

# 67 CakePHP Developer interview questions to ask applicants

## Questions

---

- undefined
- undefined
- undefined
- undefined
- undefined
- undefined
- undefined
- undefined
- What are the key components of the MVC architecture in CakePHP?
- How do you create a new controller in CakePHP?
- Explain the purpose of the 'beforeFilter' and 'afterFilter' callbacks in CakePHP controllers.
- What is the difference between '\$this->set()' and '\$this->viewBuilder()->setVars()' in CakePHP?
- How do you create and use helpers in CakePHP?
- Explain the concept of 'baking' in CakePHP and when you would use it.
- What is the purpose of the 'AppController' in CakePHP?
- How do you handle form submissions in CakePHP?
- Explain the difference between 'find()' and 'get()' methods in CakePHP models.
- How do you create and use components in CakePHP?
- What is the purpose of the 'bootstrap.php' file in a CakePHP application?
- How do you implement user authentication in CakePHP?
- Explain the concept of 'containable behavior' in CakePHP and its use.
- How do you create and use custom validation rules in CakePHP models?
- What is the purpose of the 'routes.php' file in CakePHP?
- How do you handle file uploads in CakePHP?
- Explain the difference between 'hasMany' and 'belongsToMany' associations in CakePHP.
- How do you use the CakePHP ORM to perform complex database queries?
- What are plugins in CakePHP and how do you create one?
- How do you implement AJAX requests in a CakePHP application?
- How would you implement real-time functionality in a CakePHP application?
- Describe your approach to optimizing database queries in a large-scale CakePHP application.
- How would you approach building a multi-tenant architecture in CakePHP?
- Explain how you would implement a custom authentication system in CakePHP that goes beyond the built-in Auth component.
- How would you design and implement a robust plugin system for a CakePHP application?
- How do you manage views in CakePHP, and what is their role in the MVC architecture?
- Can you explain how CakePHP's routing works and its importance in the MVC framework?
- Describe how you would implement a model in CakePHP and its interaction with controllers.
- What strategies do you use to maintain separation of concerns within the MVC pattern in CakePHP?
- How do you handle data validation in CakePHP, particularly within the context of MVC?
- Explain the process of passing data from the controller to the view in CakePHP.
- What techniques do you use to minimize redundancy in views within a CakePHP application?
- How does CakePHP's ORM simplify model interactions, and what are its limitations?
- Discuss the role of helpers in views and how they aid in following the MVC pattern.
- What is the significance of the controller layer in CakePHP, and how do you ensure it remains efficient?
- How do you handle error management in the MVC structure of a CakePHP application?
- What are some best practices for organizing code within the MVC architecture of CakePHP?
- Can you describe how CakePHP facilitates RESTful architecture and where it fits in the MVC model?
- How do you keep the model and controller layers synchronized in a CakePHP application?
- Can you explain the concept of entities in CakePHP's ORM?
- How do you define and use associations in CakePHP's ORM?
- Describe the difference between find() and get() methods in CakePHP's ORM.
- How would you implement custom finders in CakePHP's ORM?
- Explain the concept of virtual fields in CakePHP's ORM and how you would use them.
- How do you handle data validation in CakePHP's ORM?
- Describe how you would use the containable behavior in CakePHP's ORM.
- How would you implement a complex query using CakePHP's ORM?
- Explain the concept of lazy loading vs. eager loading in CakePHP's ORM.
- undefined
- undefined
- undefined