

# 데이터베이스시스템 과제 #1

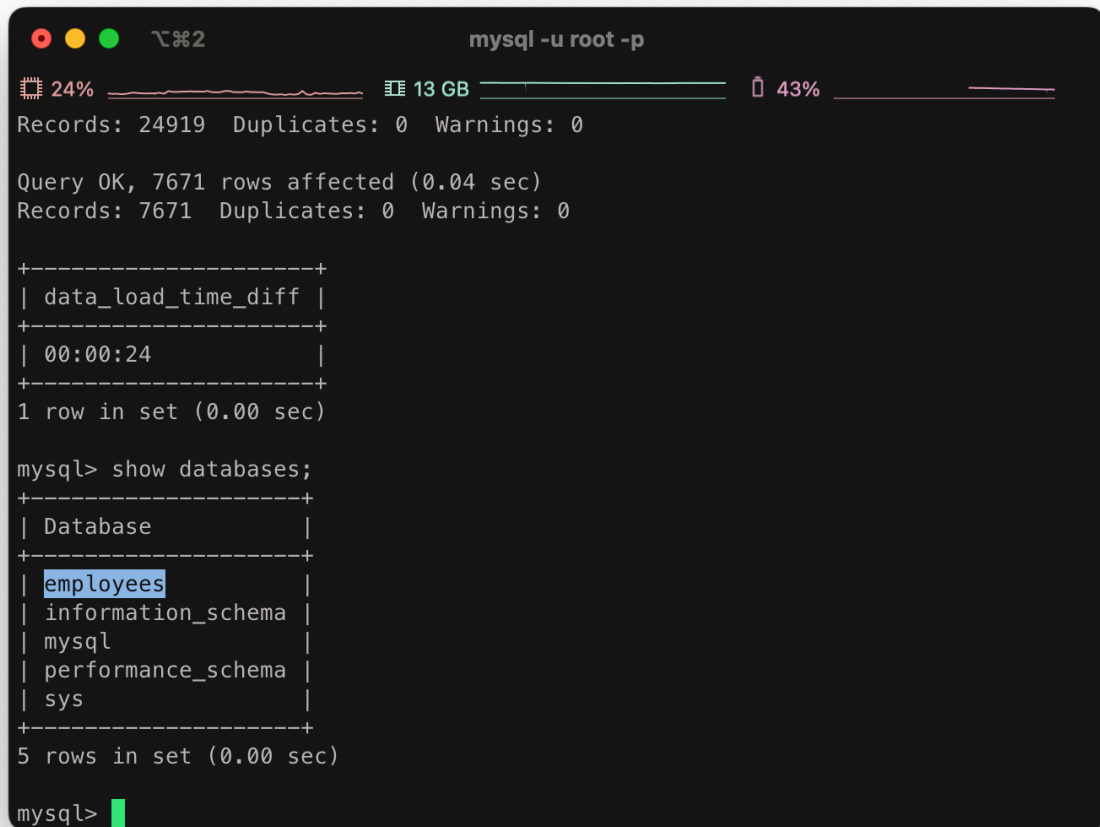
박종현, 2025-03-12

공과대학 컴퓨터정보통신공학과

## 과제 목표

1. 실습자료 1 수행 후 캡처 제출
  - 2.1. 객체지향의 철학에 대해 조사
  - 2.2. 객체지향이 가지는 장점이 무엇인지?
  - 2.3. 객체지향 프로그래밍이 가지는 특징

## 실습자료 1 수행 후 캡처 제출



```
mysql -u root -p
Records: 24919 Duplicates: 0 Warnings: 0

Query OK, 7671 rows affected (0.04 sec)
Records: 7671 Duplicates: 0 Warnings: 0

+-----+
| data_load_time_diff |
+-----+
| 00:00:24            |
+-----+
1 row in set (0.00 sec)

mysql> show databases;
+-----+
| Database |
+-----+
| employees |
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0.00 sec)

mysql>
```

## 객체지향의 철학

객체 지향은 해결하고자 하는 큰 문제를 계속해서 소분하여 더 작은 문제로 바라보는 동시에, 분리된 각 개념이 서로 상호작용하는 것에 초점을 두어 문제를 해결하는 것을 목표로 합니다. 특히 현실 세계의 다양한 주체, 객체들에 초점을 두어 이러한 것들의 특성을 분석하고 정리하거나, 각자의 행동과 상호작용을 모델링하여 전체 줄기의 큰 문제를 개별적인 작은 문제로 나눕니다.

### 1. 객체 중심 —

프로그램을 객체들의 집합으로 간주하여, 개별 객체의 역할을 명확하게 설정합니다. 이를 통해 프로그램으로 달성하려는 목표나 워크플로우가 복잡하거나 모호해지지 않도록 합니다.

## 2. 상호작용 —

모든 프로그램 흐름을 객체 사이의 상호작용으로 보고, 시스템 전체의 기능을 각 객체에 분리하여 할당합니다.

## 3. 추상화와 모델링 —

현실 세계의 복잡한 객체를 추상화하여 필요한 속성과 행동만을 추출하고, 이를 바탕으로 소프트웨어 모델을 구축합니다.

## 4. 재사용

덧붙임: 작품 「PSYCHO-PASS 3」와 함께 ——



일본의 애니메이션 「PSYCHO-PASS 3」에서는 정세를 유도하고 각종 공작을 벌여 부와 권력을 추구하는 비밀 결사 조직이 등장합니다. 작중에서 이 조직은 사람들이 만들어놓은 사소한 요인들과 공작 대상의 의사결정이 합쳐져, 최종적으로 큰 그림이 구성되도록 유도합니다.

이 과정에서 아무것도 모르는 일반 시민들도 결과적으로 범죄에 가담하게 되지만, 자신의 행동의 의미를 이해하지 못하고 전체 사건 중 일부의 비중을 갖게 되어 사회적으로 발각되지 않은 채 유지됩니다.

이렇게 공작 활동이라는 큰 문제를 작은 문제로 나누고 개별 객체 간의 상호작용으로 바라봄으로서 목표 달성을 도모한 것과 마찬가지로, 객체지향 패러다임의 주요한 철학이나 핵심 원리도 각 객체의 상호작용으로 문제 해결 도모가 골자라고 할 수 있습니다.

## 객체지향의 장점

### 1. 코드 재사용성 증대 —

객체 지향은 이미 만들어진 클래스를 상속받아 새로운 클래스를 만들 수 있기 때문에 코드의 재사용성이 높아집니다. 이는 개발 시간을 단축시키고, 코드의 중복을 줄여 유지보수를 용이하게 합니다.

### 2. 유지보수 용이성 증대 —

객체 지향은 캡슐화를 통해 데이터와 기능을 묶어 관리하므로, 특정 객체의 내부 구현이 변경되어도 다른 객체에 미치는 영향을 최소화할 수 있습니다. 이는 코드의 수정 및 디버깅을 용이하게 하고, 시스템의 안정성을 높여줍니다.

### 3. 확장성 증대 —

객체 지향은 새로운 기능을 추가하거나 기존 기능을 수정하는 것이 용이하여 시스템의 확장성을 높여줍니다. 이는 변화하는 요구사항에 유연하게 대처할 수 있도록 해줍니다.

### 4. 현실 세계 모델링 용이 —

객체 지향은 현실 세계의 객체와 개념을 직접적으로 모델링할 수 있어 소프트웨어 시스템을 더 쉽게 이해하고 설계할 수 있도록 합니다. 이는 사용자 요구사항을 정확하게 반영하고, 시스템의 완성도를 높여줍니다.

## 5. 모듈화 촉진 —

객체 지향은 객체를 독립적인 모듈로 구성하여 코드의 재사용성을 높이고 유지 보수를 용이하게 합니다. 이는 팀 단위의 개발 작업을 효율적으로 수행할 수 있도록 해줍니다.

## 객체지향 프로그래밍이 가지는 특징

### 1. 추상화 (Abstraction) —

객체의 불필요한 부분을 숨기고 핵심적인 부분만을 드러냄으로써 복잡성을 관리합니다.

### 2. 캡슐화 (Encapsulation) —

데이터와 데이터를 다루는 메서드를 하나의 객체 안에 묶어 외부로부터 데이터를 보호하고, 객체 내부의 정보 은닉을 가능하게 합니다.

### 3. 상속 (Inheritance) —

기존 클래스의 속성과 메서드를 다른 클래스가 물려받아 코드의 재사용성을 높이고, 클래스 간의 계층 구조를 형성합니다.

### 4. 다형성 (Polymorphism) —

같은 이름의 메서드가 다른 방식으로 동작할 수 있도록 하여 코드의 유연성과 확장성 높입니다.