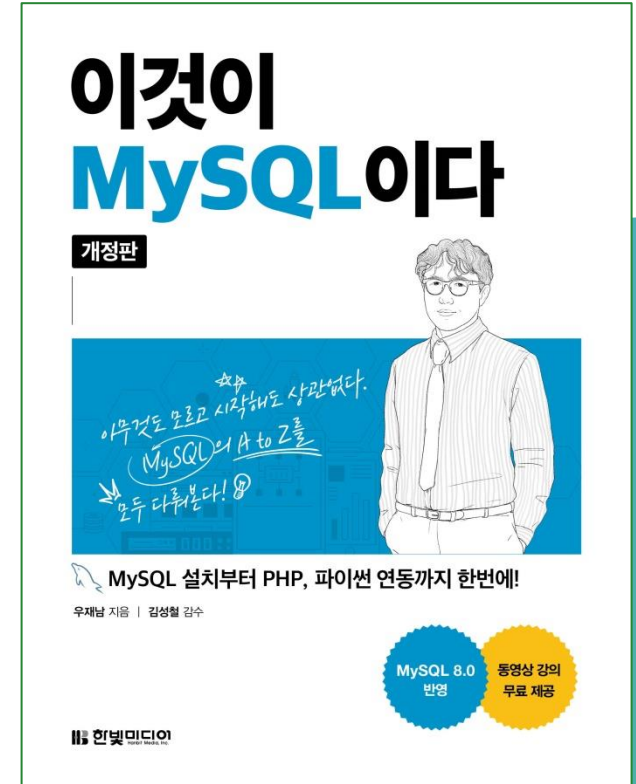


이것이 MySQL이다



저자 : 우재남

- CHAPTER 03 MySQL 전체 운영 실습
 - SECTION 01 요구사항 분석과 시스템 설계 그리고 모델링
 - 1.1 정보시스템 구축 절차 요약
 - 1.2 데이터베이스 모델링과 필수 용어
 - SECTION 02 MySQL을 이용한 데이터베이스 구축 절차
 - 2.1 데이터베이스 생성
 - 2.2 테이블 생성
 - 2.3 데이터 입력
 - 2.4 데이터 활용

- CHAPTER 03 MySQL 전체 운영 실습
 - SECTION 03 테이블 외에 데이터베이스 개체의 활용
 - 3.1 인덱스
 - 3.2 뷰
 - 3.3 스토어드 프로시저
 - 3.4 트리거
 - SECTION 04 데이터베이스 백업 및 관리
 - 4.1 백업과 복원
 - SECTION 05 MySQL과 응용 프로그램의 연결



CHAPTER 03 MySQL 전체 운영 실습

실무에서 발생하는 상황과 비슷한 설정을 하여 응용프로그램과 연동해본다.

SECTION 01 요구사항 분석과 시스템 설계 그리고 모델링

정보시스템 구축 절차 요약

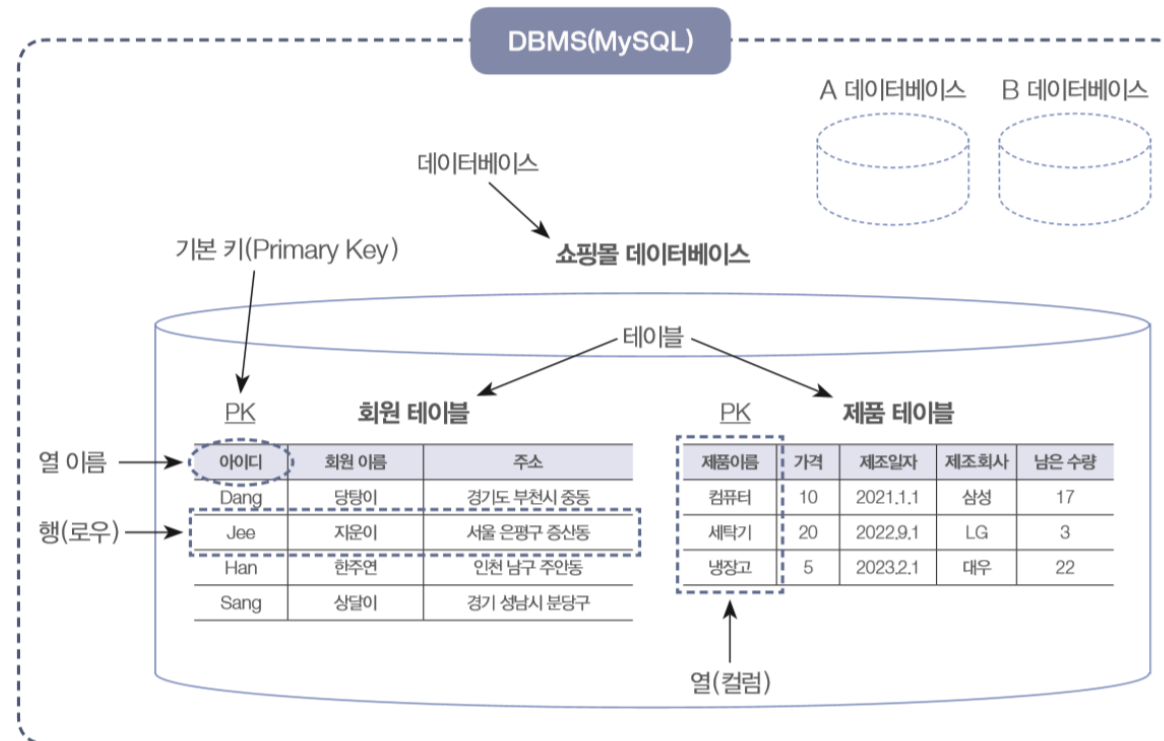
- 분석, 설계, 구현, 시험, 유지보수의 5가지 단계
- 분석
 - 구현하고자 하는 프로젝트의 가장 첫 번째 단계
 - 시스템 분석 또는 요구사항 분석이라고 불림
 - 요구사항 분석은 현재 우리가 '무엇을(What)' 할 것인지 결정
 - 사용자의 인터뷰와 업무 조사 등을 수행
 - 프로젝트의 첫 단추를 끼우는 중요한 단계
 - 분석의 결과로 많은 문서 작성
- 설계
 - 시스템 설계 또는 프로그램 설계
 - 구축하고자 하는 시스템을 '어떻게(How)' 할 것인지 결정
 - 대부분의 프로젝트에서 분석과 설계의 과정이 전체 공정의 50% 이상 차지

SECTION 01 요구사항 분석과 시스템 설계 그리고 모델링

데이터베이스 모델링과 필수 용어

데이터베이스 모델링

- 현실세계에서 사용되는 데이터를 MySQL에 어떻게 옮겨 놓을 것인지를 결정하는 과정
- 저장할 정보는 테이블(Table)이라는 형식에 맞춰 저장
- Ex) 쇼핑몰 데이터 베이스의 예



SECTION 01 요구사항 분석과 시스템 설계 그리고 모델링

데이터베이스 모델링과 필수 용어

- 데이터
 - 하나하나의 단편적인 정보
 - 정보는 있으나 아직 체계화 되지 못한 상태
- 테이블
 - 데이터를 입력하기 위해, 표 형태로 표현한 것
 - Ex) 회원 정보 테이블, 제품 정보 테이블
- 데이터베이스(DB)
 - 테이블이 저장되는 저장소
 - 각 데이터베이스는 서로 다른 고유한 이름을 가지고 있음
- DBMS (DataBase Management System)
 - 데이터베이스를 관리하는 시스템 또는 소프트웨어

SECTION 01 요구사항 분석과 시스템 설계 그리고 모델링

데이터베이스 모델링과 필수 용어

- 열(=컬럼=필드)
 - 각 테이블은 열로 구성
 - 회원 테이블의 경우에는 아이디, 회원 이름, 주소 등 3개의 열로 구성
- 열 이름
 - 각 열을 구분하기 위한 이름
 - 열 이름은 각 테이블 내에서는 중복되지 않고, 고유해야 함
- 데이터 형식
 - 열의 데이터 형식
 - 테이블을 생성할 때 열 이름과 함께 지정
- 행(=로우=레코드)
 - 실질적인 데이터
 - 회원 테이블의 경우 4건의 행 데이터, 즉 4명의 회원이 존재함

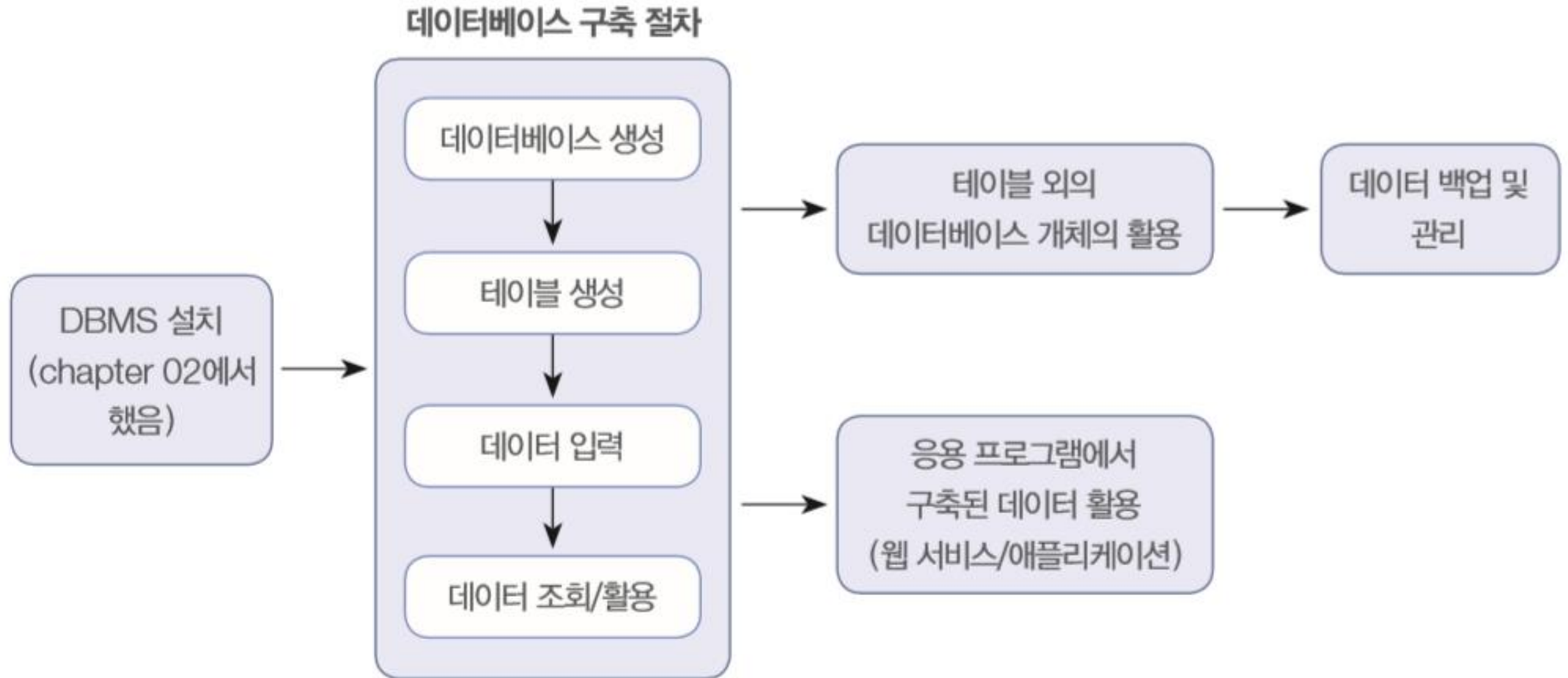
SECTION 01 요구사항 분석과 시스템 설계 그리고 모델링

데이터베이스 모델링과 필수 용어

- 기본 키 (Primary Key) 열
 - 기본 키(또는 주 키) 열은 각 행을 구분하는 유일한 열
 - 중복되어서는 안되며, 비어 있어서도 안 됨
 - 각 테이블에는 기본 키가 하나만 지정
- 외래 키(Foreign Key) 필드
 - 두 테이블의 관계를 맺어주는 키
 - 4장 이후 설명
- SQL (Structured Query Language)
 - 구조화된 질의 언어
 - 사람과 DBMS가 소통하기 위한 말(언어)
 - 6, 7장에서 자세히 다룸

SECTION 02 MySQL을 이용한 데이터베이스 구축 절차

데이터베이스 구축/관리 및 활용의 전반적인 절차



[그림 3-2] 데이터베이스 구축/관리 및 활용의 전반적인 절차

SECTION 02 MySQL을 이용한 데이터베이스 구축 절차

인터넷 쇼핑몰 구축 위한 '쇼핑몰' DB 생성

- MySQL 서버 연결 및 설정

- Windows의 [시작] >> [M] >> [MySQL] >> [MySQL Workbench 8.0 CE] 클릭해 Workbench 실행
- [MySQL Connections] 창에서 비밀번호 입력하여 접속
- Workbench의 초기 창
 - 기본적으로는 [Schemas] 탭 클릭해놓고 사용
- Workbench 종료 → 설정 저장

SECTION 02 MySQL을 이용한 데이터베이스 구축 절차

인터넷 쇼핑몰 구축 위한 '쇼핑몰' DB 생성

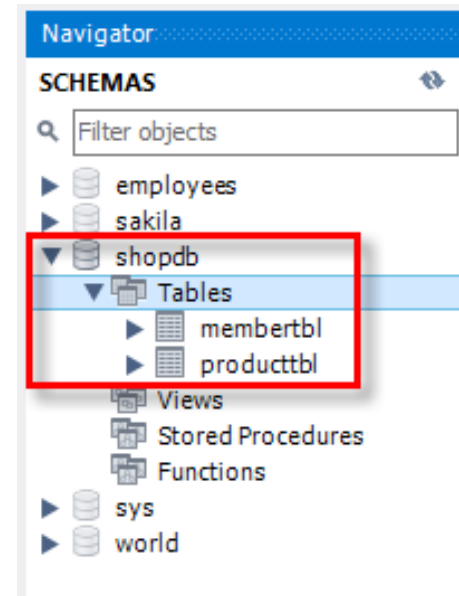
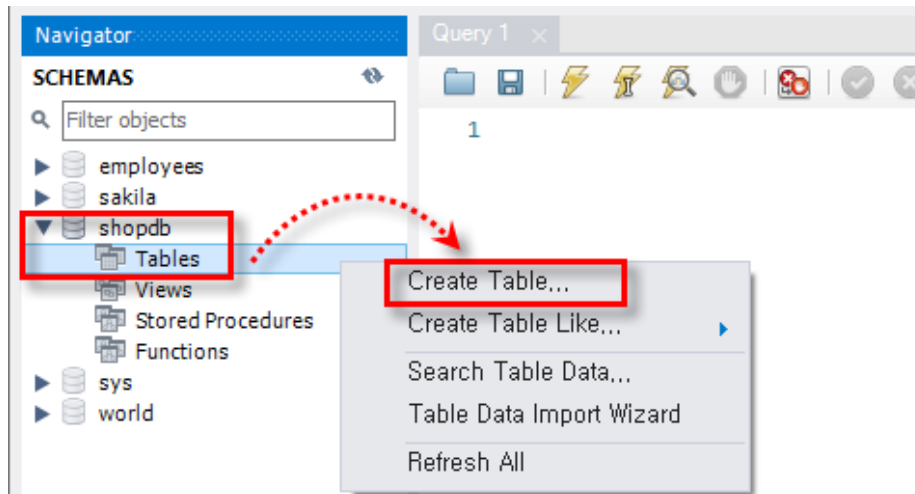
- 스키마 (Schema) 생성
 - MySQL에서는 스키마와 데이터베이스가 완전히 동일한 용어로 사용
 - Workbench의 [SCHEMAS]의 빈 부분
 - 마우스 오른쪽 버튼 클릭 후 [Create Schema](=Create Database) 선택
 - CREATE SCHEMA 'shopdb'문을 쿼리 창에서 입력하는 것과 동일한 작동
 - 이름 입력하면 DB 생성
 - 왼쪽 데이터베이스 목록에 shopdb 데이터베이스 확인
 - 아무것도 들어있지 않은 데이터베이스 생성

SECTION 02 MySQL을 이용한 데이터베이스 구축 절차

인터넷 쇼핑몰 구축 위한 '쇼핑몰' DB 생성

테이블 생성

- 회원테이블, 제품 테이블 각 열의 영문 이름 및 데이터 형식 결정
 - 데이터베이스 모델링(특히 물리적 모델링) 시에 결정
 - 데이터 형식의 자세한 내용은 7장에서 학습
- 형식이 정해지면 Create Table 실행해 테이블이름, 열이름, 데이터형식등 테이블 내용 입력
- 생성된 SQL Query를 데이터베이스에 적용해 테이블 생성 완료
- ShopDB의 [테이블]에서 생성한 테이블 확인

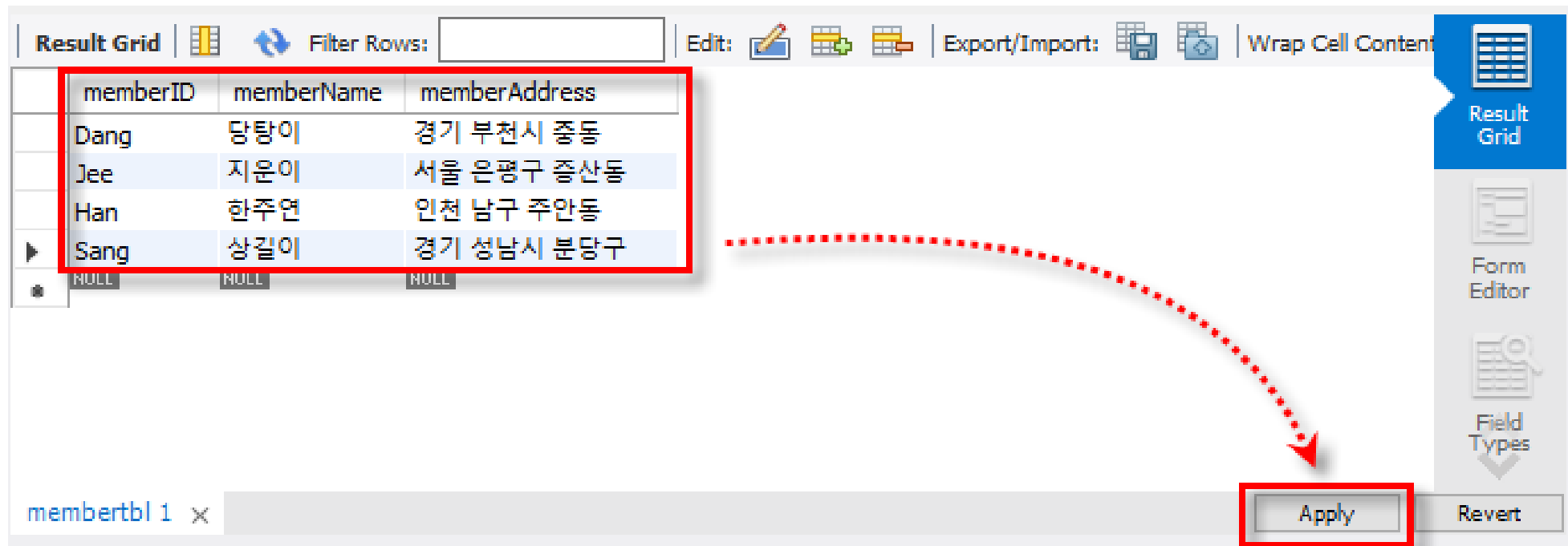


SECTION 02 MySQL을 이용한 데이터베이스 구축 절차

데이터 입력 - 행 데이터 입력

- 회원 테이블의 데이터 입력

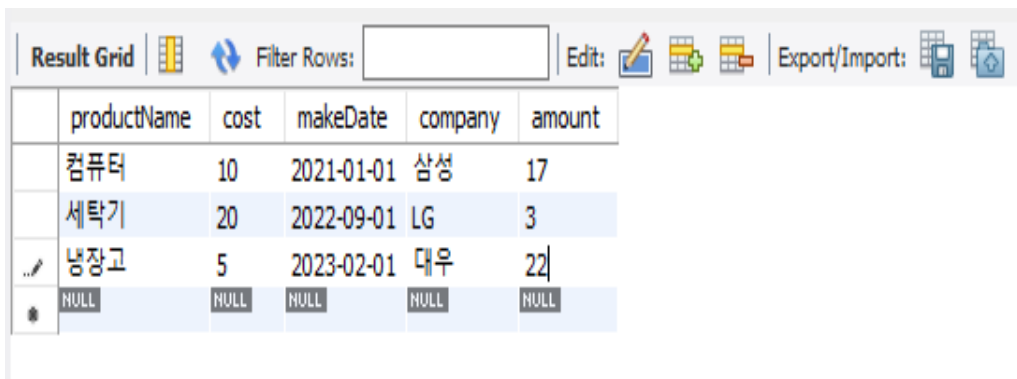
- Navigator의 [SCHEMAS]에서 [Shopdb] >> [Tables] >> [membertbl] 선택 후, 마우스 오른쪽 버튼 클릭하고 [Select Rows - Limits 1000] 선택
- 아래 그림의 회원 테이블 데이터 Grid에 입력 후 Apply 해 저장



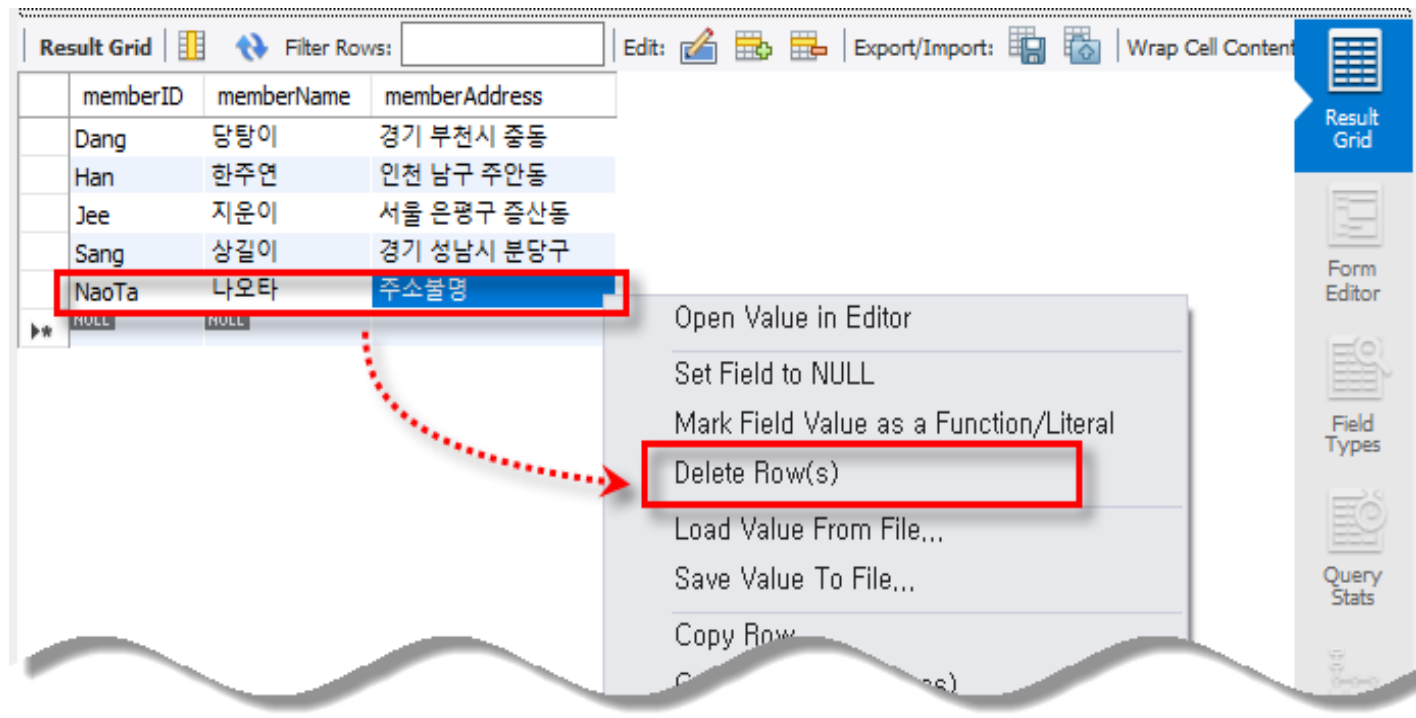
SECTION 02 MySQL을 이용한 데이터베이스 구축 절차

데이터 입력 – 행 데이터 입력

- 제품 테이블의 데이터 입력
 - 동일한 방식으로 제품 데이터 입력 후 저장
 - 데이터를 삭제하려면 삭제할 행의 앞 부분에 마우스 대고 오른쪽 메뉴 Delete 사용해 삭제



productName	cost	makeDate	company	amount
컴퓨터	10	2021-01-01	삼성	17
세탁기	20	2022-09-01	LG	3
냉장고	5	2023-02-01	대우	22
NULL	NULL	NULL	NULL	NULL

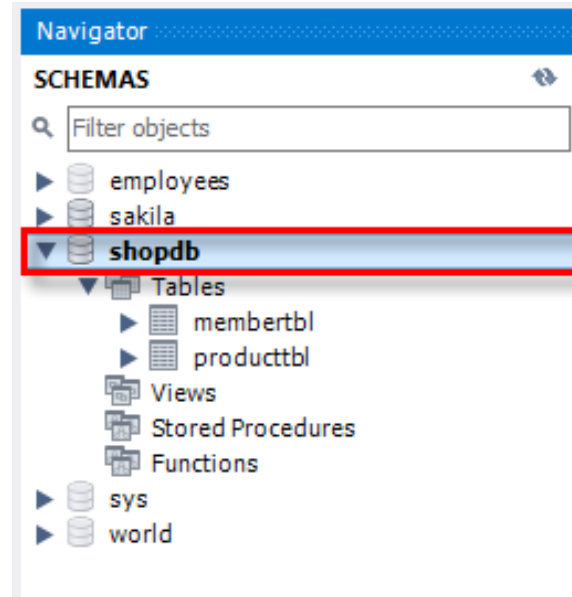


memberID	memberName	memberAddress
Dang	당탕이	경기 부천시 중동
Han	한주연	인천 남구 주안동
Jee	지운이	서울 은평구 증산동
Sang	상길이	경기 성남시 분당구
NaoTa	나오타	주소불명
NULL	NULL	

SECTION 02 MySQL을 이용한 데이터베이스 구축 절차

데이터 활용

- 주로 SELECT 문 사용해 데이터 활용
 - 6, 7장에서 자세히 학습
- 사용할 데이터 베이스 선택
 - SCHEMA에서 사용할 DB를 더블 클릭
 - 진하게 색상이 변하면서 선택 됨
- SQL 실행법
 - 툴바의 <Execute the selected portion~~> 아이콘 클릭
 - **Ctrl** + **Shift** + **Enter**
 - Workbench 메뉴의 [Query] >> [Execute(All or Selection)]
- SQL은 대소문자 구별 없음
 - 읽기 편하게 예약어는 대문자 (쿼리 창에서 파란색으로 표시)



SECTION 02 MySQL을 이용한 데이터베이스 구축 절차

데이터 활용

- SELECT 열 이름 FROM 테이블 이름 [WHERE 조건]
 - 모든 데이터 출력하기 (열 이름 대신 ' * ')
 - 열을 선택해 데이터 출력하기 (열 이름 나열)
 - 특정 데이터를 만족하는 데이터 출력하기 (WHERE절에 조건 입력)
- 새로운 테이블 생성
 - 테이블 이름에 space 가 들어간 경우의 처리(백틱[backtick]키 활용)
 - Navigator 창에서 "Refresh All" 의 중요성
 - 새로 테이블을 만든 뒤 개체가 보이지 않을 경우 필수로 실행할 것
- 테이블 삭제
 - DROP TABLE 테이블 이름

SECTION 03 테이블 외의 데이터베이스 개체의 활용

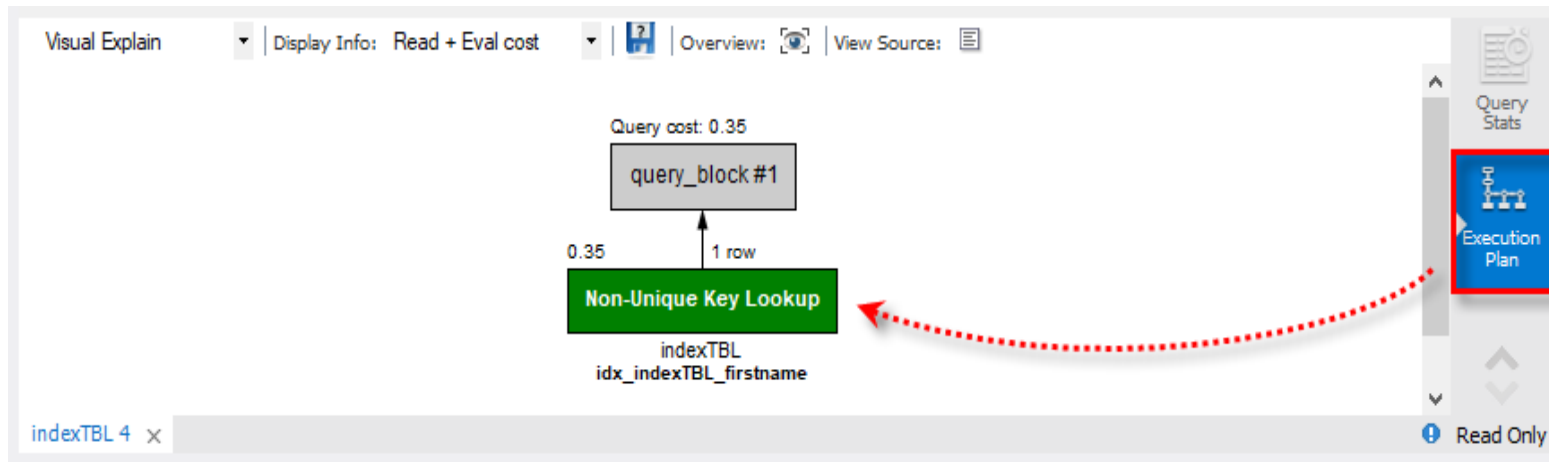
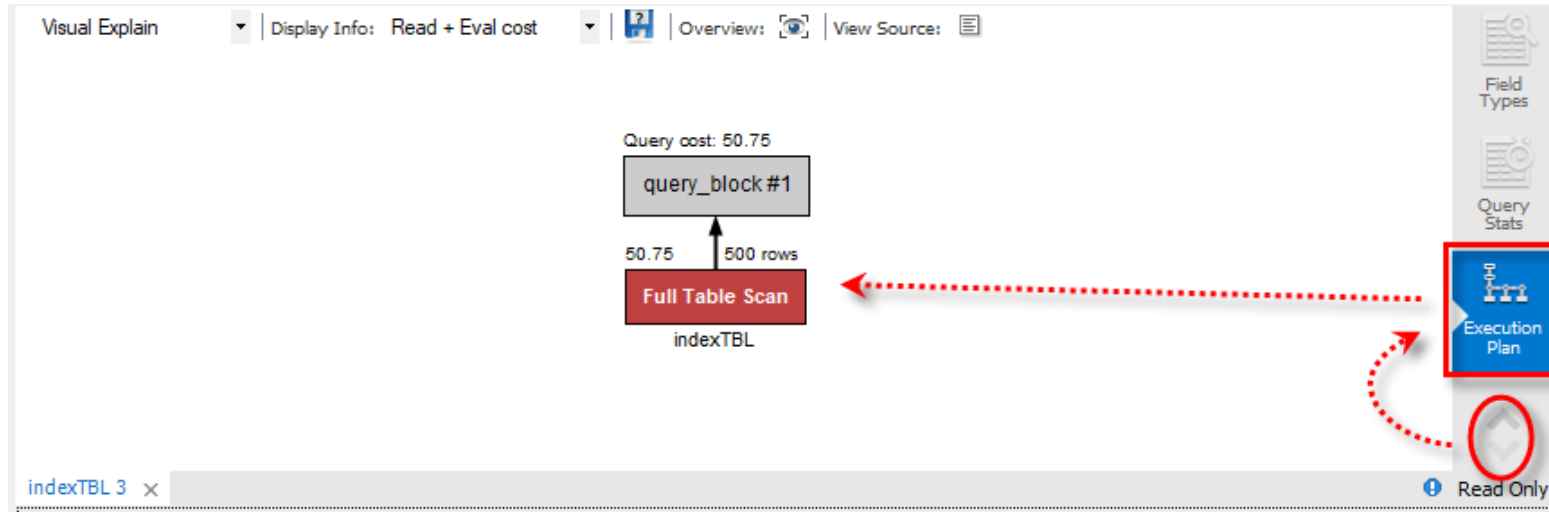
인덱스 (Index)

- 9장에서 자세히 학습 예정
- 데이터베이스 '튜닝'의 개념
 - 데이터베이스 성능 향상
 - 쿼리에 응답하는 시간 단축시키는 것
- 책 뒤에 붙어 있는 '찾아보기'(또는 색인)와 같은 개념
- 데이터의 양이 많을수록 효과적으로 작용
 - 응답속도가 현저히 차이 나는 결과
- 테이블의 열 단위에 생성

SECTION 03 테이블 외의 데이터베이스 개체의 활용

인덱스 (Index)

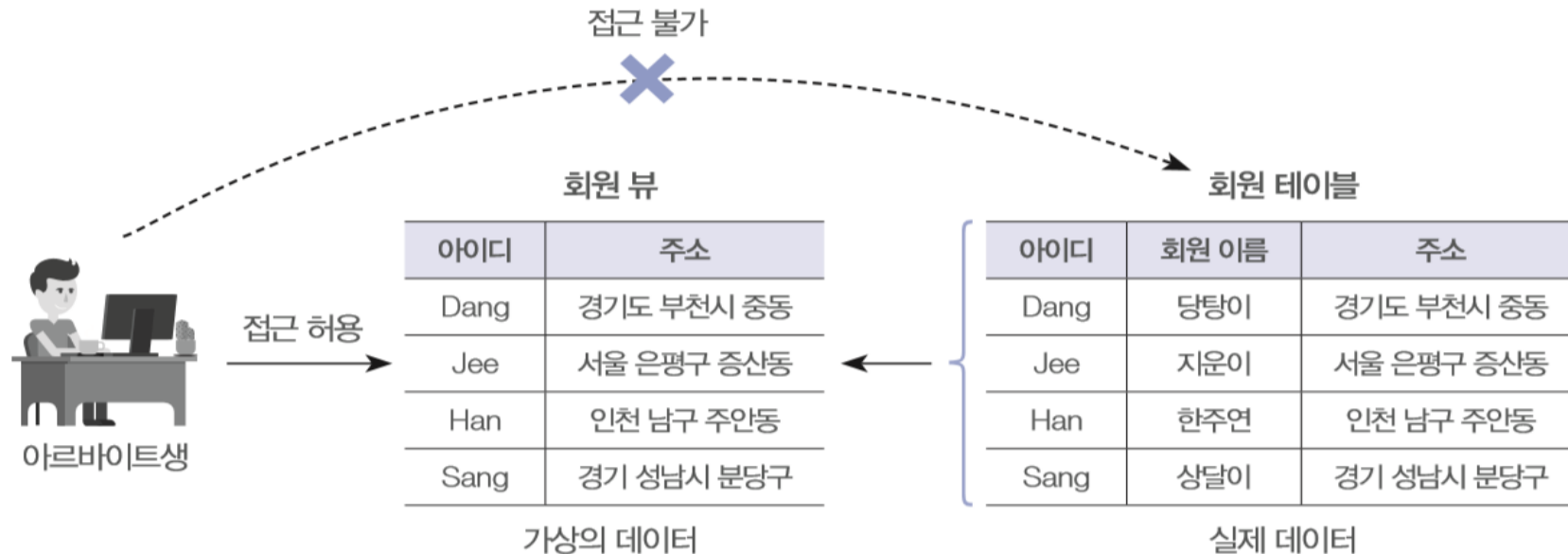
- 인덱스 사용 전/후의 실행 계획 (Execution Plan) 비교



SECTION 03 테이블 외의 데이터베이스 개체의 활용

뷰 (View)

- 가상의 테이블
- 실제 행 데이터를 가지고 있지 않음
 - 그 실체는 없는 것이며, 진짜 테이블에 링크Link된 개념
 - 뷰를 SELECT
 - 진짜 테이블의 데이터를 조회하는 것과 동일한 결과



[그림 3-33] 뷰의 사용 예

SECTION 03 테이블 외의 데이터베이스 개체의 활용

스토어드 프로시저 (Stored Procedure)

- 10장에서 학습 예정
- MySQL에서 제공하는 프로그래밍 기능
- SQL문을 하나로 묶어 편리하게 사용하는 기능
- 다른 프로그래밍 언어와 같은 기능을 담당할 수도 있음
 - 실무에서는 SQL문(주로 SELECT)을 매번 하나하나 수행 X
 - 스토어드 프로시저로 만들어 놓은 후 스토어드 프로시저 호출

SECTION 03 테이블 외의 데이터베이스 개체의 활용

트리거 (Trigger)

- 테이블에 부착되어 테이블에 INSERT나 UPDATE 또는 DELETE 작업이 발생되면 실행되는 코드
- 상세한 내용은 10장에서 학습 예정
- ex) 탈퇴회원 관리
 - 회원 테이블에서 빼서 탈퇴한 회원 관리 테이블로 옮김
 - **회원 정보 + 탈퇴한 날짜** 를 관리하는 새 테이블의 필요성

SECTION 04 데이터베이스 백업 및 관리

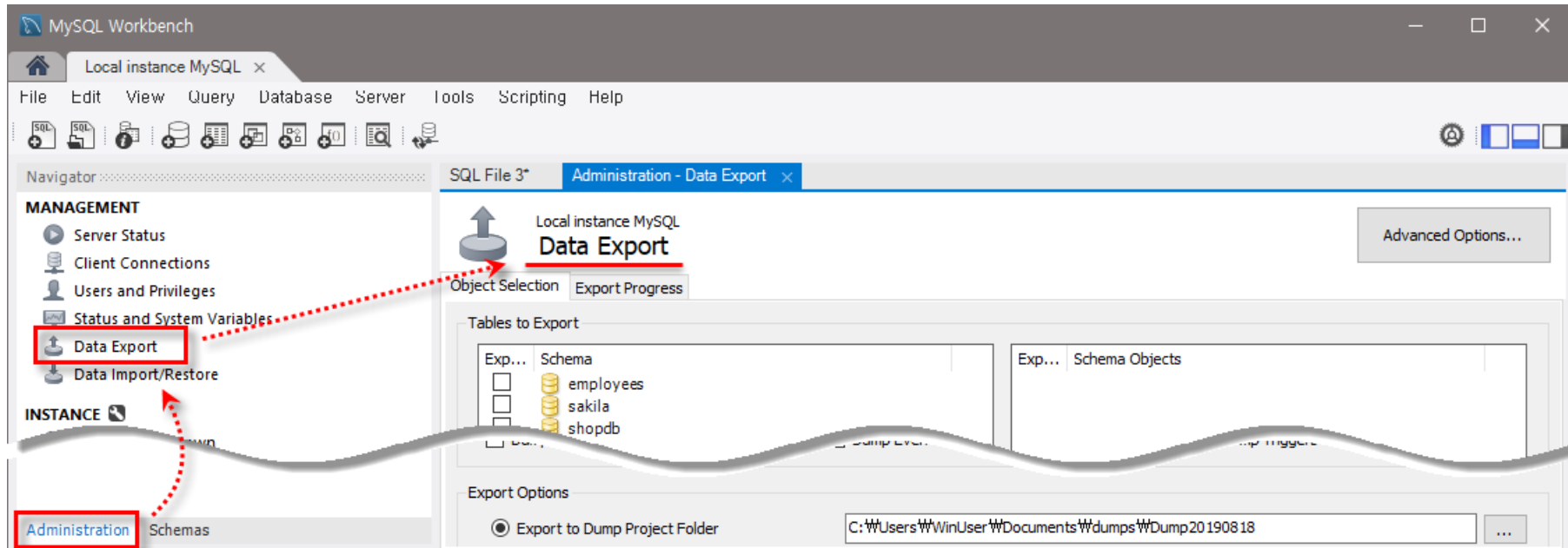
백업과 복원

- 백업
 - 현재의 데이터베이스를 다른 매체에 보관하는 작업
- 복원
 - 데이터베이스에 문제 발생 시 다른 매체에 백업된 데이터를 이용해 원상태로 돌려놓는 작업
- 백업과 복원은 DBA(DataBase Administrator: 데이터베이스 관리자)가 해야 할 가장 중요한 일

SECTION 04 데이터베이스 백업 및 관리

데이터베이스 백업

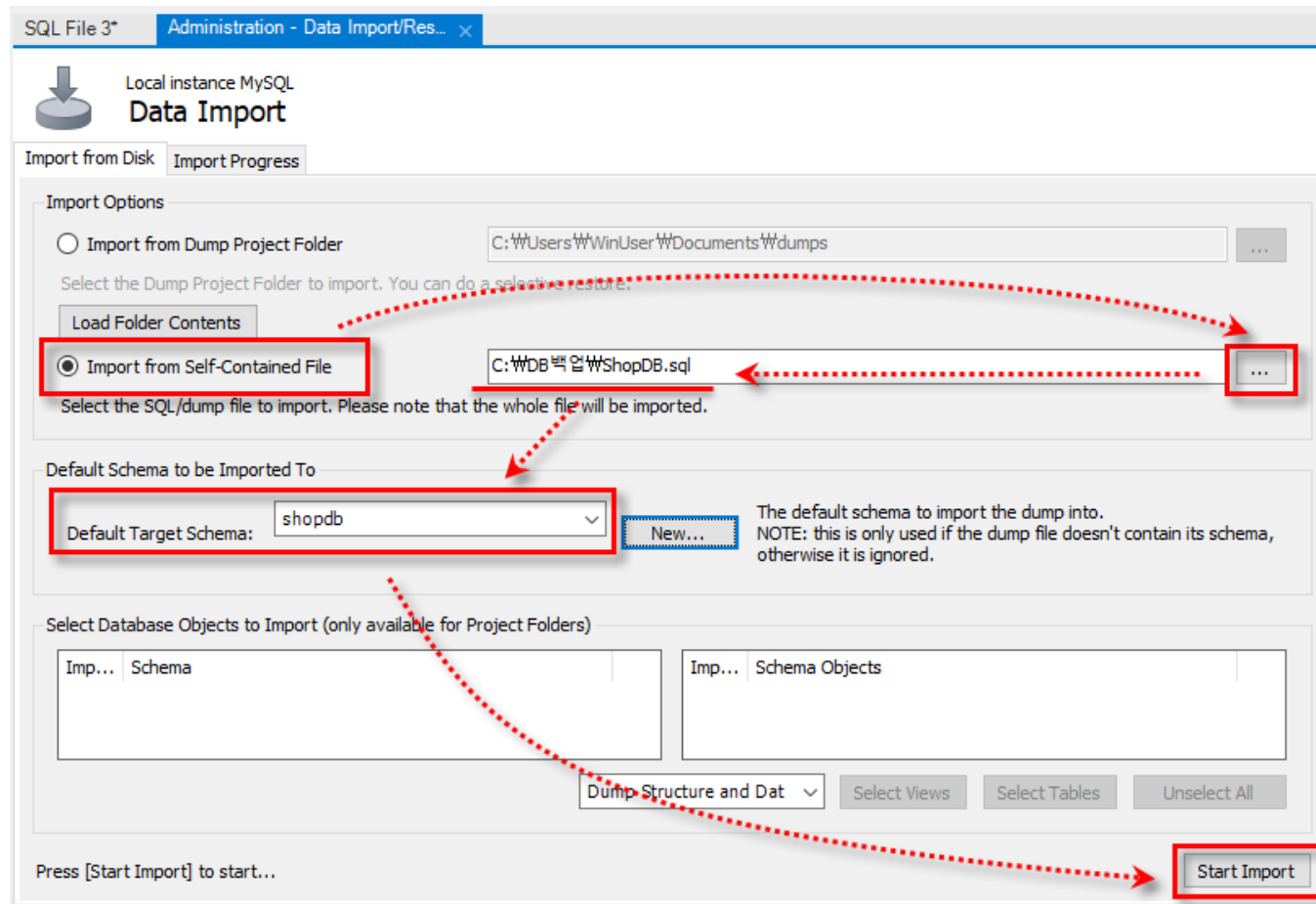
- 백업용 폴더 작성
 - 실제로는 다른 디스크에 이루어져야 의미 있음
- DB 백업
 - DB내의 모든 트리거, 스토어드 프로시저까지 백업
 - 백업 폴더에 백업파일 저장



SECTION 04 데이터베이스 백업 및 관리

데이터베이스 복구

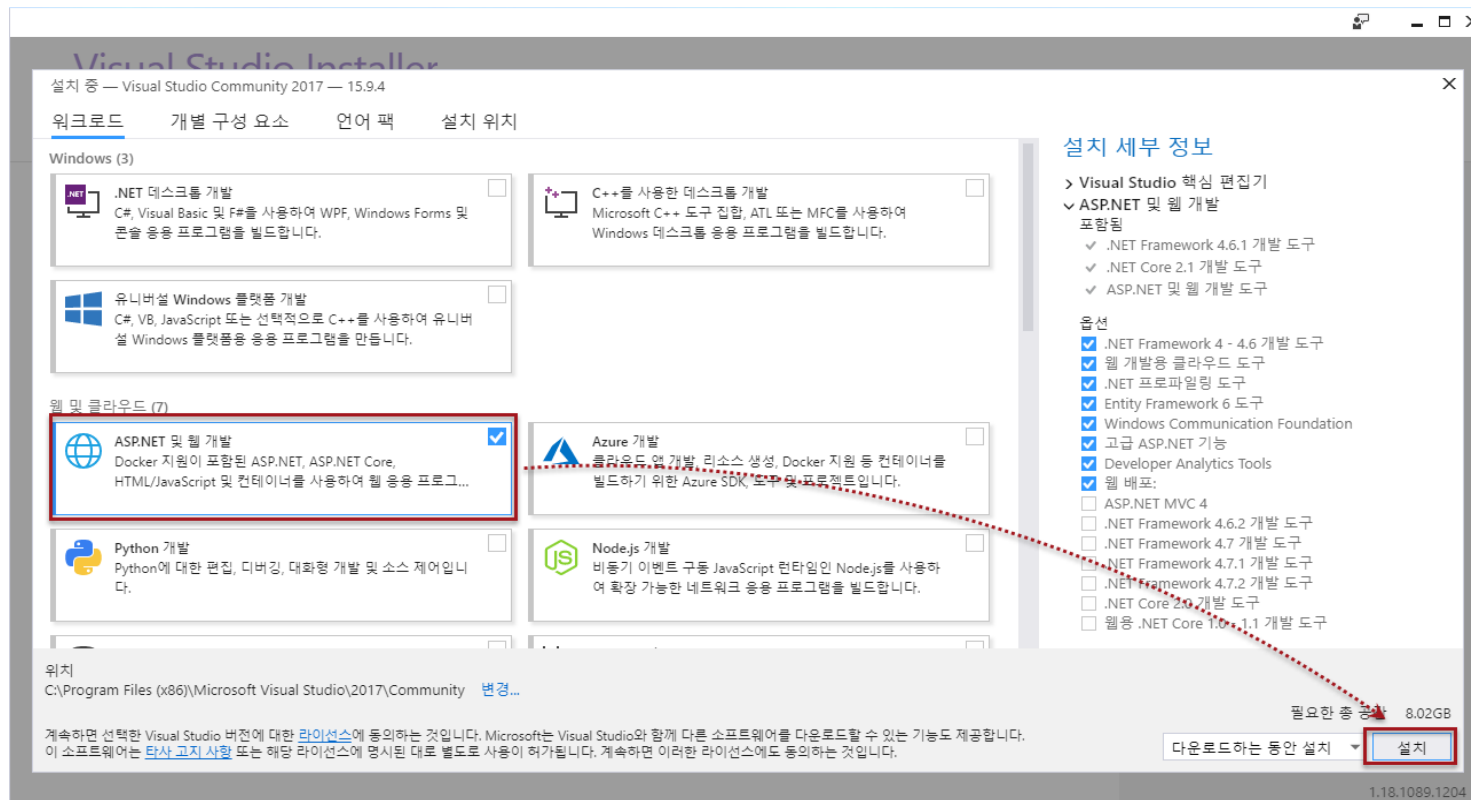
- DB 삭제 같은 큰 사고를 인위로 발생시켜 실습
- 복원 후 데이터가 온전한지 check 하는 것이 중요함



SECTION 05 MySQL과 응용프로그램의 연결

MySQL에서 구축한 쇼핑몰 데이터베이스를 웹에서 서비스

- 개발 툴로 사용할 Microsoft Visual Studio Community 2017 설치 파일 다운로드
 - <http://cafe.naver.com/thisisMySQL> 또는 <http://download.hanbit.co.kr/mysql/8.0/>
 - Visual Studio Community 2017(vs_community_2017.exe, 1.22MB) 다운로드 후 설치
 - <ASP.NET 및 웹 개발> 체크하고 <설치> 클릭



SECTION 05 MySQL과 응용프로그램의 연결

MySQL과 응용프로그램 연결

- Connector/ODBC 설치 2017 설치 파일 다운로드
 - <http://dev.mysql.com/downloads/connector/odbc/>
 - <http://cafe.naver.com/thisiMySQL>
 - 반드시 32bit용 다운로드 (64bit용은 작동에 문제 있음)
 - mysql-connector-odbc-8.0.17-win32.msi 실행해 설치
- ODBC 연결 설정
 - Windows의 [시작] >> [W] >> [Windows 관리도구] >> [ODBC Data Sources(32-bit)] 실행
 - [ODBC 데이터 원본 관리자(32비트)] [시스템 DSN] 탭 클릭 <추가> 버튼 클릭
 - [MySQL ODBC 8.0 Unicode Driver] 선택하고 <마침> 클릭

SECTION 05 MySQL과 응용프로그램의 연결

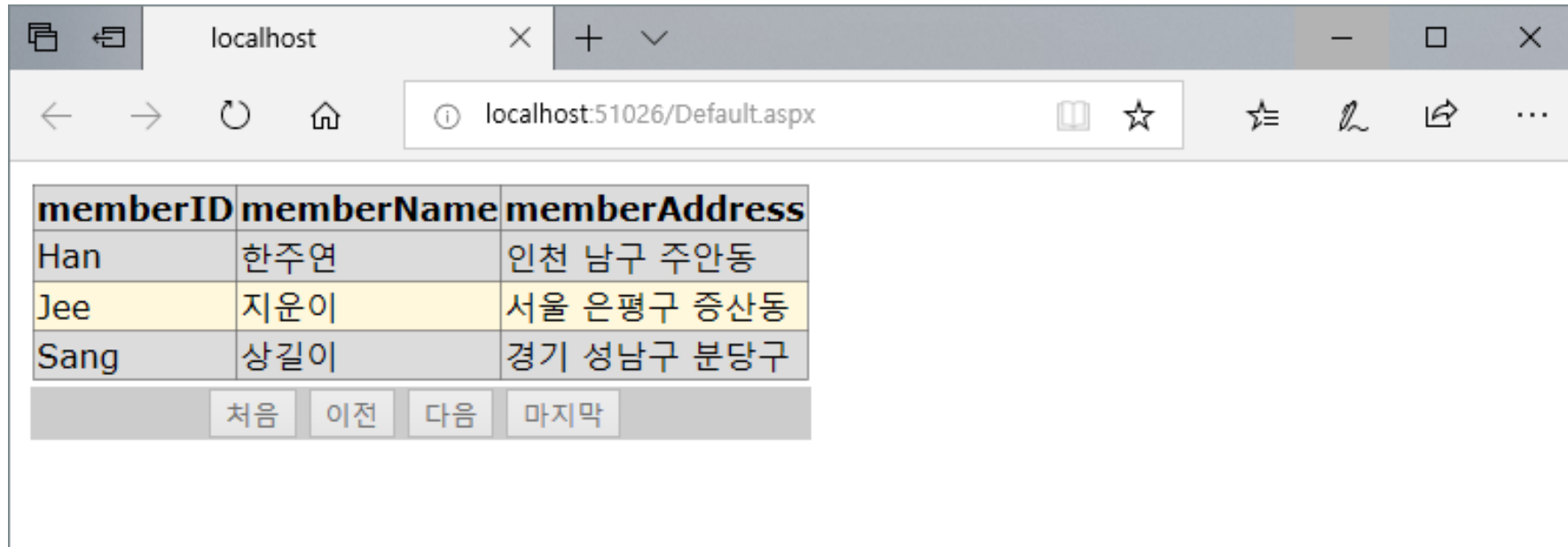
ASP.NET 웹 응용프로그램 작성

- Visual Studio 2017 실행
- GUI 모드에서 웹 사이트 작성
 - [파일] >> [새로 만들기] >> [프로젝트] 클릭, [Visual C#] >> [웹]>>[이전 버전]
<ASP.NET 빈 웹 사이트> 선택
 - 오른쪽 솔루션 탐색기에서 지구 모양 아이콘
 - 'WebSite1(1)'에서 마우스 오른쪽 버튼 클릭, [추가]>>[Web Form]
 - 이름은 'Default' 그대로 두고 <확인> 클릭
 - [디자인]을 클릭해 디자인 모드로 변경
 - [도구 상자] 클릭해 확장
 - [데이터]부분의 [SqlDataSource]를 더블클릭하거나 드래그해서 우측의 빈 디자인 창에 갖다놓기

SECTION 05 MySQL과 응용프로그램의 연결

ASP.NET 웹 응용프로그램 작성

- GUI 모드에서 웹 사이트 작성
 - 데이터 연결
 - 웹에서 보여줄 내용의 Query 입력
 - SQL문 작성
 - 적절한 레이아웃 설정
 - 웹사이트 서비스 테스트



A screenshot of a web browser window. The address bar shows 'localhost:51026/Default.aspx'. The main content area displays a table with three columns: 'memberID', 'memberName', and 'memberAddress'. The table contains three rows of data. The second row is highlighted in yellow. Below the table are four buttons: '처음' (First), '이전' (Previous), '다음' (Next), and '마지막' (Last).

memberID	memberName	memberAddress
Han	한주연	인천 남구 주안동
Jee	지운이	서울 은평구 증산동
Sang	상길이	경기 성남구 분당구

처음 이전 다음 마지막

▶ 이것이 MySQL 이다

Thank You!

