

혼자 공부하는 SQL

01-1 데이터베이스 알아보기

1. ① 데이터의 저장소 또는 데이터의 집합을 말합니다. 약자로 DB라고 부릅니다.
 - 데이터베이스
 - ② 국제표준화기구에서 지정하며, RDBMS에서 사용되는 언어를 말합니다.
 - MySQL
 - ③ 대표적인 DBMS로 데이터를 구축, 관리하기 위해 SQL을 사용합니다.
 - 테이블
 - ④ 표 형태로 구성되었으며 열과 행으로 이루어져 있습니다.
 - 표준 SQL
2. ②
 3. ① 표준 SQL, ② PL/SQL, ③ T-SQL, ④ SQL

01-2 MySQL 설치하기

1. ①
2. ③

MySQL의 기능 추가 및 제거는 윈도우즈의 [앱 및 기능] 또는 [MySQL Installer]에서 할 수 있습니다.
3. SHOW DATABASES

02-1 건물을 짓기 위한 설계도: 데이터베이스 모델링

1. ① 현실 세계를 컴퓨터 시스템으로 옮겨놓는 일련의 과정을 일컫습니다.
 - 폭포수 모델
- ② 소프트웨어 개발 절차 중 하나로 폭포가 떨어지듯 각 단계가 진행됩니다.
 - 프로젝트
- ③ 소프트웨어를 완성하는 절차를 연구하는 분야를 통틀어서 이렇게 부릅니다.
 - 소프트웨어 공학

2. 프로젝트 계획 → 업무 분석 → 시스템 설계 → 프로그램 구현 → 테스트 → 유지보수

3. 데이터베이스 모델링

4. ① 회원이나 제품의 데이터를 입력하기 위해 표 형태로 표현한 것을 말합니다. 가로와 세로로 구성되어 있습니다. • DBMS
- ② 데이터베이스를 관리하는 시스템 또는 소프트웨어를 말합니다. • 행
- ③ 실질적인 진짜 데이터를 말합니다. 테이블의 가로에 해당합니다. • 테이블
- ④ 사람과 DBMS가 소통하기 위한 말(언어)입니다. • SQL

02-2 데이터베이스 시작부터 끝까지

1. 데이터베이스 만들기 → 테이블 만들기 → 데이터 입력하기 → 데이터 조회하기

2. ④

3. ① 데이터를 수정할 때 사용 • CREATE
- ② 데이터를 조회할 때 사용 • UPDATE
- ③ 테이블이나 데이터베이스를 만들 때 사용 • DELETE
- ④ 데이터를 삭제할 때 사용 • SELECT
4. ① 데이터 형식 중에서 소수점이 없는 정수형 • CHAR
- ② 비어 있는 값을 허용하지 않음 • INT
- ③ 데이터 형식 중에서 문자형 • DATE
- ④ 데이터 형식 중에서 날짜형 • NOT NULL

5. ①

02-3 데이터베이스 개체

1. 열 이름, 데이터 형식, 기본 키
2. ②, ③
인덱스는 데이터의 건수가 많아져 체감할 수 있습니다.
3. ②
4. ④, ⑤

03-1 기본 중에 기본 SELECT ~ FROM ~ WHERE

1. ④
회원 테이블(member)과 구매 테이블(buy)은 서로 PK, FK 관계로 연결되어 있습니다.
2. ②, ③
MySQL 워크벤치를 재시작하거나, 새 쿼리 창을 열면 USE를 재지정해야 합니다.
3. ①
4. ①, ③

03-2 좀 더 깊게 알아보는 SELECT 문

1. SELECT, FROM, WHERE, ORDER BY, LIMIT
2. ① ORDER BY, ② LIMIT, ③ DISTINCT
3. ① ASC, ② DESC
4. ③
5. DISTINCT
6. ③

03-3 데이터 변경을 위한 SQL 문

1. ④

2. ③, ④

AUTO_INCREMENT는 직접 값을 입력할 수 없으며, 입력할 위치에 NULL이라고 표기해야 합니다.

3. ①

4. ②

5. ④

6. TRUNCATE

04-1 MySQL의 데이터 형식

1. TINYINT, SMALLINT, INT, BIGINT

2. ②

3. ②

4. ③

CHAR는 최대 255자까지 저장됩니다.

5. ①, ②

6. ①

7. CONVERT(), CAST()

04-2 두 테이블을 묶는 조인

1. ③

2. ① 가장 많이 사용되는 조인으로, 일반적으로 부르는 조인이다. • 자체 조인

② 한쪽 테이블에만 데이터가 있어도 결과가 나오는 조인이다. • 상호 조인

③ 한쪽 테이블의 모든 행과 다른 쪽 테이블의 모든 행을 조인시킨다. • 내부 조인

④ 한 개의 테이블이 자신과 조인되는 것을 말한다. • 외부 조인

3. ③

4. ④

5. ① 상호 조인, ② 자체 조인

04-3 SQL 프로그래밍

1. ②

2. ②

3. ① WHEN, ② CASE

4. ① WHILE, ② ITERATE, ③ LEAVE

05-1 테이블 만들기

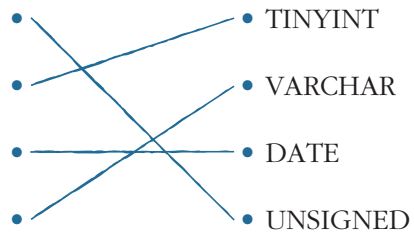
1. CHAR, VARCHAR

2. ① 정수형 데이터를 0부터 입력되도록 설정합니다.

② -128 ~ +127까지 값이 저장됩니다.

③ '2022-11-12'와 같은 데이터가 저장됩니다.

④ 가변형 문자형으로 짧거나 긴 문자가 뒤죽박죽
입력될 때 적절합니다.



3. ③

UNSIGNED는 UN 부분을 체크합니다.

4. ③

기본 키와 외래 키는 일반적으로 다른 테이블에 설정합니다.

5. 2행, 8행

2행은 PRIMARY KEY, 8행은 TINYINT UNSIGNED 또는 SMALLINT, INT, BIGINT로 설정해야 합니다.

05-2 제약조건으로 테이블을 견고하게

1. ③

2. ④

PRIMARY KEY로 설정하면 자동으로 NOT NULL이 되므로 생략해도 됩니다.

3. ④

4. ① ON DELETE CASCADE, ② ON UPDATE CASCADE

5. ① CHECK, ② DEFAULT, ③ NOT NULL

05-3 가상의 테이블: 뷰

1. ②

2. ①

뷰는 테이블에서 필요한 열만 골라서 포함시킬 수 있습니다.

3. ④

별칭에 공백이 없으면 백틱으로 묶지 않아도 됩니다.

4. ②

5. ① SHOW CREATE VIEW, ② WITH CHECK OPTION, ③ CHECK TABLE

06-1 인덱스 개념을 파악하자

1. ②

2. ③

인덱스는 약 10%의 추가 공간이 필요합니다.

3. ④

보조 인덱스는 고유 키를 설정하면 자동 생성됩니다.

4. ①

인덱스 값이 중복되는 것을 허용하는 인덱스는 단순 인덱스입니다.

5. ②, ④

클러스터형 인덱스는 테이블에 1개만 생성 가능합니다.

06-2 인덱스의 내부 작동

1. ④

2. ① 노드 중 제일 상위 노드를 말합니다.

• 루트 노드

② 노드 중 가운데 낀 노드를 말합니다.

• 리프 노드

③ 노드 중 제일 마지막 노드를 말합니다.

• 중간 노드

④ 16KB 크기의 최소한의 저장 단위입니다.

• 페이지

3. 페이지 분할

4. ②, ③

클러스터형 인덱스로 지정하면 오름차순 정렬됩니다. 보조 인덱스는 생성해도 정렬되지 않습니다.

06-3 인덱스의 실제 사용

1. ②

2. ① 인덱스를 생성하는 SQL

• SHOW INDEX

② 인덱스를 제거하는 SQL

• DROP INDEX

③ 테이블에 생성된 인덱스 이름과 열을 확인하는 SQL

• CREATE INDEX

④ 인덱스의 할당된 크기를 확인하는 SQL

• SHOW TABLE STATUS

3. ①, ②

① 클러스터형 인덱스는 1개, 보조 인덱스는 여러 개 만들 수 있습니다.

② 중복된 값이 있으면 CREATE UNIQUE INDEX 문은 오류가 발생합니다.

4. ①, ③

① SQL을 실행한 후에 확인할 수 있습니다.

③ 인덱스를 사용하면 Single Row, Index Range Scan 등 다양한 형태로 표시됩니다.

5. ①, ③

- ① 인덱스는 열 단위에 생성됩니다.
- ③ 중복도가 높으면 인덱스의 효과가 없습니다.

07-1 스토어드 프로시저 사용 방법

1. ④

스토어드 프로시저는 데이터베이스 내부에 저장됩니다.

2. ④

스토어드 프로시저의 형식은 다음과 같습니다.

```
DELIMITER $$  
CREATE PROCEDURE 스토어드_프로시저_이름( IN 또는 OUT 매개변수 )  
BEGIN  
  
    이 부분에 SQL 프로그래밍을 코드를 작성  
  
END $$  
DELIMITER ;
```

3. ④

출력 매개변수는 CALL 프로시저_이름(@변수명) 형식을 사용합니다.

4. ②

반복문은 WHILE을 사용합니다.

07-2 스토어드 함수와 커서

1. ④

스토어드 함수에서는 SELECT를 사용할 수 없습니다.

2. ① RETURNS, ② RETURN

3. 커서 선언하기 → 반복 조건 선언하기 → 커서 열기 → 데이터 가져오기 및 데이터 처리하기
→ 커서 닫기

4. ① LOOP, ② FETCH, ③ LEAVE

07-3 자동 실행되는 트리거

1. ①

SELECT는 트리거를 작동시키지 않습니다.

2. ④

트리거는 DML (INSERT, UPDATE, DELETE) 문의 이벤트가 발생하면 자동으로 작동합니다.

3. ① AFTER, ② ROW

4. ②

AFTER DELETE 트리거는 오직 DELETE 문에서만 작동합니다.

5. ④

TRUNCATE는 트리거를 작동시키지 못합니다.

08-1 파이썬 개발 환경 준비

1. ③

파이썬은 완전 무료입니다.

2. ③

3. ④

파이썬은 스크립트 언어(인터프리트 언어)입니다.

4. ④

08-2 파이썬과 MySQL의 연동

1. ④

commit()으로 저장한 후에 close()로 종료해야 합니다.

2. ④

charset에는 문자셋(utf8)를 적어 줍니다.

3. ③

커서.execute("INSERT INTO ~~")가 데이터를 입력하는 문장입니다.

4. ②

08-3 GUI 응용 프로그램

1. ③

title()은 윈도우의 제목을 표시합니다.

2. ④

bg는 background의 약자로 배경색을 지정합니다.

3. ③

command가 함수명을 지정하는 옵션입니다.

4. ③