ooa	ood	oop
• OOA is the initial phase in the software development process, where you analyze a problem domain to identify and define objects and their relationships. • It involves understanding the real-world problem or system you're going to model in your software. • The goal of OOA is to create a conceptual model of the problem domain, focusing on what needs to be done rather than how it will be implemented.	• OOD (Object-Oriented Design): • OOD is the phase that follows OOA and focuses on transforming the conceptual model created during analysis into a more detailed and implementation-oriented design. • In OOD, you define how the system's objects will be organized and interact to solve the problem. • You create class diagrams, define methods and attributes for each class, specify interfaces, and	• OOP (Object-Oriented Programming): • OOP is the actual implementation of software based on the object-oriented design created during OOA and OOD. • It involves writing code using object-oriented principles, where you create classes, objects, and use inheritance, encapsulation, and polymorphism. • OOP allows you to structure your code into reusable and self-contained objects, making it easier to

• During OOA, you identify objects, attributes, behaviors, and relationships between objects, which serve as the foundation for designing the software system. • Common techniques used in OOA include use case diagrams, class diagrams, and other modeling tools.	establish design patterns. • The goal of OOD is to produce a design that is modular, maintainable, and scalable, ensuring that the system can be implemented effectively. • OOD sets the stage for writing code (implementation) in a structured and organized way.	understand, maintain, and extend. • It promotes the use of classes to model real-world entities and their behaviors. • Popular programming languages that support OOP include Java, C++, Python, and C#.