

현실세계를 통한 객체이해

스터디 1회차 발표 - ES

섹션1 객체 개념 설명 중...

- 역할(인터페이스)과 구현(인터페이스 구현 객체)로 세상을 구분
 - ex) 자동차, 공연 등
- 객체는 서로 '협력'한다
 - 클라이언트(요청) ↔ 서버(응답)
 - 객체 클라이언트 ↔ 객체 서비스
- 실세계의 역할과 구현 → 다형성 → 객체 세상으로 가져옴

현실세계와 객체

객체지향 소프트웨어는 실세계의 투영이며 객체란 현실세계에 존재하는 사물에 대한 추상화다 어쩌구저쩌구살ㄹ랴살라



현실세계와 객체

“소프트웨어 시스템이 해결하려는 실재는 잘해봐야
먼 친척밖에는 되지 않는다” - 버트란드 마이어

- 객체지향의 목표는 실세계 모방(X) 새로운 세계 창조(O)
- BUT 소프트웨어 객체의 **자율성**을 설명하는데 효과적
- 즉, 실세계의 모방이란 객체지향 개념은
 - 프로그램 설계, 구현하는 실무적 관점에서 X
 - 하지만 객체지향 개념 학습엔 효과적

현실세계 이야기

출근길 카페



노비

역할 커피를 주문하는 손님
책임 커피 주문할 책임



카페 캐시어

주문 받는 캐시어
주문 받는 책임

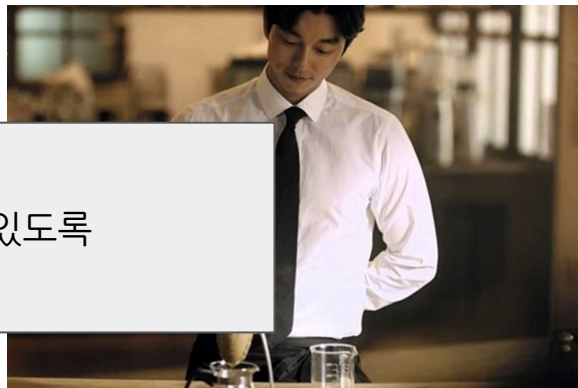
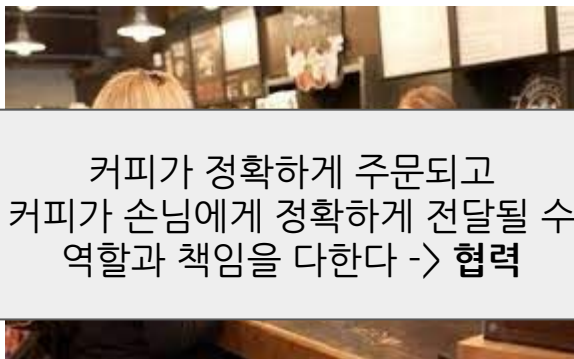


바리스타

커피 제조하는 바리스타
커피 제조하는 책임

현실세계 이야기

출근길 카페



커피가 정확하게 주문되고
주문된 커피가 손님에게 정확하게 전달될 수 있도록
역할과 책임을 다한다 -> 협력

노비

역할 커피를 주문하는 손님
책임 커피 주문할 책임

카페 캐시어

주문 받는 캐시어
주문 받는 책임

바리스타

커피 제조하는 바리스타
커피 제조하는 책임

현실세계 이야기

- 협력 = 요청 + 응답
 - 손님 → 캐시어 → 바리스타
 - 문제해결 위해 요청의 연쇄적 발생, 응답도 반대방향으로 연쇄적 전달
- 역할과 책임
 - 역할은 **책임**의 개념을 내포 ex) 손님 : 커피를 주문할 책임
 - 여러 사람이 동일한 역할을 수행할 수 있다
 - 역할은 대체 가능성을 의미
 - 책임을 수행하는 방법을 자율적으로 선택 가능
 - 한 사람이 동시에 여러 역할 수행 가능



객체에 현실 적용

- 사람 : 객체 / 요청 : 메시지 / 요청 처리 방법 : 메서드 => 객체지향
- 협력
 - “어떤 객체도 섬이 아니다” - 켄트 벅
 - 사람 : 공통의 목표를 달성 위해 협력
 - 객체 : 애플리케이션 기능 구현을 위해 협력
- 역할과 책임
 - 시스템 : 역할과 책임 수행하는 객체로 분할
 - 시스템의 기능 : 객체 간 연쇄적 요청과 응답의 흐름으로 구성된 협력으로 구현
 - 객체지향 설계 : 적절한 객체에게 적절한 책임을 할당하는 것

객체에 현실 적용

- 역할 특징
 - 여러 객체가 동일한 역할을 수행할 수 있다
 - 역할은 대체 가능성을 의미한다
 - 각 객체는 책임을 수행하는 방법을 자율적으로 선택할 수 있다
 - 하나의 객체가 동시에 여러 역할을 수행할 수 있다

객체에 현실 적용

- 협력의 품질 : 협력 품질 결정하는 건 객체의 품질
 - 1. 객체는 충분히 **협력적**이어야 한다
 - 명령복종(X) 요청 응답(O), 객체 스스로 결정
 - 2. 객체는 충분히 **자율적**이어야 한다
 - 자율적? 자신의 행동을 스스로 결정하고 책임지는 것
- 즉, 협력의 품질을 높이려면
 - 다른 객체와 조화롭게 협력할 수 있을 만큼 충분히 개방적이며
 - 협력에 참여하는 방법을 스스로 결정할 수 있을 만큼의 자율적인 객체

객체란?

- 객체 = 상태(state) + 행동(behavior)
 - 행동을 위해 필요한 상태 + 특정 행동 수행 방법을 스스로 결정
= 상태와 행위를 하나의 단위로 묶는 자율적 존재
- 메시지
 - 메시지 전송하는 객체(sender) + 메시지 수신하는 객체의 관계(receiver)
- 메서드와 자율성
 - 메서드 : 수신된 메시지 이해 여부 판단 후 자신만의 방법에 따라 메시지 처리
 - 메시지와 메서드의 분리 = 객체들간의 자율성 향상

정리

- 객체지향? 시스템을 상호작용하는 **자율적인 객체들의 공동체**로 바라보고 객체를 이용해 시스템을 분할하는 방법
- 자율적인 객체? **상태와 행위**를 함께 지니며 스스로 자기 자신을 책임지는 객체
- 객체? 시스템의 행위를 구현하기 위해 다른 객체와 **협력**, 각 객체는 협력 내에서 정해진 **역할**을 수행하며 역할은 관련된 **책임**의 집합
- 객체는 다른 객체와 협력하기 위해 메시지를 전송, **메시지**를 수신한 객체는 메시지를 처리하는데 적합한 **메서드**를 자율적으로 선택