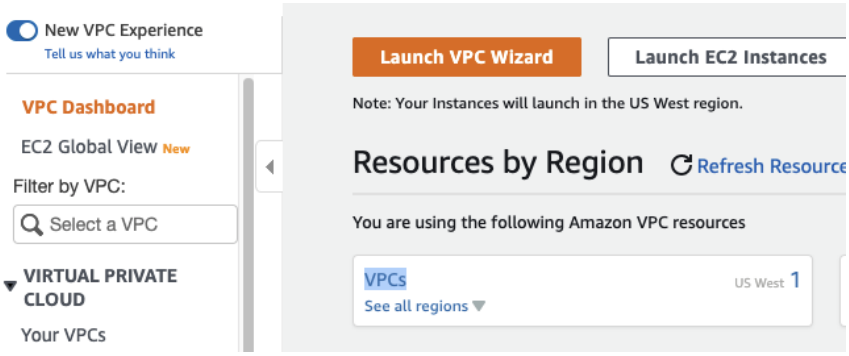
01) PostgreSQL, Dyanmodb 환경구성

[1] 실습 주의 사항

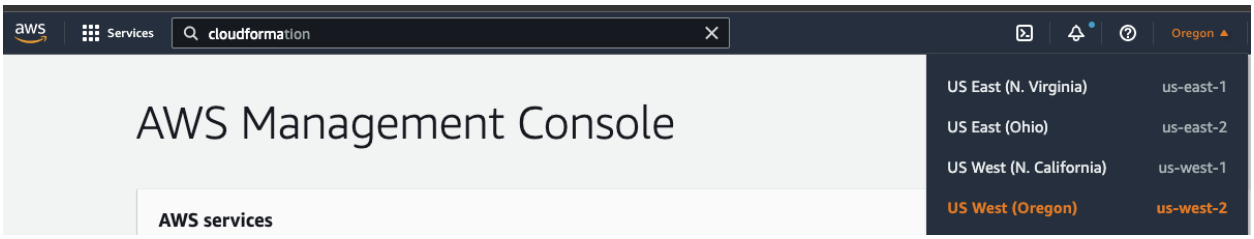
- IAM 사용자 : AWS 모범 사례에 따라 루트 계정 사용을 지양하고, 실습에 맞는 권한을 가진 IAM 사용자를 생성하시기를 바랍니다.
- 실습 region : (us-west-2 인 Oregon) 또는 (us-east-1 인 N.virginia) 을 기준으로 진행합니다
- VPC : 해당 리전의 실습용 VPC 2개 (해당 리전에서 사용하시는 VPC가 3개 이하이면 됩니다.)
    - 확인방법 : AWS management Console 돋보기에서 VPC 입력 —> VPC새화면 에서 보이는 VPCs 값이 3보다 작으면 됩니다.



    - VPC 값이 4 또는 5 인 경우 : 삭제 가능한 VPC 삭제 후 3개 이하로 낮춤

[2] cloudformation 이용 PostgreSQL, Dyanmodb setup

1. AWS management Console에서 우측 상단 region 을 (us-west-2 인 Oregon) 또는 (us-east-1 인 N.virginia) 선택하시고 해당 리전은 다시 지정 필요가 있어 메모해 두시기 바랍니다 (%주의 필요)

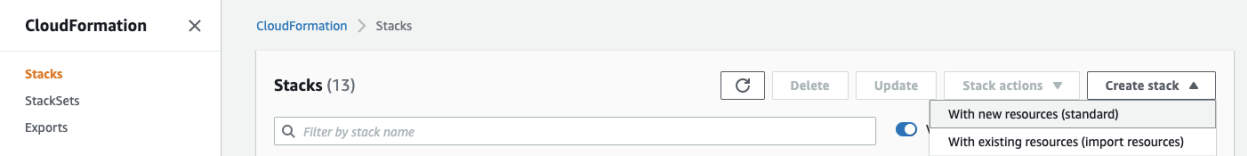


2. 첨부파일 builder202204.zip 파일의 cloudformation 폴더 안에 PurposeBuiltDB.json 파일 확인

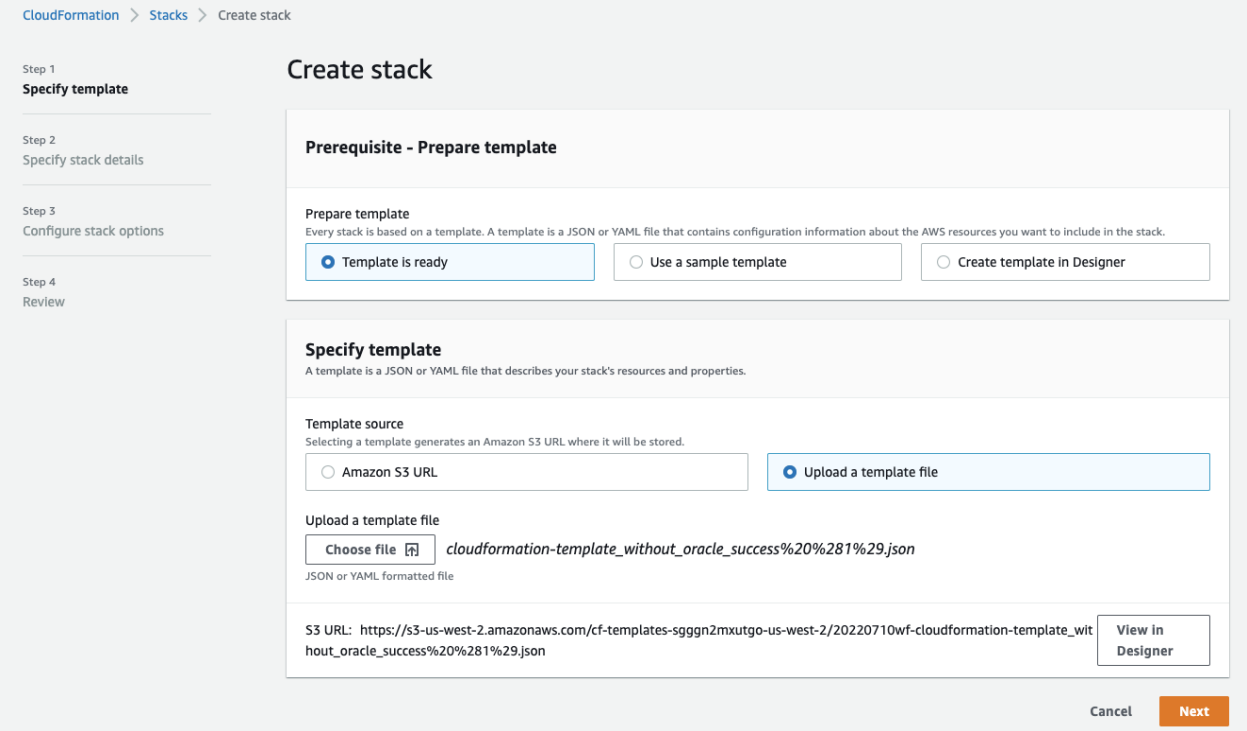
https://s3-us-west-2.amazonaws.com/secure.notion-static.com/fbf4bcf1-4850-4c6f-b0ed-b711043e963f/builder202204.zip

3. AWS management Console 돋보기에서 cloudformation 입력 —> cloudformation 새 화면 —> 메뉴 좌측 상단 stack 선택

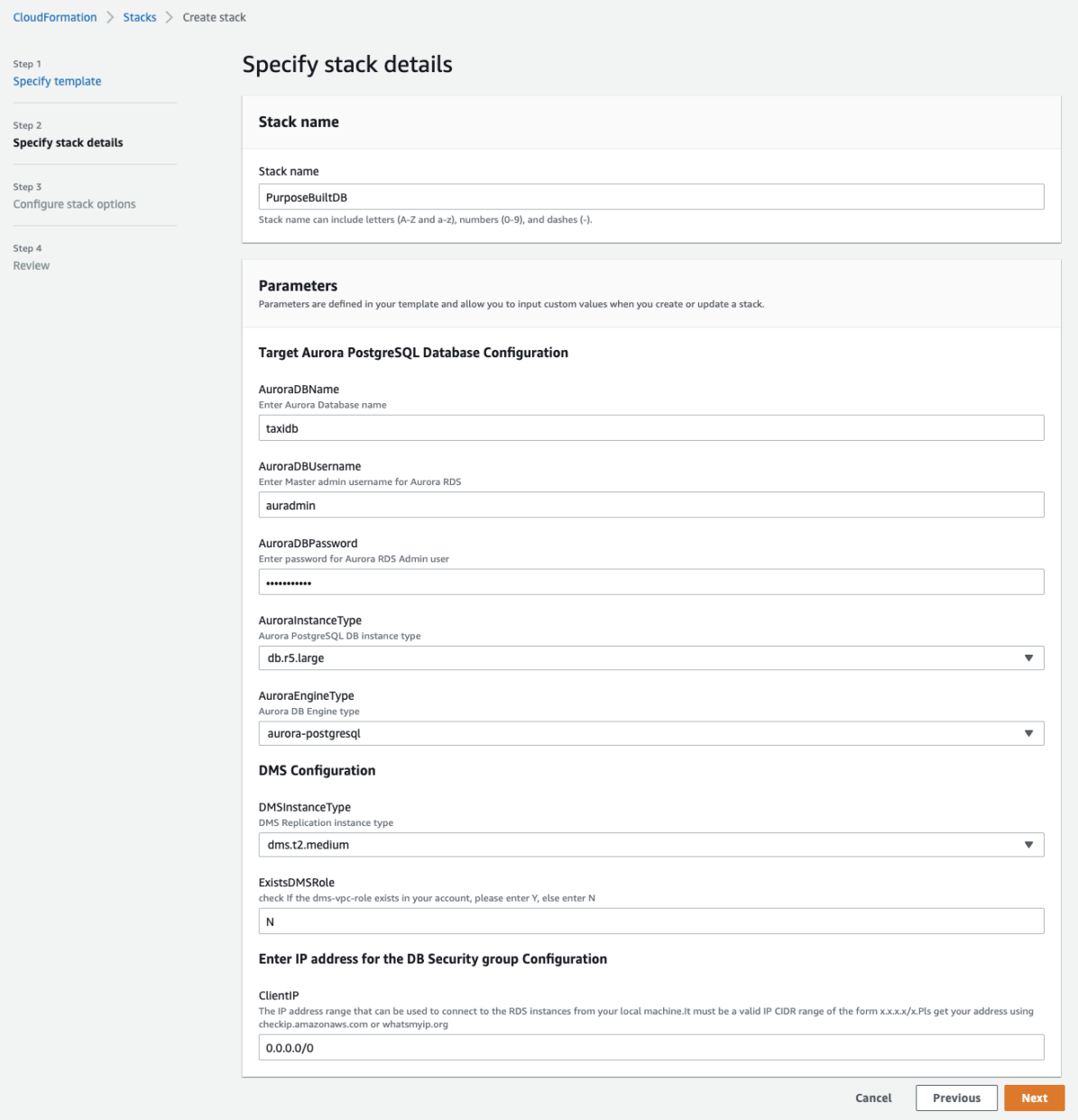
4. cloudformation 메뉴 → 우측 상단 create stack → with new resource (standard) 선택



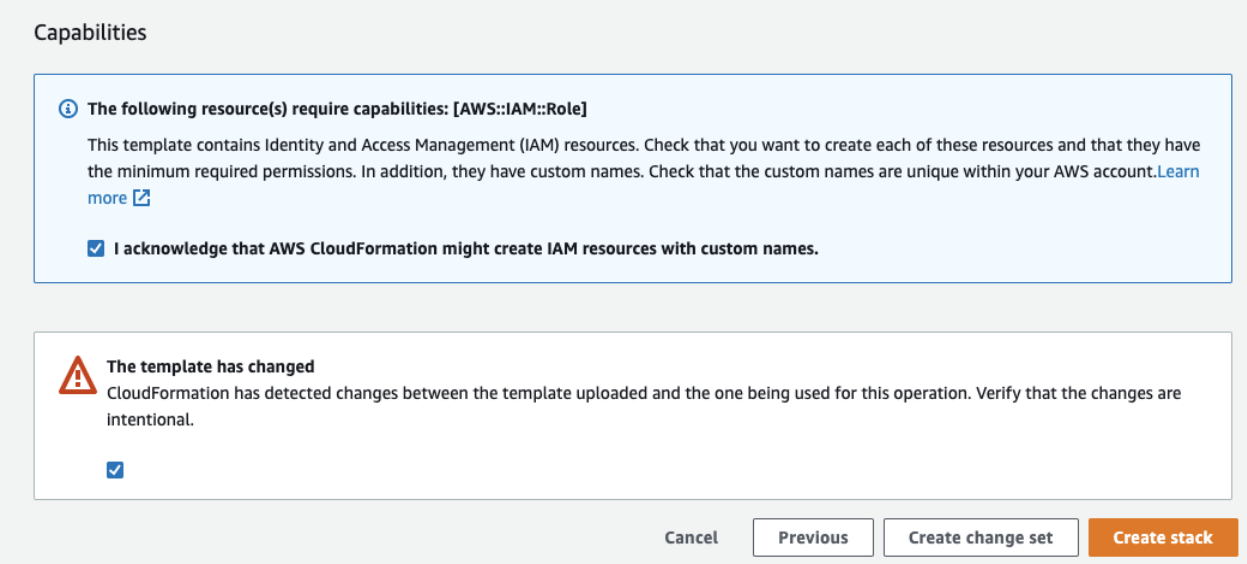
5. 아래 화면기준 cloudformation step1 단계 --> 아래와 같이 설정후 Upload a template file 부분에서 **PurposeBuiltDB.json**을 선택 → next 클릭
   - (다운로드한 Builder202204.zip 파일 압축을 풀고 cloudformation 폴더 안의 PurposeBuiltDB.json 임)



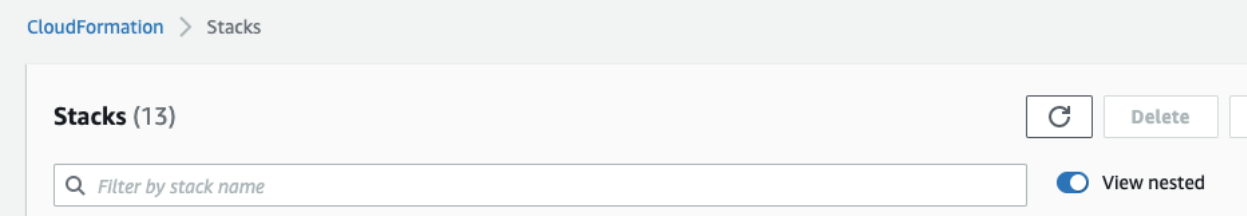
6. 아래 화면기준 cloudformation step2. step3 → stack name: **PurposeBuiltDB** 입력 → next 클릭 → next 클릭
   - 참고 AuroraDBUsername : auradmin 이며 AuroraDBPassword 는 auradmin123 입니다.)



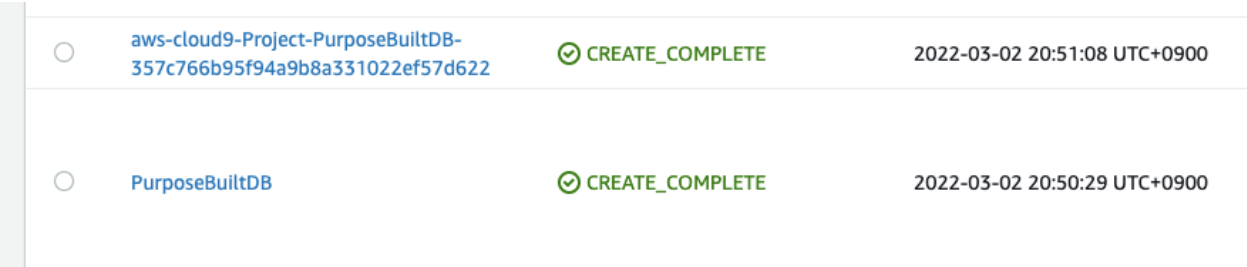
7. 아래 화면기준 step 4 Review→ 아래에 체크 → create stack 클릭



8. cloudformation 설치확인 → PurposeBuiltDB 와 aws-cloud9-Project-PurposeBuiltDB... 모두 CREATE_COMPLETE 임을 확인 (약 10분 소요)

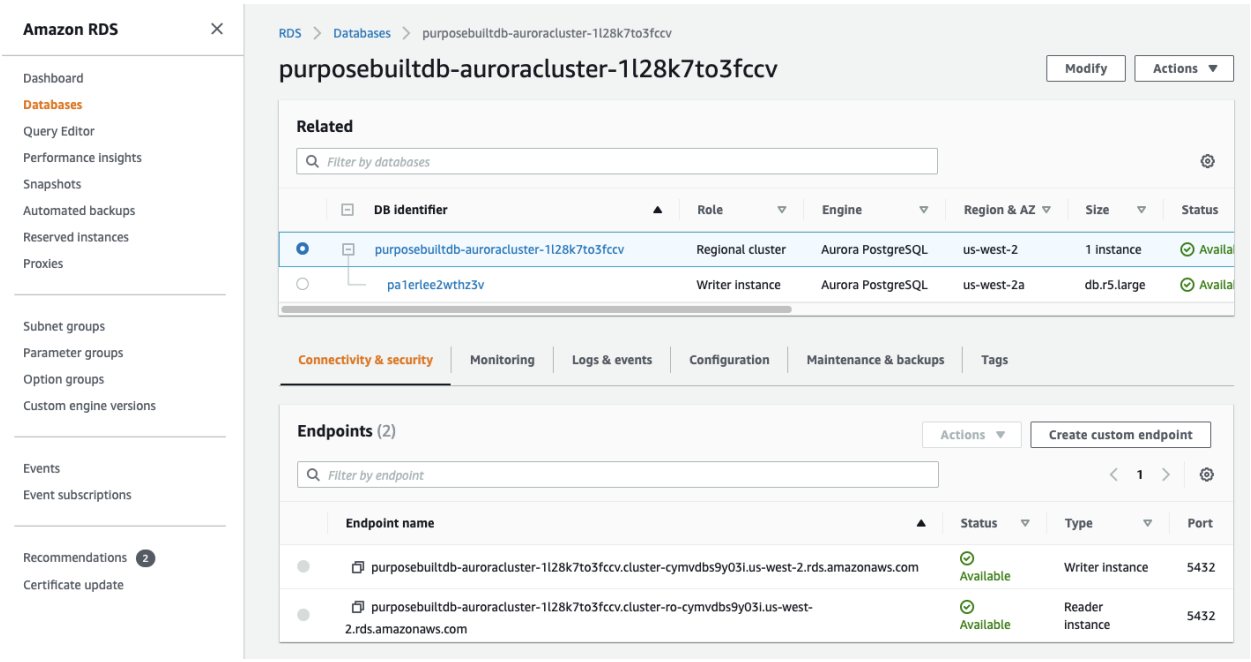


..

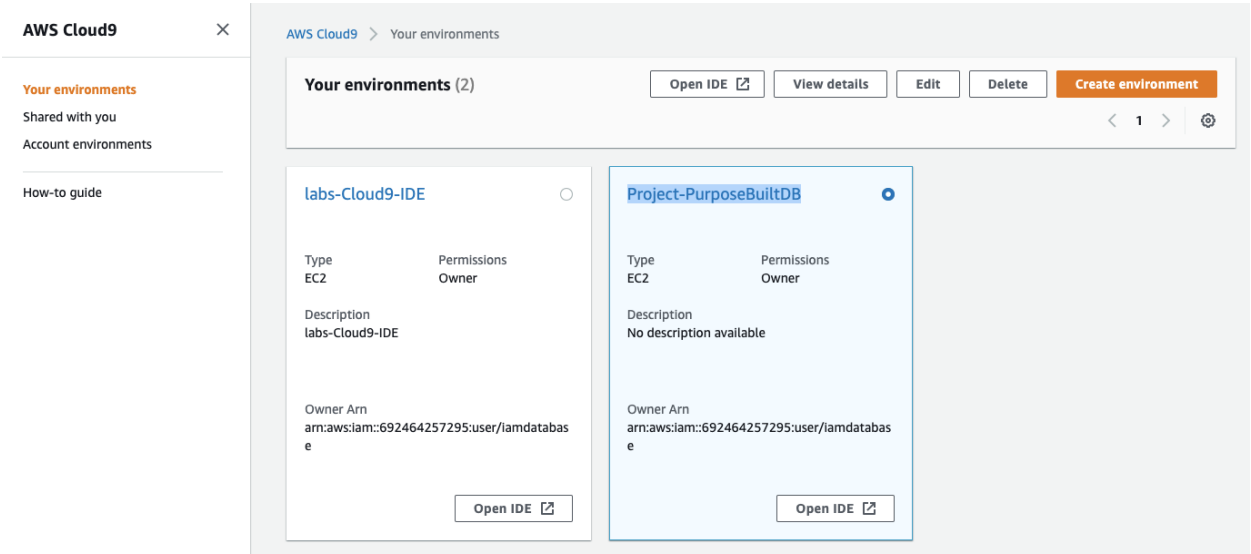


9. AWS Management console 돋보기에서 RDS 입력 → 새화면 Amazon RDS 에서 좌측 Databases 클릭 → 우측 상단에 생성된 DB identifier중 상단 purposebuiltdb-auroracluster... 을 클릭 → 하단의

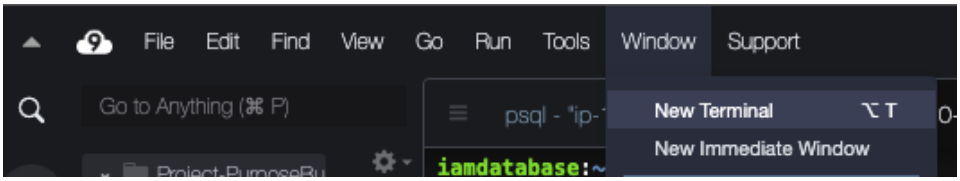
2개 endpoint중 **writer instance endpoint 복사 및 저장** (이후 데이터베이스 접속시 사용)



10. AWS Management console → 돋보기에서 cloud9 입력 및 이동 → 새화면 AWS cloud9 에서 Project-PurposeBuiltDB 선택 → open IDE클릭



11. 새화면 cloud9 에서 상단 window → new Terminal 을 클릭하여 새로운 terminal 을 오픈함



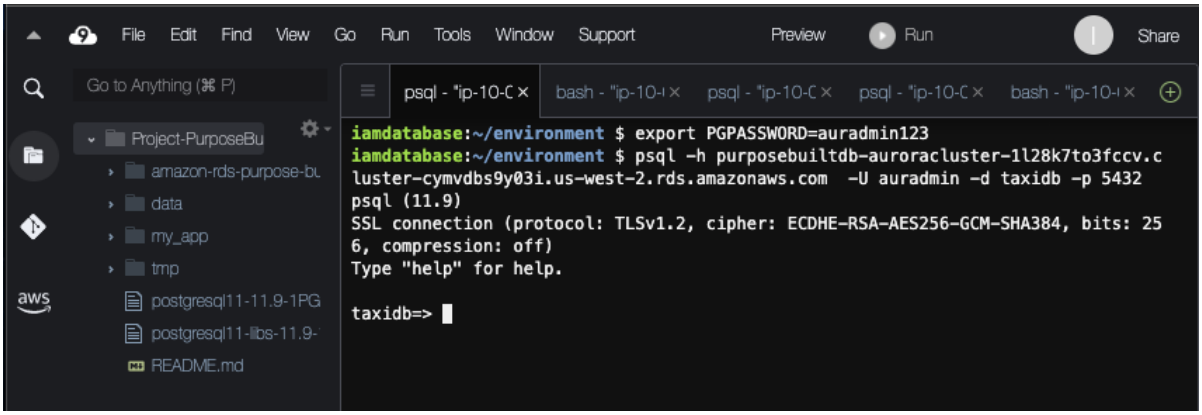
12. cloud9 터미널에서 다음 내용 입력하여 posgreSQL 관련 설치 작업 수행

```
# shell 에서 수행
cd ~/environment
wget https://yum.postgresql.org/11/redhat/rhel-6.9-x86_64/postgresql11-libs-11.9-1PGDG.rhel6.x86_64.rpm
wget https://yum.postgresql.org/11/redhat/rhel-6.9-x86_64/postgresql11-11.9-1PGDG.rhel6.x86_64.rpm

sudo yum clean all
sudo rpm -ivh postgresql11-libs-11.9-1PGDG.rhel6.x86_64.rpm
sudo rpm -ivh postgresql11-11.9-1PGDG.rhel6.x86_64.rpm
```

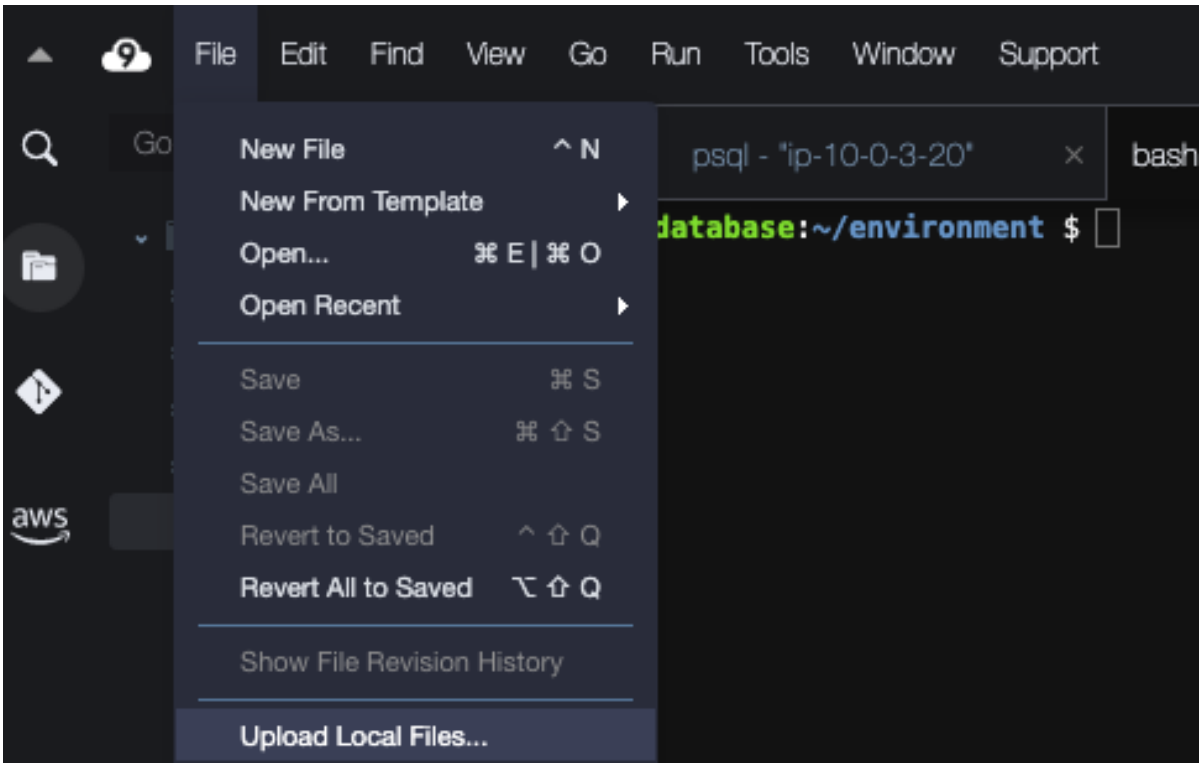
13. cloud9 터미널에서 psql 을 수행하여 postgreSQL database 접속 (관련 형식 `psql -h <위에서 복사한 endpoint>  -U <Username> -d <Database Name> -p <Port>`

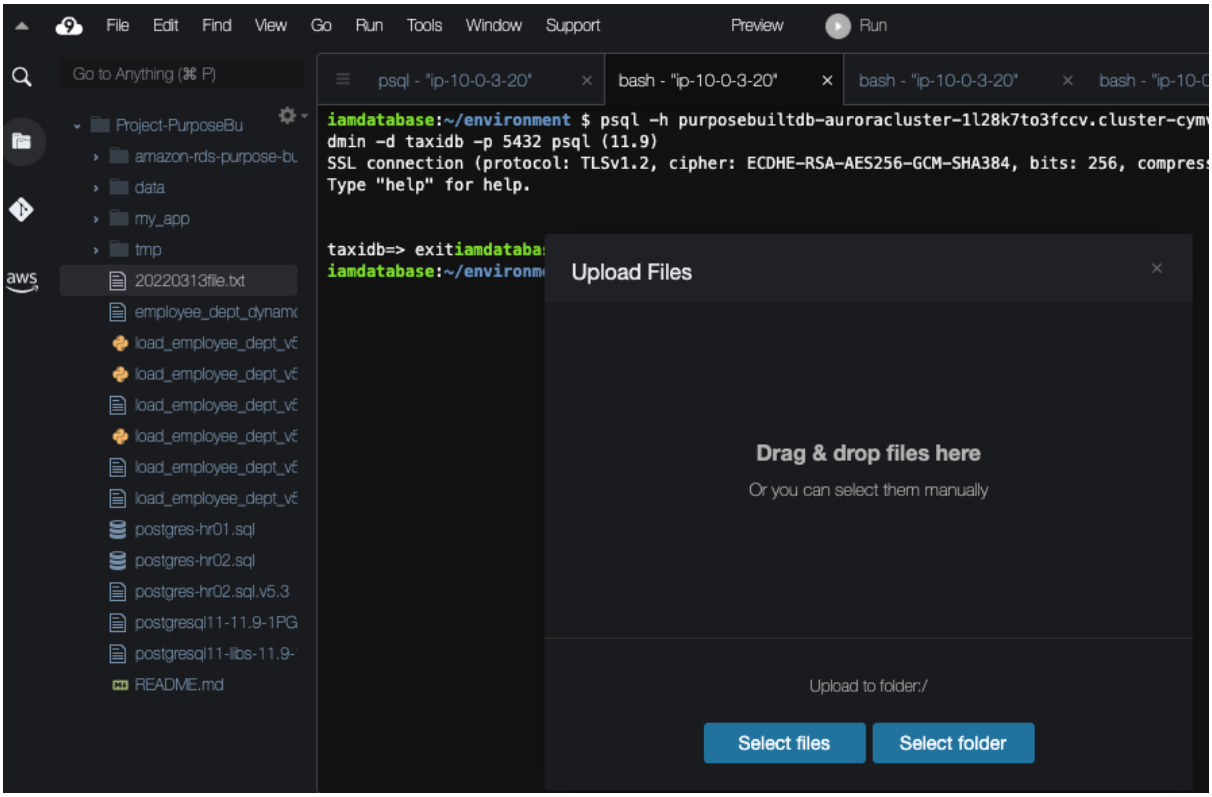
- 아래와 같이 taxidb=> 결과가 보이면 성공입니다.

```
# shell 에서 수행
cd ~/environment
export PGPASSWORD=auradmin123

# postgreSQL 에서 수행
psql -h <앞에서복사한 postgreSQL endpoint>  -U auradmin -d taxidb -p 5432
exit
```

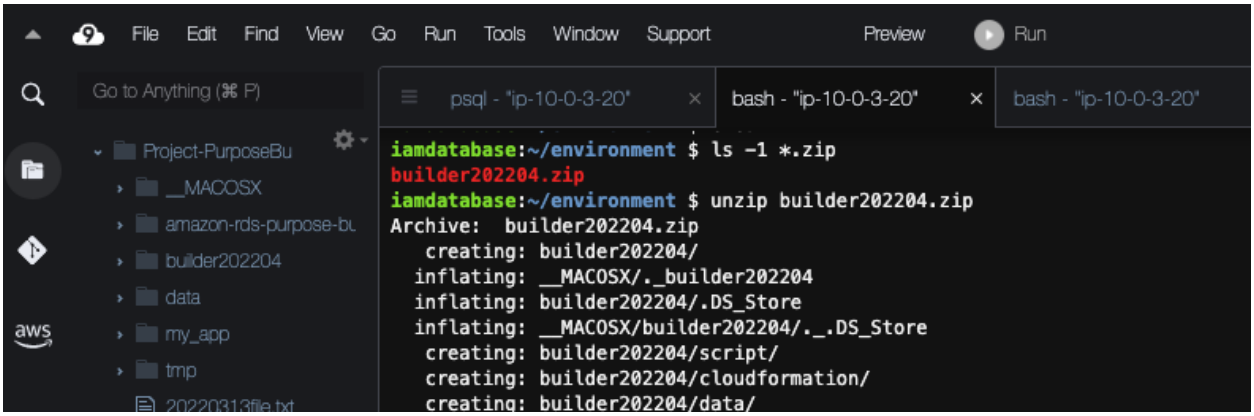


14. 터미널에서 cd ~/environment 수행후 → cloud9 메뉴 → file → upload local files 선택 → 팝업된 화면위에 다운로드한 builder202204.zip 파일 을 drag & drop —> upload files 화면 종료





15. 터미널에서 ls -1 *.zip을 통해 해당 builder202204.zip 파일이 아래와 같이 업로드 됨을 확인 —> unzip builder202204.zip 수행




```
# shell 에서 수행
# 파일이 존재함을 확인
ls -1 *.zip
```

```
# 해당 파일 압축 풀기
unzip builder202204.zip
```

- 수고하셨습니다. 다음 챕터로 이동하세요 —> 🧽 02) PostgreSQL 에서 초기데이터적재, 입력, 수정,삭제,조회 수행