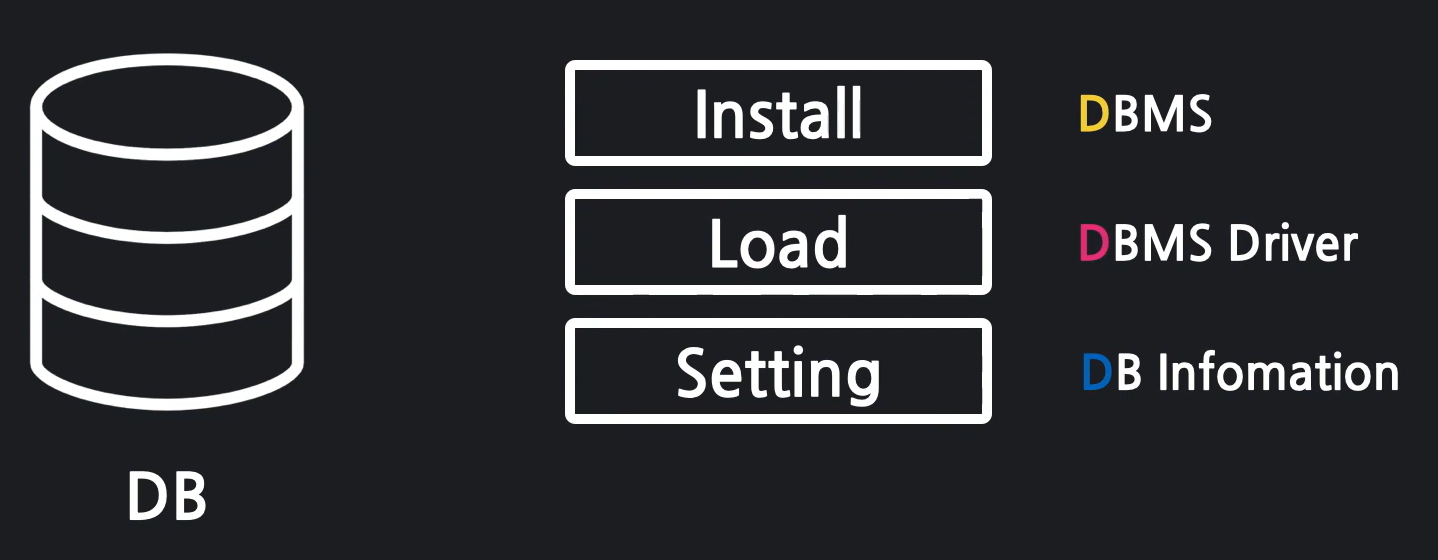
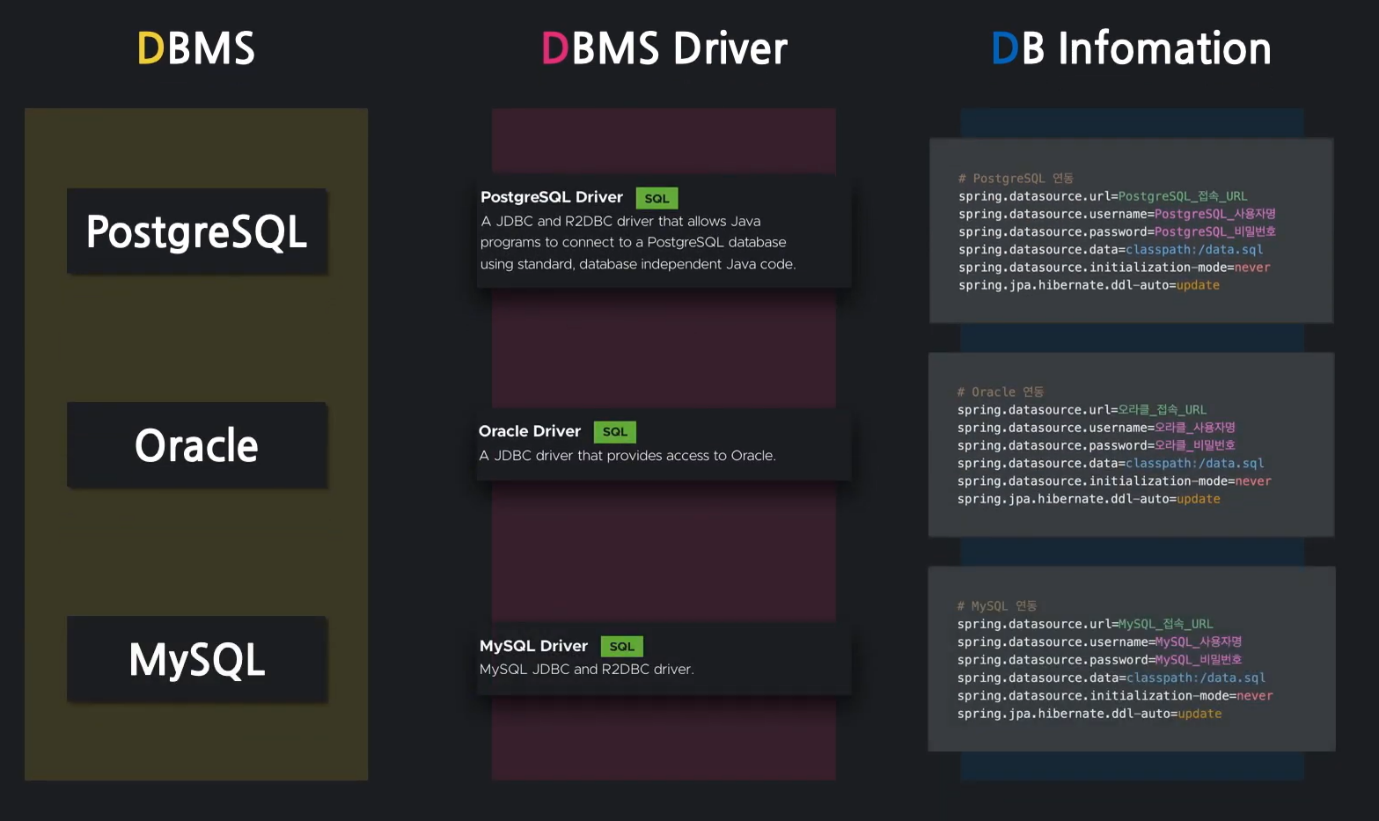
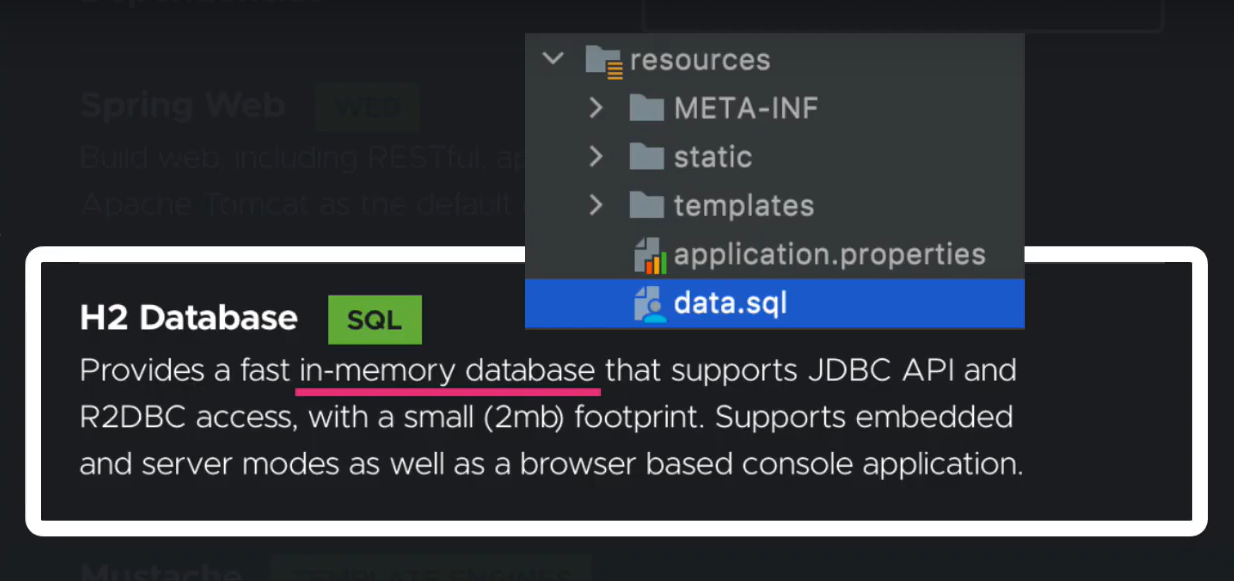
DB 연동하기



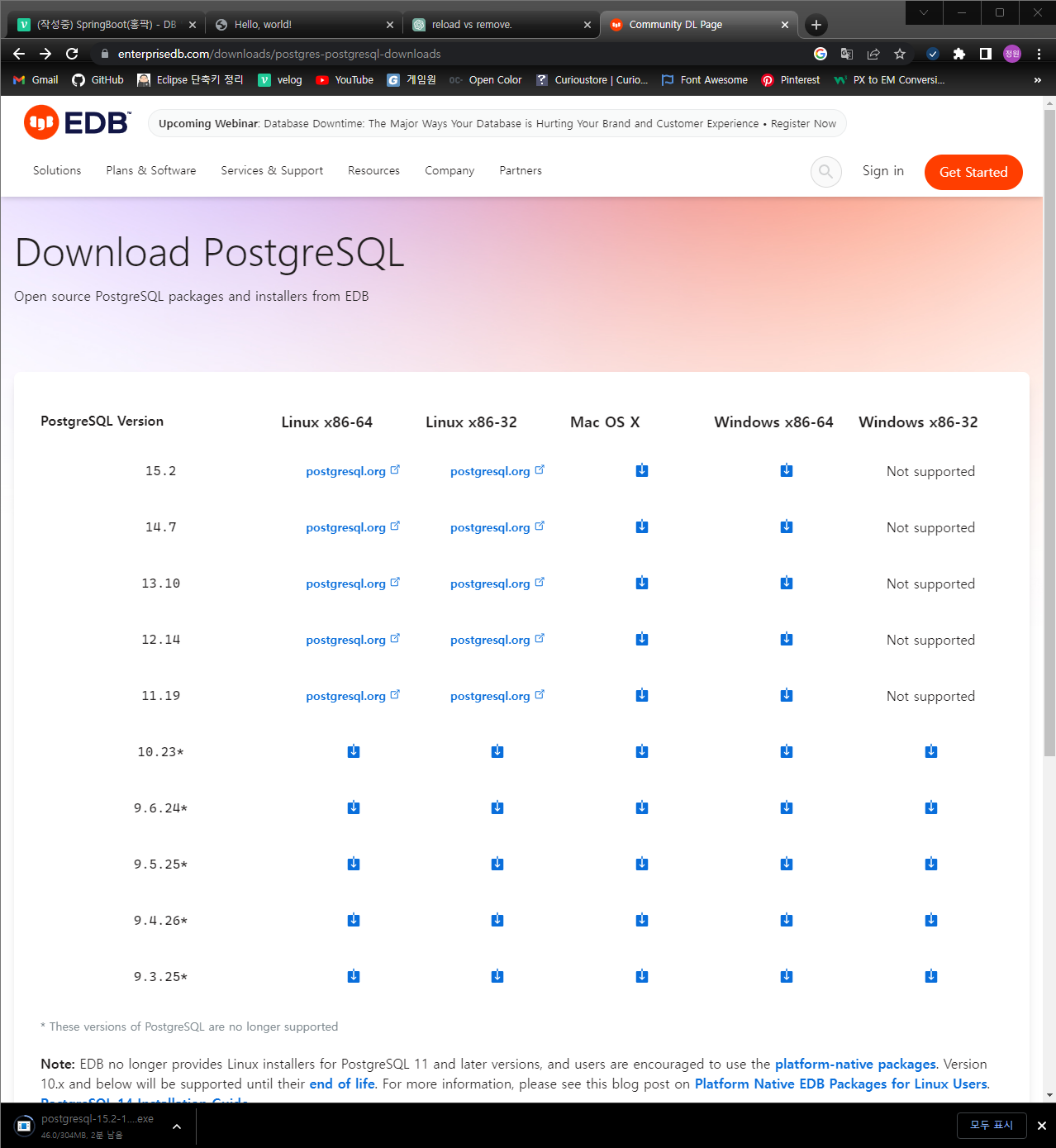
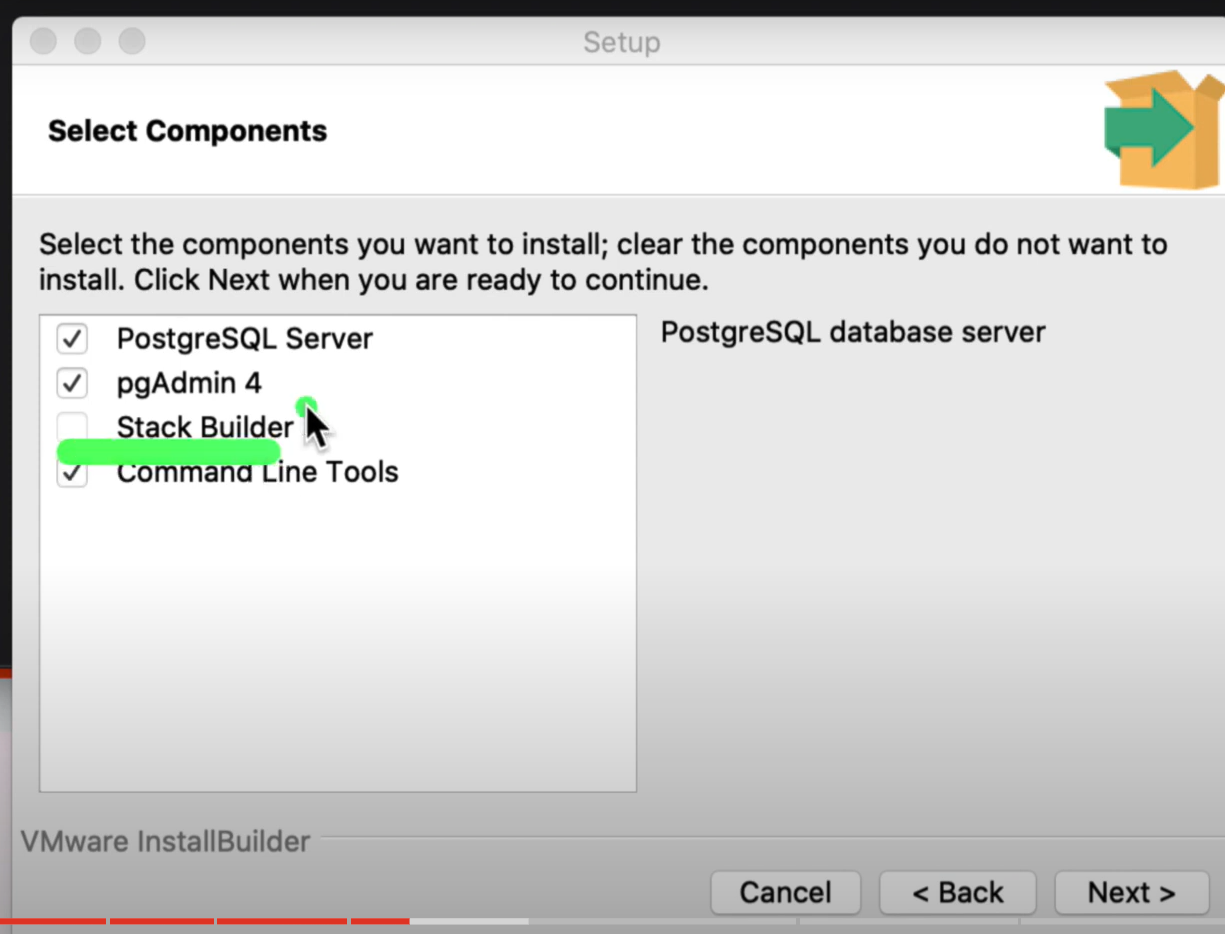


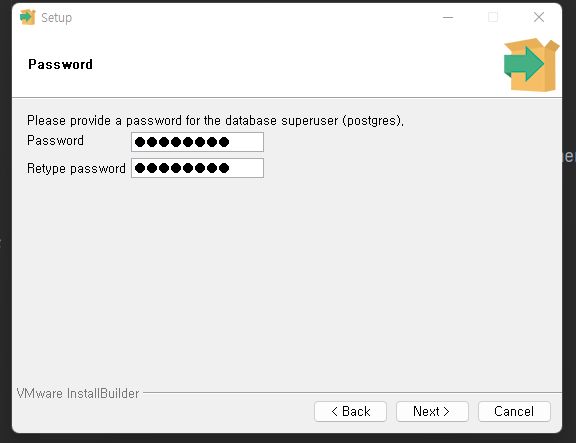
기존에는 H2 Database를 사용하여  
data.sql에 직접 데이터를 추가해서 사용했었다.  


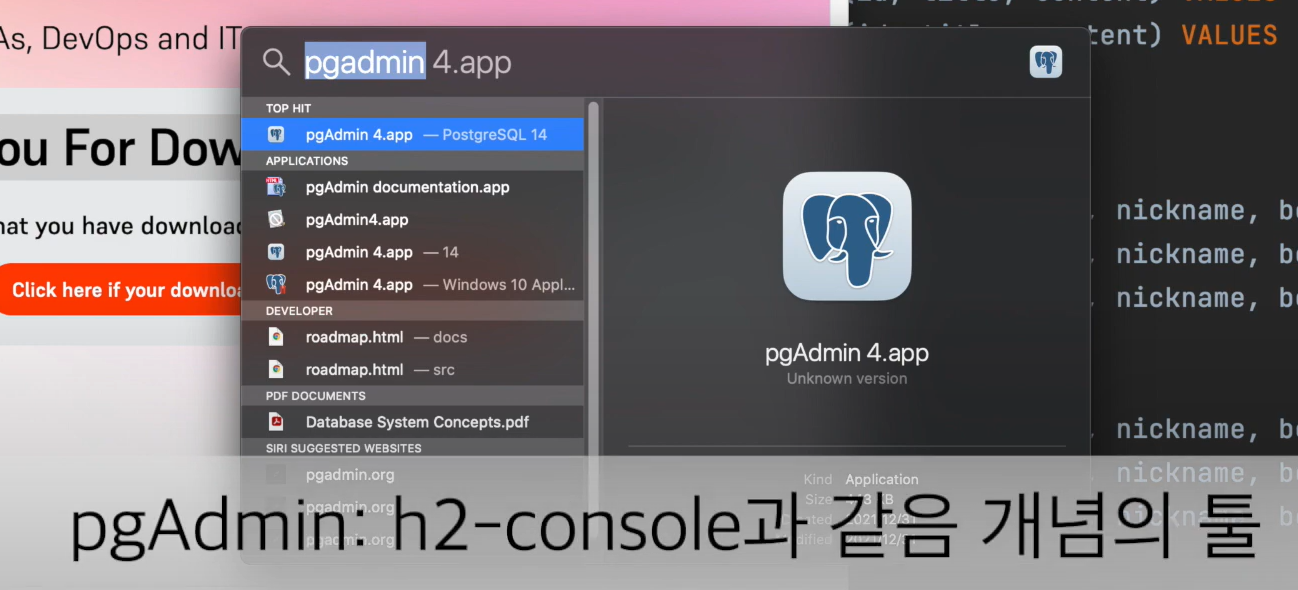
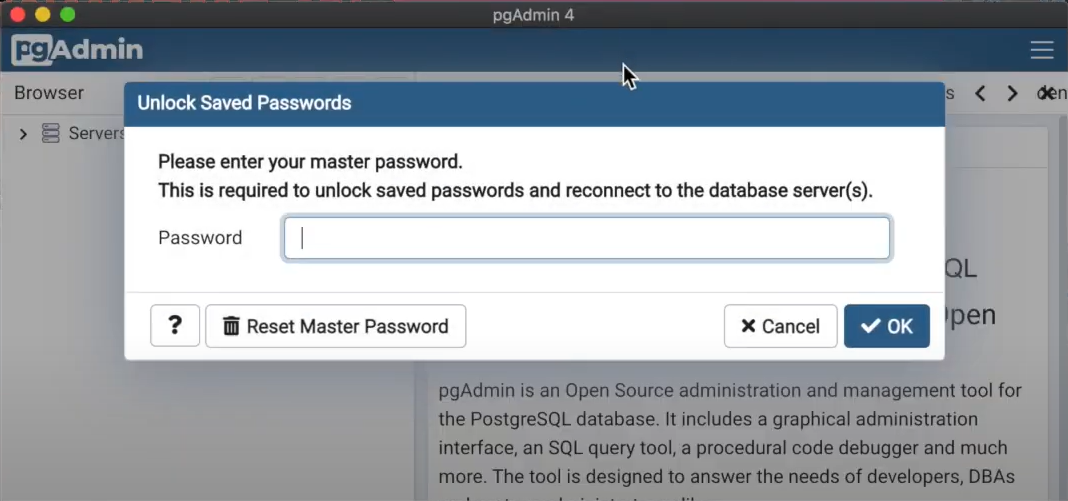
이제는 데이터를 초기화하지 않고  
반영구적으로 저장 및 관리할 수 있도록 DB를 사용해보자.

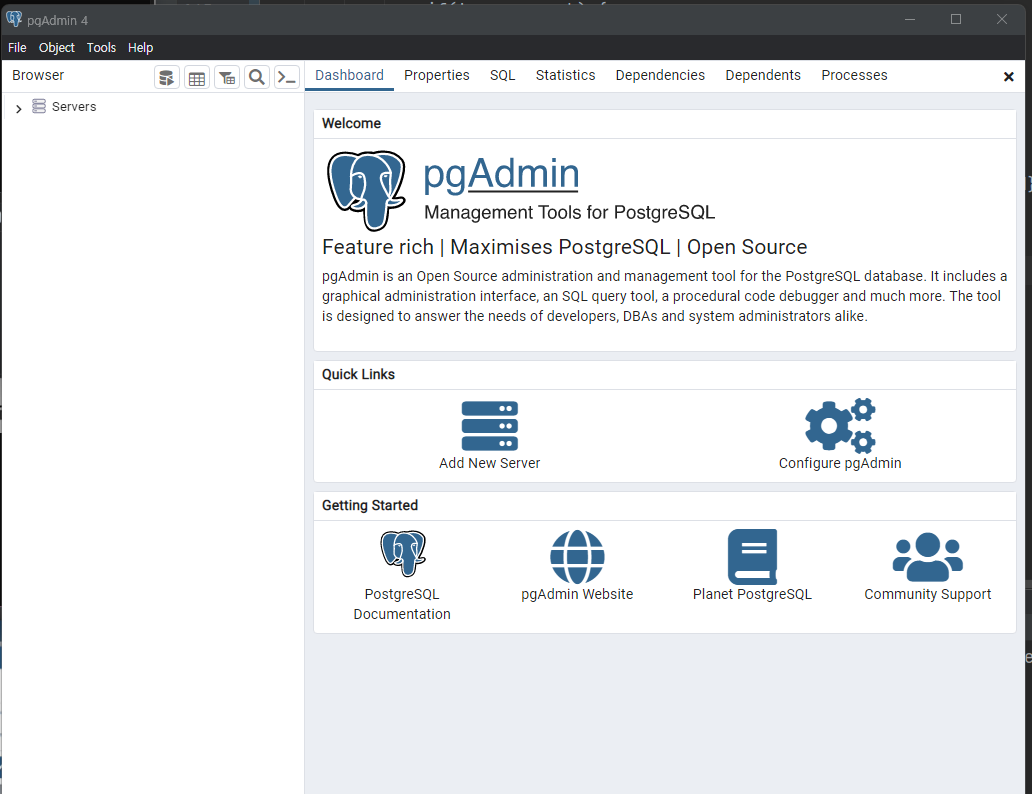
* 사용DB : **PostgreSQL**

**PostgreSQL 설치**

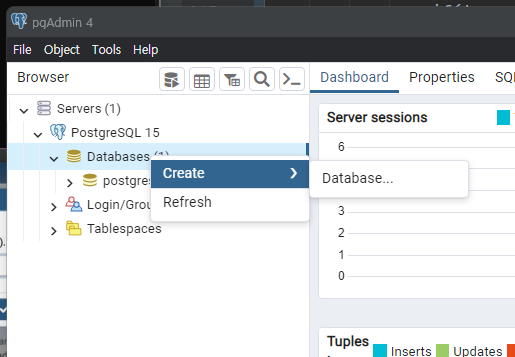
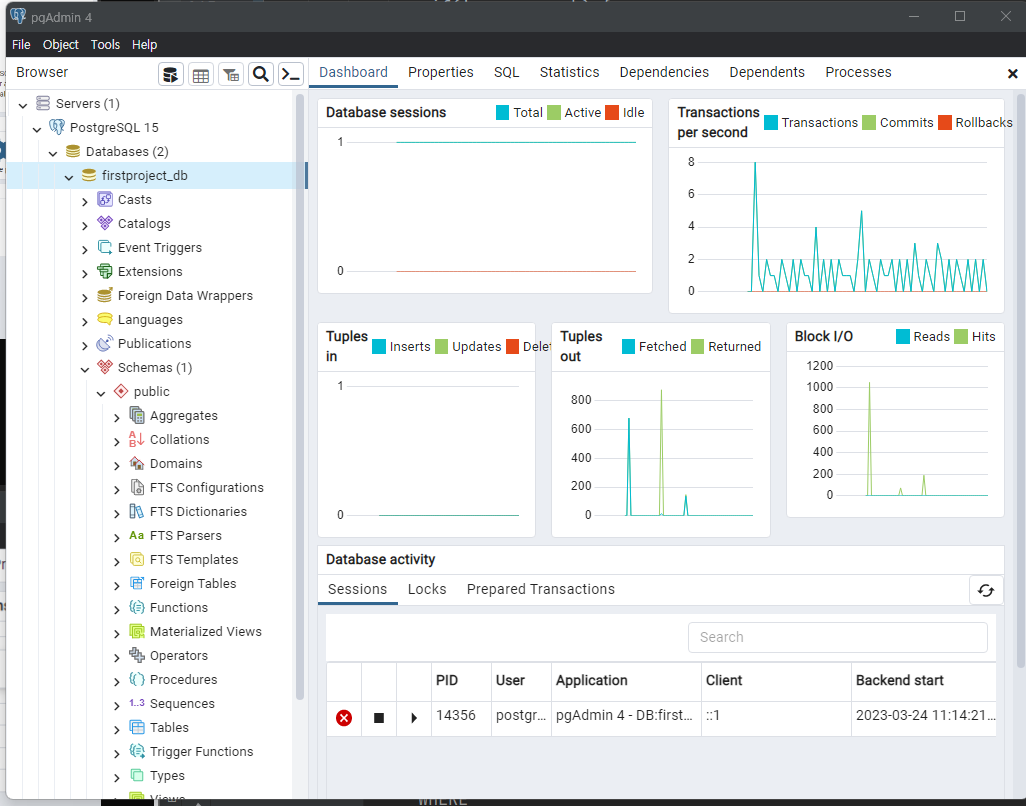
<https://www.postgresql.org/>  
15.2 버전 설치  
  
components 설치에서 Stack Builder는 빼고 설치  


superuser(postgres)의 비밀번호 설정.  
똑같이 postgres로 설정.  
  
👌 설치 완료  
pgAdmin 검색해서 실행.

  
비밀번호 입력 후 실행(postgres)  




**DB 새로 만들기**

**DB 드라이버 추가**

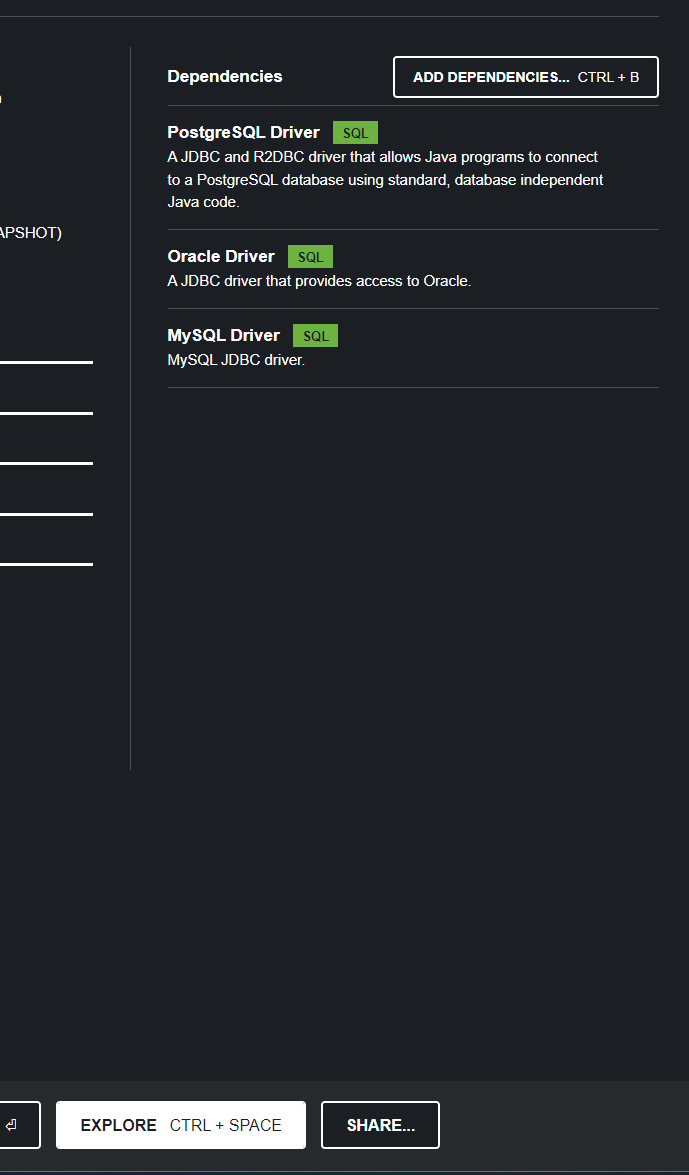
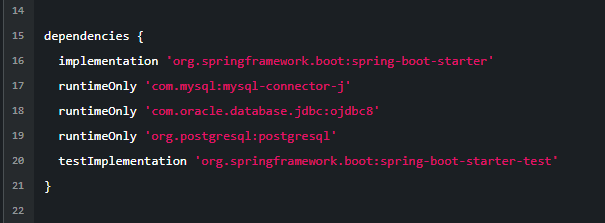
스프링부트에 build.gredle에 라이브러리 추가.

dependencies {

// PostgreSQL 드라이버 추가

runtimeOnly 'org.postgresql:postgresql'

}

postgreSQL말고  
Oracle이나 Mysql을 사용할때 드라이버 추가 코드는  
[Spring Initializr](https://start.spring.io/)에서 Dependencies에 원하는 DB 추가후 밑에 EXPLORE누르면 확인이 가능하다.  
  


**DB 연동 설정**

드라이버를 스프링부트에 연동하기.

**apllication.properties**

DB 설정 추가하기.

# PostgreSQL 연동

# 연동할 DB의 주소와 id,pw 넣기

spring.datasource.url=jdbc:postgresql://localhost:5432/firstproject\_db

spring.datasource.username=postgres

spring.datasource.password=postgres

# 데이터를 초기화할 정보

spring.datasource.data=classpath:/data.sql

# 서버실행 시 초기화를 매번 실행

spring.datasource.initialization-mode=always

# 서버가 켜질때 테이블과 데이터를 생성하고 서버가 꺼지면 삭제된다.

spring.jpa.hibernate.ddl-auto=create-drop

이렇게 DB연동을 하고 데이터를 삭제했다가  
서버를 다시 시작하면

spring.datasource.data=classpath:/data.sql

spring.datasource.initialization-mode=always

위의 설정으로 인해서 data.sql 데이터가 다시 등록된다.

**DB 데이터 유지하기**

이제는 서버 on/off 유무에 상관없이  
데이터를 유지해보자.

**apllication.properties**

spring.datasource.data=classpath:/data.sql

spring.datasource.initialization-mode=always

spring.jpa.hibernate.ddl-auto=create-drop

위의 코드 아래와 같이 변경.

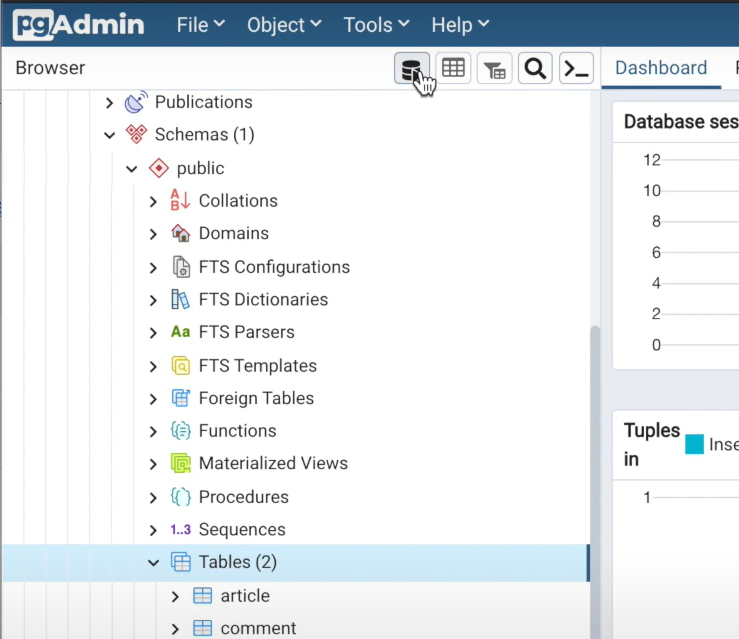
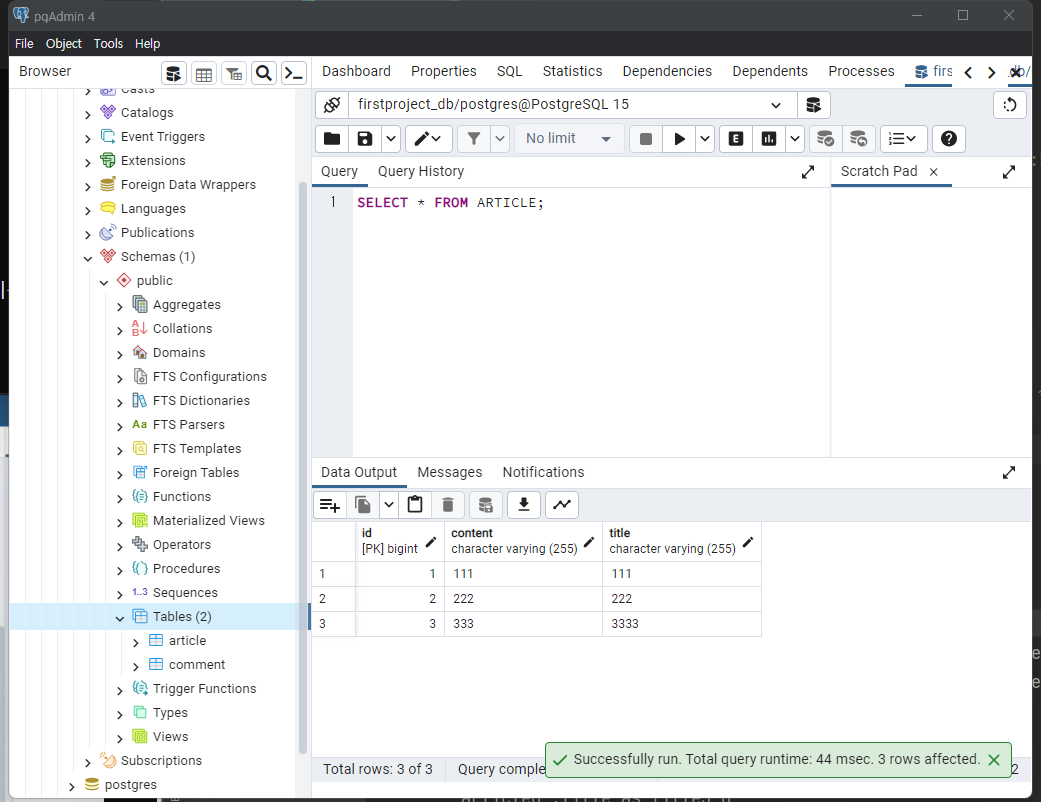
spring.datasource.data=classpath:/data.sql

spring.datasource.initialization-mode=never

spring.jpa.hibernate.ddl-auto=update

이제 데이터가 data.sql에 있는 내용대로  
초기화 되지 않는다.

**DB로 직접 데이터 확인**

pgAdmin에서 쿼리 확인 할때는 Query Tool을 이용해서 확인할 수 있당.  
  


**PostgreSQL과 Oracle**

PostgreSQL과 Oracle은 모두 각자의 장단점이 있으며, 선택은 사용 사례와 개발자 선호도에 따라 달라집니다.  
그러나 일반적으로 PostgreSQL은 소규모 또는 중규모의 프로젝트에 적합하며,  
Oracle은 대규모 엔터프라이즈급 프로젝트에 적합합니다.

다음은 PostgreSQL과 Oracle의 장단점입니다.

**PostgreSQL의 장점:**

* 오픈소스 라이센스를 사용하여 라이센스 비용이 없습니다.
* ACID 준수 및 트랜잭션 처리 등 기본 기능이 강력합니다.
* 고성능 쿼리 처리와 대규모 데이터 집합에 대한 확장성이 뛰어납니다.
* JSON 및 XML과 같은 비정형 데이터 유형에 대한 내장 지원이 있습니다.
* 뛰어난 커뮤니티 지원과 문서화가 있습니다.

**Oracle의 장점:**

* 대규모 엔터프라이즈급 데이터베이스 시스템에 적합합니다.
* 막대한 기능과 옵션을 제공합니다.
* 효율적인 데이터 복제 및 클러스터링을 제공합니다.
* 뛰어난 보안 기능과 강력한 백업 및 복원 기능이 있습니다.
* 고성능 쿼리 처리 및 대규모 데이터 집합에 대한 확장성이 뛰어납니다.

따라서 프로젝트의 크기, 예산, 요구 사항 등에 따라 PostgreSQL과 Oracle 중 어느 것이 더 적합한지를 판단해야 합니다.