1. **오라클 서버 설치**
   1. 사용 서버
      1. ip : 192.168.50.152
      2. CentOS 7.5
      3. Docker 참고
         1. <https://github.com/wnameless/docker-oracle-xe-11g.git>
   2. Docker oracle 서버 실행

|  |  |
| --- | --- |
| $ | docker run -d -p 49161:1521 --privileged --cap-add=SYS\_ADMIN -v /sys/fs/cgroup:/sys/fs/cgroup -v /root/oracle/dump:/root/dump --name oracle\_db wnameless/oracle-xe-11g-r2 |

* + 1. 위의 명령어는 docker 이미지를 다운로드 받고, 오라클 서버를 구동함
    2. Dump 파일은 호스트의 /root/oracle/dump/ 에 위치하고, docker 안에서는 /root/dump/ 에 위치함
    3. 해당 서버의 49161 포트로 오라클 서버 접속 가능
  1. Docker oracle 서버 종료 및 삭제
     1. 서버가 더 이상 필요 없으면 종료 및 삭제함 (한번 삭제하면 복구 할 수 없음)

|  |  |
| --- | --- |
| $ | docker stop oracle\_db |
| $ | docker rm oracle\_db |

1. **Dump 파일 Import**
   1. Dump 파일을 oracle 계정이 접근할 수 있는 위치로 복사하고 권한을 줌

|  |  |
| --- | --- |
| $ | cp -rf /root/dump/ /u01/app/oracle/dump |
| $ | cd /u01/app/oracle |
| $ | chown -R oracle dump/ |
| $ | chgrp -R dba dump/ |
| $ | su - oracle |
| $ | cd /u01/app/oracle/dump |

* 1. Oracle 내에서 해당 경로에 대한 이름을 만들고 접근 할 수 있게 함

|  |  |
| --- | --- |
| $ | sqlplus /nolog |
| SQL> | connect sys/oracle as sysdba |
| SQL> | drop directory dump; |
| SQL> | create directory dump as '/u01/app/oracle/dump/'; |
| SQL> | grant read, write on directory dump to public; |
| SQL> | grant create any directory to public; |

* 1. 경로가 정상적으로 생성됐는지 확인

|  |  |
| --- | --- |
| SQL> | SELECT \* FROM dba\_directories;  SYS                            DUMP  /u01/app/oracle/dump/  SYS                            DATA\_PUMP\_DIR  /u01/app/oracle/admin/XE/dpdump/  SYS                            XMLDIR  /u01/app/oracle/product/11.2.0/xe/rdbms/xml |

* 1. 이번에 사용한 오라클 dump 파일은 ‘FPS’ 라는 테이블스페이스를 필요로 함
     1. dump 파일을 vi 로 열면 ascii 판독 가능
     2. 테이블 생성 쿼리를 보면 테이블스페이스 이름 확인 가능

|  |  |
| --- | --- |
| SQL> | CREATE TABLESPACE FPS DATAFILE  'FPS' SIZE 2048M AUTOEXTEND ON NEXT 4M MAXSIZE UNLIMITED  LOGGING  PERMANENT  EXTENT MANAGEMENT LOCAL AUTOALLOCATE  BLOCKSIZE 8K  SEGMENT SPACE MANAGEMENT MANUAL  FLASHBACK ON |

* 1. SQL 모드에서 ‘exit’ 명령으로 빠져나와 쉘로 돌아감
  2. 쉘에서 import 를 실행
     1. dump 파일이 expdp 로 생성되었다면 impdp 를 사용, exp 로 생성되었다면 imp 를 사용
     2. ***이번에 사용한 오라클 dump 파일은 imp 를 이용함***
     3. impdp

|  |  |
| --- | --- |
| $ | impdp system/oracle@xe DIRECTORY=dump FULL=y DUMPFILE=track\_kowp0428.dmp |

* + 1. imp

|  |  |
| --- | --- |
| $ | imp system/oracle@xe FULL=y FIlE=track\_kowp0428.dmp  Import: Release 11.2.0.2.0 - Production on Fri May 8 05:51:27 2020  Copyright (c) 1982, 2009, Oracle and/or its affiliates.  All rights reserved.  Connected to: Oracle Database 11g Express Edition Release 11.2.0.2.0 - 64bit Production  Export file created by EXPORT:V11.02.00 via conventional path  Warning: the objects were exported by FPS, not by you  import done in US7ASCII character set and AL16UTF16 NCHAR character set  import server uses AL32UTF8 character set (possible charset conversion)  export client uses KO16MSWIN949 character set (possible charset conversion)  . importing FPS's objects into SYSTEM  . importing FPS's objects into SYSTEM  . . importing table           "TH\_TRACK\_KOWP\_0225"   12962290 rows imported  Import terminated successfully without warnings. |

1. **클라이언트 접속 및 Export**
   1. 이제 윈도우(원격 컴퓨터)에서 Oracle 서버에 접속함
   2. Oracle SQL Developer 는 오라클 서버에 접속할 수 있는 클라이언트 도구임, 해당 프로그램을 설치하고 실행
   3. 다음과 같은 정보로 접속
      1. user : system
      2. pw : oracle
      3. host : 192.168.50.152
      4. port : 49161
      5. sid : xe
   4. 접속하면 테이블 목록에 import 된 테이블이 보임 (‘TH\_TRACK\_KOWP\_0225’)
   5. 다음의 절차를 이용해서 csv 로 export 함
      1. 해당 테이블 이름 위에서 오른쪽 단추를 클릭
      2. 데이터 익스포트만 선택
      3. 형식 csv, 행 터미네이터 윈도우 CR/LF
      4. 경로를 설정하고 export 수행