

# Fall 2018: COMP322003 Introduction to Database (데이터베이스)

## Term Project: 쇼핑몰 'X' 구축

### Phase 2: 관계형 데이터베이스 구축

**제출기한: 2018년 10월 18일 오전 8시 59분 59초**

#### [개요]

이번 단계에서는 관계형 데이터베이스를 구축하도록 한다. 1단계에서 생성한 개념 스키마를 바탕으로 데이터베이스를 구축하고, 쇼핑몰의 현 상태에 대한 데이터를 생성하여 입력하는 것이 주요한 task이다.

1단계에서 생성한 결과물과 다소 차이가 있을 수 있으며, 해당 부분에 대해서는 감안을 하여 평가가 진행된다. 평가를 위하여 데이터베이스에 입력될 정보는 **최소한의 제약 사항**이 존재하며, 제약 사항 이외의 조건들은 스스로 판단을 하여야 한다. 생성할 데이터의 양이 **제법** 많으므로, 간단한 데이터 생성 및 적재 애플리케이션 프로그램을 작성해서 활용해도 좋다. (해당 데이터 생성/적재 프로그램은 제출할 필요가 없다. 단, 관련 제출물은 반드시 .sql 파일이어야 한다.) 데이터 생성 후, 데이터베이스에 입력을 한 것을 확인하기 위하여 최소한의 질의문을 구현하여야 한다. 이 최소한의 질의문을 바탕으로 2단계가 평가된다.

#### [과제 목적]

- 1) ER 스키마가 관계형 스키마로 어떻게 매핑될 수 있는지 이해한다.
- 2) (실습 때 사용되는 상용 DBMS와는 별도로) 오픈소스 데이터베이스 관리 시스템인, MySQL DBMS을 경험토록 한다.
- 3) DBMS Vendor들에 따라 SQL의 실제 구현에 있어서 Variation이 있음을 이해한다.
- 4) 리눅스 기반의 데이터베이스 설치 및 응용 프로그램 제작을 경험토록 한다.

## [관계형 데이터베이스 설계 - 테이블 생성]

Phase 1에서 제출한 ER 스키마에 따라 관계형 데이터베이스 구축을 위한 테이블들을 생성하라. 수업 시간에 다룬 것처럼 ER-to-Relational 매핑 알고리즘을 활용토록 한다. 만약 매핑 과정동안 ER 스키마에서 문제점을 발견하였다면 수정토록 한다. 단 수정 내용이 반드시 매핑 기술 파일에 적시되어야 한다. 최종 수정된 ER 스키마에 근거 하여, CREATE TABLE 문을 이용해 테이블을 생성한다. 테이블 생성 시, DDL에 반드시 테이블 내, 혹은 테이블 간 제약 조건도 명시할 것.

## [관계형 데이터베이스 구축 - 데이터 생성 상세 사항 및 제약조건]

본 절에서는 실제 데이터베이스에서 사용될, 또한 평가 기준으로 이용될 데이터를 생성에 관한 내용이다. 평가를 위하여 최소한의 데이터가 필요하며, 이를 서술하였다. 아래의 내용을 토대로 데이터를 생성하여야 하며, 데이터 생성 프로그램을 만들어도 좋다. 아래의 내용에 명시되지 않은 사항에 대해서는 팀별로 임의로 설정토록 한다.

### 1. 상품 (ITEM)

설명	제약 사항
재고가 0인 상품	최소 3개 이상
재고가 1~20개인 상품	최소 10개 이상
재고가 1000개 이상인 상품	최소 20개 이상
그 외	재고를 0 ~ 5000안의 값으로 임의적으로 지정하도록 한다.

### 2. 배송업체 (SHIPPING COMPANY)

설명	제약 사항
각 배송업체 별 배송 횟수	각 최소 50회 이상

### 3. 고객 (CUSTOMER)

고객 설명	제약 사항
기본 제공 정보만 입력한 고객	최소 15인 이상
성별, 나이 정보를 추가 입력한 고객	최소 10인 이상
성별, 나이, 직업 정보, 타입을 추가 입력한 고객	최소 30인 이상
그 외	전체 고객 수는 최소 60인 이상이다..

고객 설명	제약 사항
주문을 한 번도 하지 않은 고객 수	3~5인
주문을 7번 이상 진행한 고객 수	최소 30인 이상
그 외	나머지 고객은 주문 횟수를 1~20번의 사이에서 적절히 고르면 된다.

### 4. 장바구니 (SHOPPINGBAG)

설명	제약 사항
장바구니에 상품을 15개 이상 채운 고객 수	최소 15인 이상
장바구니에 상품을 1~14개 채운 고객 수	최소 10인 이상
장바구니에 상품이 하나도 없는 고객 수	최소 20인 이상

## 5. 주문 (ORDER)

설명	제약 사항
총 주문 횟수	최소 200건 이상
주문 최소 단위	한 주문은 최소 물건 한 개 이상 주문을 하여야 한다.
주문 시기	주문이 진행된 시기는 임의로 정하나, 최소 2년의 기록을 가지게 하며, 각 월별 주문은 최소 3회 이상 진행되어야 한다.
제품 개수	도매, 소매업을 하는 사람의 경우 주문 시, 대량의 제품(여러 개의 물품, 한 물품당 많은 수)를 주문하게 한다.

### [필수 구현 질의문]

본 절은 설계한 관계형 데이터베이스가 옳게 정의되고 적절히 Population 되었는지 검수하기 위한 최소한의 질의문을 설술한다. 각 팀은 아래의 질의 문들을 반드시 SQL로 구현해야 한다.

Q1	전체 재고가 20 개 이하인 제품은 몇 개인가?
Q2	현재 가입한 고객의 수는 몇 명인가?
Q3	주문을 7 번 이상 진행한 고객은 누구인가?
Q4	지난 6 개월간 가장 많은 주문 수를 기록한 브랜드 Top3 와 그 브랜드를 공급하는 공급업체들은 어느 것인가?
Q5	주문 횟수가 7 번 이상인 고객들이 가장 많이 주문한 제품은 무엇인가?
Q6	장바구니에 상품이 하나도 없는 고객들을 대상으로 과거 매출 Top 5 는 무엇인가?
Q7	소매업을 하는 고객을 대상으로 가장 높은 매출을 낸 제품의 브랜드는 무엇이며, 그를 공급하는 업체가 취급하는 다른 브랜드에 대해 각각 브랜드 별 소매업대상 가장 높은 매출을 낸 제품은 무엇인가?
Q8	가장 많은 배송을 진행한 배송업체가 가장 많이 배송한 중분류는 무엇이며, 그 중분류를 가장 자주 주문하는 고객의 타입은 무엇인가?
Q9	현재 장바구니에 담긴 모든 물품이 주문된다고 하였을 때, 전체 재고가 모자란 제품은 무엇인가?
Q10	현재까지 가장 높은 매출을 낸 제품을 장바구니에 담고 있는 고객의 정보와, 그 제품의 같은 카테고리 (대분류, 중분류, 소분류가 모두 같은)의 제품들을 담고 있는 고객의 정보를 출력하라.

## [개발 환경]

- OS: Linux (Ubuntu 18.04) (실습실 PC 또는 가상 머신 활용)
- 가상 머신 활용 시, 아래 자료 참고.
  - o Ubuntu Desktop:
    - Ubuntu 18.04.1 LTS:  
<https://www.ubuntu.com/download/desktop>
  - o Virtual Machine Hypervisor:
    - VirtualBox:  
<https://www.virtualbox.org/wiki/Downloads>
    - VMware Player:  
[https://my.vmware.com/en/web/vmware/free#desktop\\_end\\_user\\_computing/vmware\\_workstation\\_player/15\\_0](https://my.vmware.com/en/web/vmware/free#desktop_end_user_computing/vmware_workstation_player/15_0)
- DBMS Server: MySQL 5.7.21-1ubuntu1

## [관련 자료]

- Ubuntu에 MySQL 설치:  
<https://www.digitalocean.com/community/tutorials/how-to-install-mysql-on-ubuntu-18-04>  
<https://support.rackspace.com/how-to/installing-mysql-server-on-ubuntu/>  
<https://opentutorials.org/module/1256/8238>  
<https://opentutorials.org/course/195/1465>  
기타 구글링 할 것.
- MySQL 튜토리얼  
[https://www.w3schools.com/php/php\\_mysql\\_intro.asp](https://www.w3schools.com/php/php_mysql_intro.asp)  
<https://www.tutorialspoint.com/mysql/>  
<http://www.mysqltutorial.org/basic-mysql-tutorial.aspx>  
<http://dreamlog.tistory.com/534>  
기타 구글링 할 것.
- MySQL 클라이언트에서 \*.sql 스크립트 실행 방법  
<http://wwwi.tistory.com/83>

## [Phase 2-제출 정보]

- 제출기한: **2018년 10월 18일 오전 8시 59분 59초**
- **제출물**: 크게 3가지 + 추가 파일들.
  - 1) **TeamX-P2.sql**: 팀별 관계형 데이터베이스 생성 SQL 스크립트.  
[중요] 스크립트에는 다음의 내용을 반드시 포함해야 함.
    - 해당 .sql 스크립트 파일에 제약조건들을 포함한 DDL
    - DDL을 통해 생성된 테이블에 대한 INSERT문들
    - Q1 ~ Q10까지의 Query문들위의 SQL이 순서대로 작성되어 실행될 수 있도록 할 것.  
[MySQL 클라이언트에서 스크립트 실행 방법]  

```
mysql> source TeamX-P2.sql
```
  - 2) **팀x-etr\_mapping.txt**: 팀별 ER 스키마가 어떻게 관계형 스키마로 매핑되었는지 관계 순으로 설명할 것. 만약 ER 스키마에 수정사항이 있으면 반드시 명시할 것
  - 3) **팀x-README.txt**: SQL 스크립트 동작 환경 (OS 버전, MySQL 버전)
  - 4) 수정된 ER 스키마 파일
  - 5) 기타 팀별 추가 파일들, if any.
- **배점**: (30점 / 100점)
- **주의 사항**:
  - o 반드시 위에서 기술된 **개발 환경** (MySQL on Ubuntu) 준수하도록 한다.
  - o 만약 팀이 Phase 1에서 제출한 ER 스키마에 변동이 있을 시, 반드시 해당 사항을 기술하고 **\*-etr-mapping.txt**에 기술하고 해당 수정 ER 스키마 파일을 제출할 것.
- **제출 방법**: LMS 팀 프로젝트 메뉴에서 Phase 2 게시물에 팀별로 결과물 제출

## [Q&A]

질문 있으면 강의 교수([yksuh@knu.ac.kr](mailto:yksuh@knu.ac.kr))나 TA ([kshy9598@knu.ac.kr](mailto:kshy9598@knu.ac.kr))의 이메일로 연락할 것.  
혹은 사전 약속 후, 연구실로 방문할 것.

## [Late Submission]

하루 당 50%씩 감점. 단, 상황에 따라 기한 연장을 허용할 수 있음.

## [Plagiarism]

절대 허용되지 않음. 발견되는 즉시 남은 결과에 상관없이 “F”학점 부여.