

오라클 데이터베이스 다운로드하여 나만의 로컬 컨테이너 만들기 (가볍고 구성 가능한 Oracle 12c Docker 컨테이너)

오라클은 저작권 관련하여 매우 민감한 어플리 케이션이다.
도커 이미지들이 수시로 만들어지고, 수시로 사라지는 것도 모두 저작권 관련 문제이다.
오라클 도커를 삭제하고 다시 설치하려면
상당히 많은 스트레스를 받게 된다.

다행히 도커 오라클 빌드 툴이 공개되어 있어
자신만의 오라클 도커 이미지 컨테이너를 만들어 사용할 수 있다.

단, docker hub에 공개하려면 상당한 민감한 문제게 준비를 해야 할 것이다.

도커 빌드툴 클론 하기

아래 github에 접속하여 빌드툴을 클론한다.

<https://github.com/steveswinsburg/oracle12c-docker>

로컬디렉토리 적당한 곳에서 클론을 실행하여 빌드 툴 다운로드 받기

steveswinsburg / oracle12c-docker

Watch 0 Star 4 Fork 1

Code Issues 2 Pull requests 0 Projects 0 Security Insights

Join GitHub today Dismiss
GitHub is home to over 40 million developers working together to host and review code, manage projects, and build software together.
Sign up

A docker container for running Oracle 12c

14 commits 2 branches 0 releases 1 contributor Apache-2.0

Branch: master New pull request Find File Clone or download

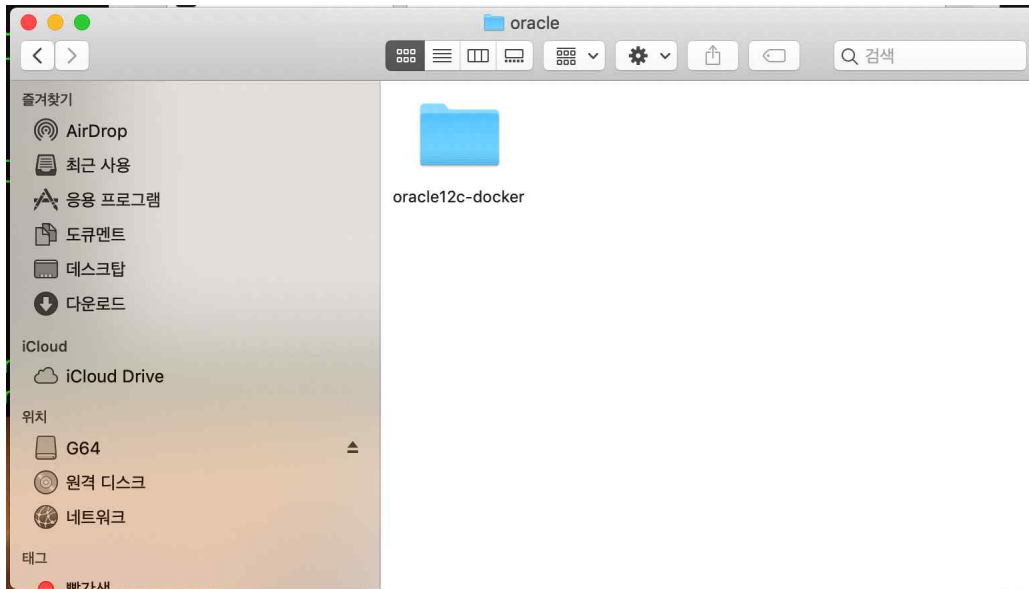
steveswinsburg Reorganise the building and running section of the readme.

.gitignore	Reorganise the build	
Dockerfile	Reorganise the build	
LICENSE	Initial commit	
README.md	Reorganise the building and running section of the readme.	2 years ago
checkDBStatus.sh	Reorganise the build	2 years ago
checkSpace.sh	Reorganise the build	2 years ago
createDB.sh	Update createDB.sh to allow more config to be passed in at run time	2 years ago
db_inst.sh	Reorganise the build	2 years ago








Clone with HTTPS
Use Git or checkout with SVN using the web URL.
<https://github.com/steveswinsburg/oracle12c-docker.git>
Open in Desktop Download ZIP

```
callor-MacBookPro:oracle callor$ pwd
/Users/callor/Downloads/oracle
callor-MacBookPro:oracle callor$ git clone https://github.com/steveswinsburg/oracle12c-docker.git
```

클론 한 후 oracle12c-docker 디렉토리



oracle.org에 접속하여 로그인 후 오라클 DBMS 리눅스 버전을 다운로드 받는다.

Oracle Database 12c Release 2		
(12.2.0.1.0) - Standard Edition 2 and Enterprise Edition		
Name	Download	Note
Microsoft Windows x64 (64-bit)	 File 1 (2.8 GB)	See All
Linux x86-64	 File 1 (3.2 GB)	See All
Oracle Solaris (SPARC systems, 64-bit)	 File 1 (3.1 GB)	See All
Oracle Solaris (x86 systems, 64-bit)	 File 1 (2.8 GB)	See All
HP-UX Itanium	 File 1 (3.7 GB)	See All
AIX (PPC64)	 File 1 (3.1 GB)	See All
Linux on System z (64-bit)	 File 1 (2.5 GB)	See All

다운로드 받은 파일을 oracle12c-docker 디렉토리에 옮겨 놓는다

※ 압축 파일을 해제 하지 말것!!!

이름	수정일	크기
checkDBStatus.sh	어제 오후 7:45	
checkSpace.sh	어제 오후 7:45	
createDB.sh	어제 오후 7:45	
db_inst.rsp	어제 오후 7:45	
dbca.rsp.tmpl	어제 오후 7:45	
Dockerfile	어제 오후 7:45	
installDBBinaries.sh	어제 오후 7:45	
LICENSE	어제 오후 7:45	
linuxx64_12201_database.zip	어제 오후 8:12	
README.md	어제 오후 7:45	
runOracle.sh	어제 오후 7:45	
runUserScripts.sh	어제 오후 7:45	
setPassword.sh	어제 오후 7:45	
setupLinuxEnv.sh	어제 오후 7:45	
startDB.sh	어제 오후 7:45	

빌드 실행

터미널에서 oracle12c-docker 폴더에서 다음 명령 실행

```
sudo docker build --force-rm=true --no-cache=true --shm-size=1G -t callor/oracle12c-ee .
```

가. callor/oracle12c-ee : 컨테이너 이미지 이름, 본인이 사용할 이름으로 설정

나. 반드시 sudo 계정으로 실행할 것, 그렇지 않으면 빌드 중간에 Fail이 발생할 수 있음

※ 꽤 많은 시간이 소요되므로 커피한잔 하며 잠시 휴식 ~~~

이미지 확인

docker 명령을 실행하여 컨테이너 이미지가 잘 생성 되었는지 확인

```
callor-MacBookPro:oracle callor$ docker images
REPOSITORY          TAG             IMAGE ID        CREATED         SIZE
callor/oracle12c-ee latest          484f2eb89fab   2 hours ago    13.3GB
oraclelinux          7-slim         874477adb545   4 weeks ago    118MB
kutsyy/oracle-12c    latest         848fb498dd10   2 years ago    5.69GB
callor-MacBookPro:oracle callor$
```

생성된 이미지로 오라클 실행

```
docker run \
--name callor-oracle-12c \
-p 1521:1521 -p 5500:5500 -p 9090:8080 \
-e ORACLE_SID=xe \
-e ORACLE_PWD=oracle \
-e ORACLE_MEM=4000 \
-v /opt/oracle/oradata:/u01/app/oracle \
-d \
callor/oracle12c-ee
```

가. callor-oracle12c : 오라클 실행 프로세스 이름

나. xe : DB Name, DB Sid

다. oracle : system 계정의 비밀번호

라. -v /opt/oracle/oradata:/u01/app/oracle : 데이터 저장, 도커폴더와 실제 저장폴더

마. callor/oracle12c-ee : 생성한 오라클 컨테이너 이미지

참조, 도커 표준 파라미터 변수 설명

```
--name : 컨테이너 이름 (설정하지 않으면 자동 생성)
-p : 컨테이너 포트에 대한 호스트 포트의 포트 매핑.
      1521 (Oracle Listener),
      5500 (OEM Express)
      8080 (Web Connect)
-e ORACLE_SID : 사용해야하는 Oracle Database SID (기본값 : ORCLCDB)
-e ORACLE_PWD : Oracle Database SYS, SYSTEM 및 PDB_ADMIN 비밀번호 (기본값 :
auto)
-e ORACLE_MEM : Oracle에 할당 할 메모리 용량 (MB). Docker에 더 많은 메모리를 할당하
기 위해 Docker 설정을 변경해야 할 수도 있음 (기본값 : 2048)
-v /opt/oracle/oradata 데이터베이스에 사용할 데이터 볼륨. 컨테이너 내부의 Unix
"oracle"(uid : 54321) 사용자가 write 할수 있어야 한다. 생략하면 데이터베이스가 컨테이
너 재 실행할 때 유지 되지 않는다.
-v /opt/oracle/scripts/startup | /docker-entrypoint-initdb.d/startup 옵션 : 데이터베이스
시작 후 실행할 사용자 정의 스크립트가 있는 볼륨.
-v /opt/oracle/scripts/setup | /docker-entrypoint-initdb.d/setup 옵션 : 데이터베이스 설정
후 실행할 사용자 정의 스크립트가있는 볼륨.
-d : 분리 모드로 실행 그렇지 않으면 Ctrl-C로 컨테이너가 종료된다.
```

SQL Plus 실행 접속 테스트

```
docker exec -it oracld-12c sqlplus
```