Grokking the Low Level Design Interview Using OOD Principles

Intermediate 173 Lessons

50h Certificate of Completion

A battle-tested guide to Object Oriented Design Interviews – developed by FAANG engineers. Master OOD fundamentals & practice real-world interview questions.

AI-POWERED

Code Feedback

Mock Interview

Explanations Prompt

Unlock Full Access

Continue Learning

THIS COURSE INCLUDES

21 AI Prompts 21 AI Feedbacks

8 Playgrounds 8 Quizzes

Course Overview

With hundreds of potential problems to design, preparing for the object-oriented design (OOD) interview can feel like a daunting task. However, with a strategic approach, OOD interview prep doesn't have to take more than a few weeks.

In this course, you'll learn the fundamentals of object-oriented design with an extensive set of real-world problems to help you prepare for the OOD part of a typical software engineering interview process at major tech companies like Apple, Google, Meta, Microsoft, and Amazo... Show More

TAKEAWAY SKILLS

Java | Software Engineering | Prepare For Interview | Object Oriented Design

What You'll Learn

- ✓ An understanding of the essential object-oriented concepts like design principles and patterns to ace the OOD interview
- ✓ The ability to efficiently breakdown an interview design problem into multiple parts using a bottom-up approach
- ✓ Familiarity with the scope of each interview problem by accurately defining the requirements and presenting its solution
- ✓ Learn to design class, use case, sequence and activity diagrams of the interview problems
- ✓ Hands-on experience to create sequence and activity diagrams for the interview problems

✓ Explore the implementation code of each designed problem using multiple programming languages (Java, C++, C#, Python, and JavaScript)

Course Content

Collapse All ↕

1. Introduction

This chapter introduces the course, including its prerequisites, structure, strengths, and the target audience.

- Overview
- Introduction to the Course

2. Cornerstones of Object-oriented Programming

This chapter covers the background of OOP, focusing on encapsulation, abstraction, inheritance, and polymorphism, ending with a quiz on basics.

- Background of Object-oriented Programming (OOP)
- Encapsulation
- Abstraction
- Inheritance
- Polymorphism
- Quiz: Object-oriented Basics

3. Object-oriented Design

This chapter explores how to conduct object-oriented analysis and design using UML, covering various diagrams like use case, class, sequence, and activity.

- Introduction to Object-oriented Analysis and Design (OOAD)
- Introduction to the Unified Modeling Language
- Types of UML Diagrams
- Use Case Diagram
- Class Diagram
- Sequence Diagram
- Activity Diagram
- Quiz: Object-oriented Design

4. Object-oriented Design Principles

This chapter covers the SOLID design principles: Single Responsibility, Open Closed, Liskov Substitution, Interface Segregation, and Dependency Inversion.

- Introduction to SOLID Design Principles
- SOLID: Single Responsibility Principle
- SOLID: Open Closed Principle
- SOLID: Liskov Substitution Principle
- SOLID: Interface Segregation Principle
- SOLID: Dependency Inversion Principle
- Quiz: Object-oriented Design Principles

5. Design Patterns

This chapter teaches you about the classification of design patterns, including creational, structural, and behavioral patterns, followed by a quiz.

- Introduction to the Design Patterns

- Classification of Design Patterns
- Creational Design Patterns
- Structural Design Patterns
- Behavioral Design Patterns
- Quiz: Design Patterns

6. Real-world Design Problems

In this chapter, you will discover how to approach and solve real-world design problems using object-oriented principles and methodologies.

- An Approach to Solve a Real-world Problem

7. Designing a Parking Lot

This chapter covers the design of a parking lot system, including requirements, UML diagrams, and implementation code to illustrate practical application.

- Getting Ready: Parking Lot
- Requirements for the Parking Lot Design
- Use Case Diagram for the Parking Lot
- Class Diagram for the Parking Lot
- Sequence Diagram for the Parking Lot
- Activity Diagram for the Parking Lot
- Code for the Parking Lot

8. Designing an Elevator System

This chapter shows the design of an elevator system, detailing the requirements, use case, class, sequence, and activity diagrams, and the implementation code.

- Getting Ready: Elevator System
- Requirements for the Elevator System
- Use Case Diagram for the Elevator System
- Class Diagram for the Elevator System
- Sequence Diagram for the Elevator System
- Activity Diagram for the Elevator System
- Code of Elevator System

9. Designing a Library Management System

This chapter teaches you the design of a library management system with requirements, use case, class, sequence, activity diagrams, and implementation code.

- Getting Ready: Library Management System
- Requirements for the Library Management System
- Use Case Diagram for the Library Management System
- Class Diagram for the Library Management System
- Sequence Diagram for the Library Management System
- Activity Diagram for the Library Management System
- Code of Library Management System

10. Designing the Amazon Locker Service

This chapter covers the design of the Amazon Locker Service, focusing on requirements, UML diagrams, and implementation code to create a complete system.

- Getting Ready: Amazon Locker Service
- Requirements for the Amazon Locker Service
- Use Case Diagram for the Amazon Locker Service
- Class Diagram for the Amazon Locker Service
- Sequence Diagram for the Amazon Locker Service
- Activity Diagram for the Amazon Locker Service
- Code for the Amazon Locker Service

11. Designing a Vending Machine

In this chapter, you will learn the design of a vending machine, including requirements, use case, class, activity diagrams, and implementation code.

- Getting Ready: Vending Machine
- Requirements for the Vending Machine
- Use Case Diagram for the Vending Machine
- Class Diagram for the Vending Machine
- Activity Diagram for the Vending Machine
- Code for the Vending Machine

12. Designing an Online Blackjack Game

This chapter explores how to design an online blackjack game, covering requirements, use case, class, activity diagrams, and implementation code.

- Getting Ready: Online Blackjack Game

- Requirements for the Online Blackjack Game
- Use Case Diagram for the Online Blackjack Game
- Class Diagram for the Online Blackjack Game
- Activity Diagram for the Online Blackjack Game
- Code for the Online Blackjack Game

13. Designing a Meeting Scheduler

This chapter shows the design of a meeting scheduler system, detailing the requirements, UML diagrams, and implementation code for the complete design.

- Getting Ready: The Meeting Scheduler Problem
- Requirements for the Meeting Scheduler
- Use Case Diagram for the Meeting Scheduler
- Class Diagram for the Meeting Scheduler
- Sequence Diagram for the Meeting Scheduler
- Activity Diagram for the Meeting Scheduler
- Code for the Meeting Scheduler

14. Designing a Movie Ticket Booking System

This chapter covers the design of a movie ticket booking system, including requirements, UML diagrams, and implementation code to build the system.

- Getting Ready: Movie Ticket Booking System
- Requirements for the Movie Ticket Booking System
- Use Case Diagram for the Movie Ticket Booking System

- Class Diagram for the Movie Ticket Booking System
- Sequence Diagram for the Movie Ticket Booking System
- Activity Diagram for the Movie Ticket Booking System
- Code for the Movie Ticket Booking System

15. Designing a Car Rental System

This chapter teaches you the design of a car rental system, detailing requirements, UML diagrams, and implementation code for a functional system.

- Getting Ready: The Car Rental System
- Requirements for the Car Rental System
- Use Case Diagram for the Car Rental System
- Class Diagram for the Car Rental System
- Sequence Diagram for the Car Rental System
- Activity Diagram for the Car Rental System
- Code for the Car Rental System

16. Designing ATM

In this chapter, you will discover the design of an ATM system, focusing on requirements, use case, class, sequence, activity diagrams, and implementation code.

- Getting Ready: The ATM System
- Requirements for the ATM System
- Use Case Diagram for the ATM System

- Activity Diagram for the ATM System
- Code for the ATM System

17. Designing a Chess Game

This chapter shows the design of a chess game, including requirements, use case, class, activity diagrams, and implementation code to develop the game.

- Getting Ready: The Chess Game
- Requirements for the Chess Game
- Use Case Diagram for the Chess Game

18. Designing a Hotel Management System

This chapter covers the design of a hotel management system, detailing the requirements, UML diagrams, and implementation code for a complete system.

- Getting Ready: The Hotel Management System
- Requirements for the Hotel Management System
- Use Case Diagram for the Hotel Management System
- Class Diagram for the Hotel Management System
- Sequence Diagram for the Hotel Management System
- Activity Diagram for the Hotel Management System

- Code for the Hotel Management System

19. Designing the Amazon Online Shopping System

This chapter teaches you the design of the Amazon online shopping system, focusing on requirements, UML diagrams, and implementation code.

- Getting Ready: The Amazon Online Shopping System
- Requirements for the Amazon Online Shopping System
- Use Case Diagram for the Amazon Online Shopping System
- Class Diagram for the Amazon Online Shopping System
- Sequence Diagram for the Amazon Online Shopping System
- Activity Diagram for the Amazon Online Shopping System
- Code for the Amazon Online Shopping System

20. Designing Stack Overflow

In this chapter, you will learn the design of Stack Overflow, including requirements, use case, class, sequence, activity diagrams, and implementation code.

- Getting Ready: Stack Overflow
- Requirements for Stack Overflow
- Use Case Diagram for Stack Overflow
- Class Diagram for Stack Overflow
- Sequence Diagram for Stack Overflow
- Activity Diagram for Stack Overflow
- Code for Stack Overflow

21. Designing a Restaurant Management System

This chapter explores how to design a restaurant management system, covering requirements, UML diagrams, and implementation code for the system.

- Getting Ready: The Restaurant Management System
- Requirements for the Restaurant Management System
- Use Case Diagram for the Restaurant Management System
- Class diagram for the Restaurant Management System
- Sequence Diagram for the Restaurant Management System
- Activity Diagram for the Restaurant Management System
- Code for the Restaurant Management System

22. Designing Facebook

This chapter shows the design of the Facebook system, detailing requirements, use case, class, sequence, activity diagrams, and implementation code.

- Getting Ready: The Facebook System
- Requirements for Facebook
- Use Case Diagram for Facebook
- Class Diagram for Facebook
- Sequence Diagram for Facebook
- Activity Diagram for Facebook
- Code for Facebook

23. Designing an Online Stock Brokerage System

This chapter covers the design of an online stock brokerage system, including requirements, UML diagrams, and implementation code.

- Getting Ready: An Online Stock Brokerage System
- Requirements for the Online Stock Brokerage System
- Use Case Diagram for the Online Stock Brokerage System
- Class Diagram for the Online Stock Brokerage System
- Sequence Diagram for the Online Stock Brokerage System
- Activity Diagram for the Online Stock Brokerage System
- Code for the Online Stock Brokerage System

24. Designing a Jigsaw Puzzle

In this chapter, you will discover the design of a jigsaw puzzle, focusing on requirements, class diagram, and implementation code for the puzzle.

- Getting Ready: Jigsaw Puzzle
- Requirements for the Jigsaw Puzzle
- Class Diagram for the Jigsaw Puzzle
- Code of Jigsaw Puzzle

25. Designing an Airline Management System

This chapter shows the design of an airline management system, detailing the requirements, UML diagrams, and implementation code for the system.

- Getting Ready: The Airline Management System
- Requirements for the Airline Management System

- Use Case Diagram for the Airline Management System
- Class Diagram for the Airline Management System
- Sequence Diagram for the Airline Management System
- Activity Diagram for the Airline Management System
- Code for the Airline Management System

26. Designing Cricinfo

This chapter covers the design of the Cricinfo system, including requirements, use case, class, sequence, activity diagrams, and implementation code.

- Getting Ready: The Cricinfo System
- Requirements for Cricinfo
- Use Case Diagram for Cricinfo
- Class Diagram for Cricinfo
- Sequence Diagram for Cricinfo
- Activity Diagram for Cricinfo
- Code for Cricinfo

27. Designing LinkedIn

This chapter explores how to design the LinkedIn system, covering requirements, UML diagrams, and implementation code for a complete design.

- Getting Ready: The LinkedIn System
- Requirements for LinkedIn
- Use Case Diagram for LinkedIn
- Class Diagram for LinkedIn

- Sequence Diagram for LinkedIn
- Activity Diagram for LinkedIn
- Code for LinkedIn

28. Wrapping Up

This chapter concludes the course with valuable tips for excelling in object-oriented design interviews.

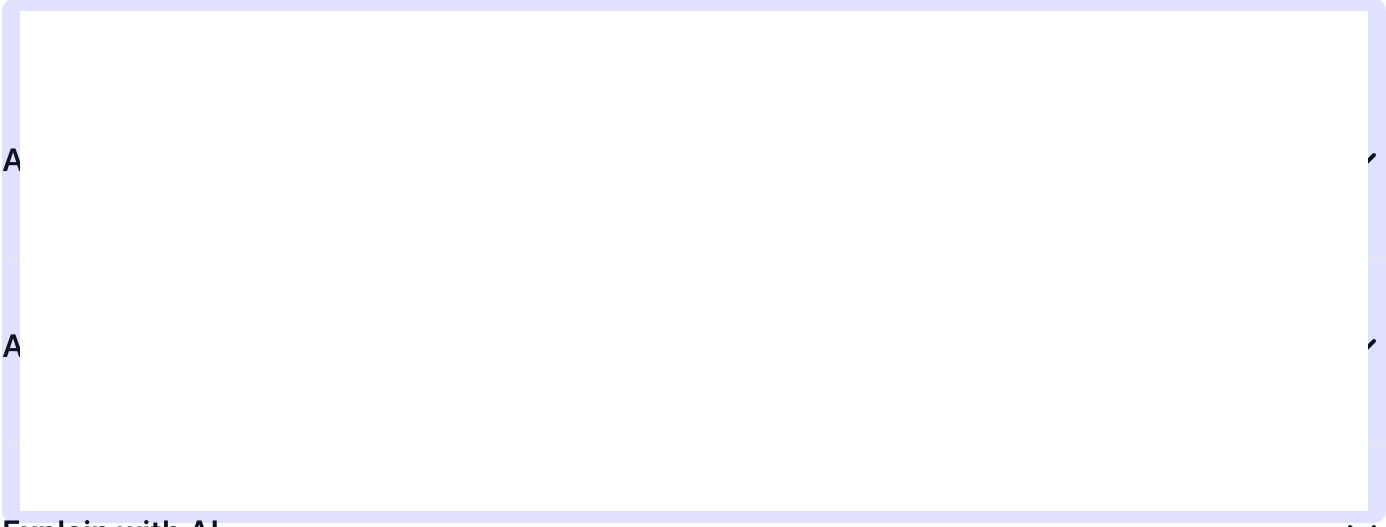
- Interview Tips
- Conclusion

Hands-on Learning Powered by AI

See how Educative uses AI to make your learning more immersive than ever before.

Instant Code Feedback

Evaluate and debug your code with the click of a button. Get real-time feedback on test cases, including time and space complexity of your solutions.



Explain with AI

AI Code Mentor

Unlock Full Access

Recommended before starting this course

Course
Learn Object-Oriented Programming in Java
Intermediate 6 hour

Course
JavaScript Design Patterns for Coding Interviews
Intermediate 9 hour

Course
Data Structures for Coding Interviews in Java
Beginner 35 hour

Related Courses and Skill Paths

Course
Grokking the Behavioral Interview
Beginner 5 hour

Course
Grokking the Coding Interview Patterns
Intermediate 85 hour

Course
Grokking the Modern System Design Interview
Intermediate 26 hour

Frequently Asked Questions

What is an object-oriented design interview?

An object-oriented design (OOD) interview is a specialized technical interview used to evaluate your understanding of object-oriented programming principles and your ability to apply these principles to solve design problems. Typically lasting 45-60 minutes, this interview covers the basics of object-oriented programming and more complex concepts such as design principles and patterns. This is crucial for roles at major tech companies like Apple, Google, Meta, Microsoft, and Amazon.

How to prepare for an OOD interview?

What are the principles of Low-Level Design?

How do you answer Low-Level Design questions?

What is the primary focus of this course?

Who should take this course?

What skills will I gain from this course?

What tools will I use to present solutions?

How will this course help me in a software engineering interview?

What companies' interview processes are referenced in this course?