)

CakePHP supports **Cache** out of the box. Below is the list of **Cache** engines supported by **CakePHP**.

- APCu
- File Based
- Memcached
- Redis
- Wincache
- XCache

Q10. [What is Composer? How to create a CakePHP Project using Composer?](#)

Composer is a tool for managing project dependencies. You can create a CakePHP project using Composer by running below commands on terminal.

```
php composer.phar create-project --prefer-dist cakephp/app my_app_name
```

Q11. [What are components in CakePHP. List some commonly used CakePHP components ?](#)

In CakePHP, components are packages of logic that are shared between controllers. CakePHP comes with a fantastic set of core components you can use to aid in various common tasks.Using component in your application makes your controller code cleaner and allows you to reuse code between different controllers.

Below are the list some commonly used CakePHP components

- Authentication
- Cookie
- Cross Site Request Forgery
- Flash
- Security
- Pagination
- Request Handling

Q12. [What is name of default function and controller of CakePHP which is called automatically?](#)

In CakePHP Default controller is **indexController.php** and Default function is **index**.

Q13. List Types of association supported by CakePHP?

There are four association supported by CakePHP they are:

- **hasOne:** One to One Relationship
- **hasMany:** One to many Relationship
- **belongsTo:** Many to One Relationship
- **hasAndBelongsToMany (HABTM):**Many to Many Relationship

Q14. List minimum server requirements to install CakePHP ?

Minimum server requirements to install CakePHP 3.8.5

- HTTP Server. For example Apache. Having mod_rewrite is preferred, but by no means required.
- PHP 5.6.0 or greater (including PHP 7.1).
- mbstring PHP extension installed and enabled
- intl PHP extension
- simplexml PHP extension