

객체지향 분석 방법론 중 Coad와 Yourdon 방법 설명해줘.

Coad와 Yourdon의 객체지향 분석 방법론은 객체지향 분석(Object-Oriented Analysis, OOA)을 체계적으로 수행하기 위한 방법론으로, 문제를 객체 중심으로 모델링하고 분석하는 데 중점을 둡니다. 이 방법론은 다음과 같은 주요 특징과 절차를 포함합니다.

Coad와 Yourdon 방법론의 주요 특징

1. 객체 중심 분석:

- 데이터(상태)와 프로세스(행위)를 통합하여 실제 세계의 객체를 모델링합니다.
- 기존의 전통적 분석 방법론(예: 데이터 흐름도, ER 다이어그램)에서는 데이터와 프로세스를 별도로 다루는 반면, Coad와 Yourdon 방법은 이를 객체로 통합하여 표현합니다^{[1] [2]}.

2. 다섯 가지 주요 단계:

Coad와 Yourdon은 객체지향 분석을 다음 다섯 가지 단계로 나누어 수행합니다^{[3] [4]}:

- **클래스 및 객체 식별**: 시스템에서 필요한 클래스와 객체를 찾아내고 정의합니다.
- **구조 식별**: 클래스 간의 관계를 정의하며, 상속(Generalization-Specialization) 및 집합(Whole-Part) 구조를 포함합니다.
- **주제(Subjects) 정의**: 관련된 클래스와 객체를 그룹화하여 하나의 주제로 묶음으로써 복잡성을 줄입니다.
- **속성 정의**: 각 클래스의 속성을 명확히 하고, 객체의 상태를 설명합니다.
- **서비스 정의**: 각 클래스가 제공하는 기능이나 메서드를 정의하여 객체의 행위를 나타냅니다.

3. 주제 기반 모델링:

- "주제(subject)"는 서로 밀접하게 관련된 클래스 및 객체의 그룹으로, 시스템을 더 쉽게 이해하고 관리할 수 있도록 돕습니다.
- 주제 다이어그램은 시스템의 상위 수준 개요를 제공하며, 이를 기반으로 세부적인 클래스 다이어그램을 설계할 수 있습니다^{[5] [3]}.

4. 그래픽 표현:

- 객체 다이어그램(Object Diagram)은 데이터 흐름도(DFD)와 확장된 ER 다이어그램을 결합한 형태로 사용됩니다.
- 이 접근법은 사용자와 개발자 간의 의사소통을 개선하고 시스템 요구사항을 명확히 하는 데 유용합니다^{[2] [4]}.

Coad와 Yourdon 방법론의 절차

1. 클래스 및 객체 식별

- 시스템에서 중요한 엔티티를 찾아내고 이를 클래스와 객체로 정의합니다.
- 예: 고객(Customer), 주문(Order), 제품(Product) 등.

2. 구조 식별

- 클래스 간 관계를 정의하며, 상속 관계(일반화-특수화)와 집합 관계(전체-부분)를 포함합니다.
- 상속은 부모 클래스와 자식 클래스 간의 관계를 나타내며, 집합 관계는 특정 클래스가 다른 클래스를 포함하는 경우를 설명합니다.

3. 주제 정의

- 관련된 클래스를 그룹화하여 "주제(subject)"로 묶습니다.
- 예: "주문 관리" 주제에는 고객, 주문, 제품 등이 포함될 수 있습니다.

4. 속성 정의

- 각 클래스의 속성을 명확히 하여 데이터를 설명합니다.
- 예: 고객 클래스는 이름(name), 주소(address), 연락처(contact info) 등의 속성을 가질 수 있습니다.

5. 서비스 정의

- 각 클래스가 제공하는 기능이나 메서드를 정의하여 객체 간 상호작용을 지원합니다.
- 예: 주문 클래스는 "주문 생성(createOrder)" 또는 "주문 취소(cancelOrder)" 서비스를 가질 수 있습니다.

Coad와 Yourdon 방법론과 UML 비교

특징	Coad/Yourdon 방법론	UML
목적	문제 영역을 객체 중심으로 분석	표준화된 시각적 모델링 언어 제공
다이아그램 종류	객체 다이어그램, 데이터 흐름도 등	다양한 UML 다이어그램 (클래스, 유스케이스 등)
표현 방식	단순하고 분석 중심	복잡하고 설계 및 구현까지 지원
표준화 여부	독립적인 방법론	OMG에 의해 표준화됨

활용 사례

Coad와 Yourdon 방법론은 소규모 프로젝트나 분석 중심 작업에 적합하며, 복잡한 시스템에서는 UML과 같은 표준화된 모델링 언어로 대체되는 경우가 많습니다. 그러나 이 방법론은 초기 객체지향 분석에서 중요한 기여를 했으며, 현대적인 OOAD(Object-Oriented Analysis and Design)의 기초가 되었습니다.^{[6] [1]}

✻

1. <https://www.conceptdraw.com/How-To-Guide/yourdon-and-coad-diagram>
2. <https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=5f7c86263a26c2057bba5019efd0f668e1ac4931>
3. <https://dl.acm.org/doi/10.5555/130444>
4. https://books.google.com/books/about/Object_oriented_Analysis.html?id=M8S9-gGxk1AC
5. <https://softwareengineering.stackexchange.com/questions/311432/object-oriented-analysis-peter-coad-edward-yourdon-notation>
6. https://en.wikipedia.org/wiki/Object-oriented_analysis_and_design