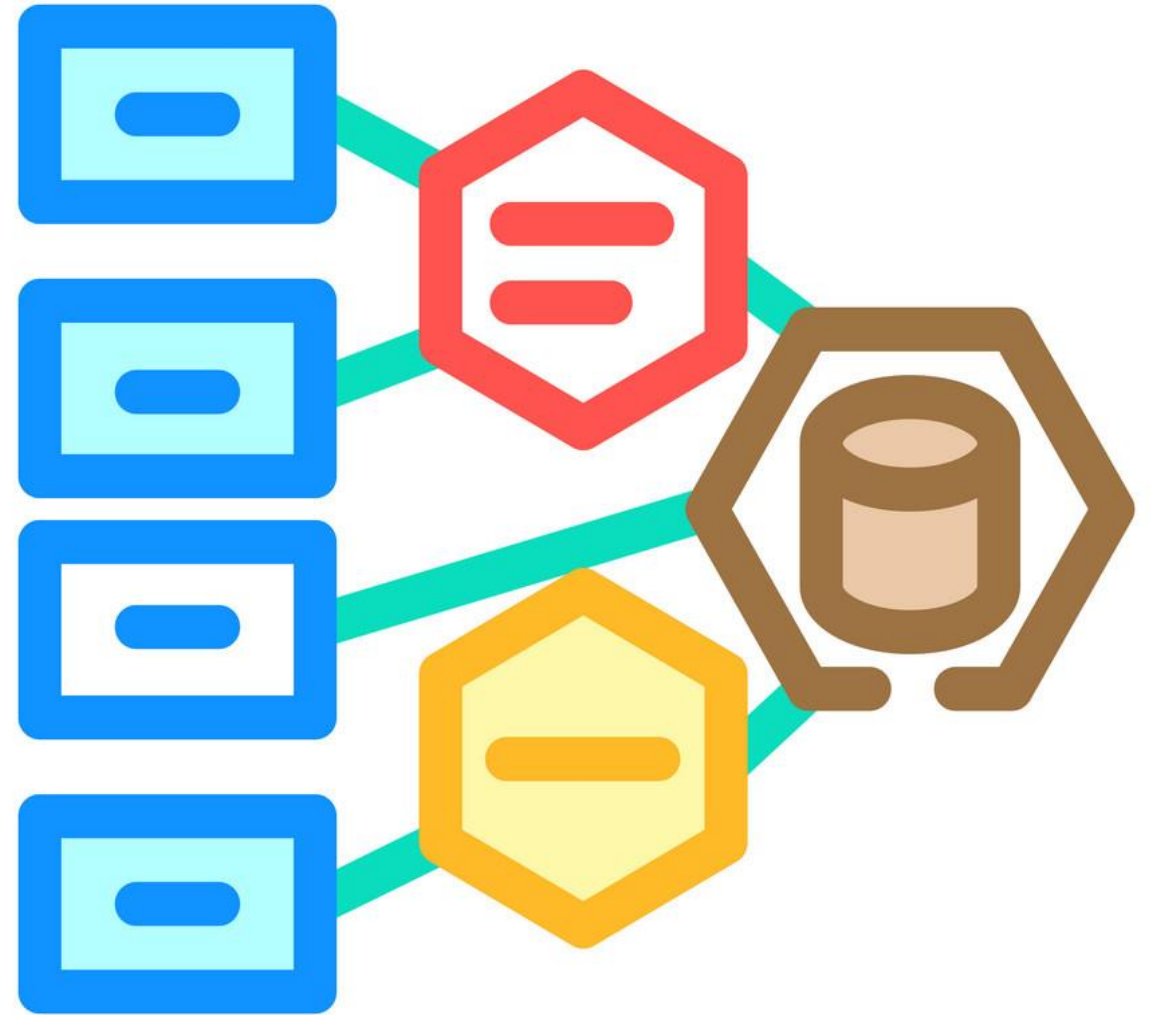


Software Design and architecture

Fall 2024

Dr. Natalia Chaudhry



Outline

- How 'Object-Oriented' Is Our Design?

- Object-oriented design (OOD) focuses on creating systems structured around objects, which encapsulate both data and behavior.
- Evaluating how "object-oriented" a design is involves assessing adherence to core OOD principles, including encapsulation, inheritance, polymorphism, and modularity.

Key Concepts in Object-Oriented Design

- 1.Encapsulation:** Combining data and methods that operate on that data into a single unit (class) and controlling access through visibility modifiers.
- 2.Inheritance:** Sharing common functionality through a hierarchical class structure.
- 3.Polymorphism:** Using a single interface to represent different data types or operations.
- 4.Abstraction:** Hiding complexity and exposing only the necessary details.
- 5.Modularity:** Breaking the system into independent, interchangeable components.

Assessing Object-Oriented Design

- Analyze the **clarity** and **cohesion** of classes.
- Ensure **low coupling** (minimal dependencies between classes).
- Verify that the design adheres to **SOLID principles**:
 - Single Responsibility Principle
 - Open-Closed Principle
 - Liskov Substitution Principle
 - Interface Segregation Principle
 - Dependency Inversion Principle

Role of Refactoring in OOD

- Refactoring is the process of improving the internal structure of code without altering its external behavior.
- It is vital for evolving a design to become more object-oriented and aligned with OOD principles.

Benefits of Refactoring

- Enhances code readability and maintainability.
- Reduces technical debt.
- Improves adherence to OOD principles.
- Facilitates future changes and scalability.

Evaluating how "object-oriented" a design is involves not only initial design but continuous improvement through refactoring.

Object-oriented principles and refactoring work together to create a robust, maintainable, and scalable system.

That's it