|  |
| --- |
| **1. 주제**  오픈 소스 프로젝트 배포    **분반, 팀, 학번, 이름**  오픈소스기초설계(나), 5팀,  20180381 윤승회,  20180367 박상혁,  20180403 허석문 |

|  |
| --- |
| **실습 #1 도커 버전 확인 및 기능 테스트**        <그림 실습#1-1>  그림 실습#1-1 은 docker system info 명령어를 실행했을 때 결과 화면입니다.      <그림 실습#1-2>  그림 실습#1-2 는 docker –version 명령어와 docker run hello-world를 실행했을 때 결과 화면입니다.  Docker의 버전이 20.10.20 으로 설치가 잘 된것을 확인할 수 있었습니다. |

|  |
| --- |
| **실습 #2 도커 환경에서의 코드 작성**      <그림 실습#2-1>  그림 실습#2-1 은 docker 컨테이너 내부에서 python3 의 버전을 확인하고 hello.py라는 파일의 내용을 출력하고 hello.py 를 실행한 결과창입니다.  이전에 과정은 다음과 같습니다.  1. Docker pull 명령어를 통해서 ubuntu:20.04를 다운로드 해주었습니다.  2. Docker images 를 확인하게 되면 ubuntu:20.04 이미지가 저장되어 있는 것을 확인 할 수 있었습니다.  3. Docker run –it –name (컨테이너 이름) ubuntu:20.04 명령어를 이용하여 컨테이너를 생성해주었습니다.  4. 컨테이너를 생성하면 처음 우분투를 설치하는 것과 동일한 환경이 됩니다. 따라서 apt를 업데이트 해주고 필요한 (vim, python3) 를 설치해 주었습니다.  5. Home 디렉토리로 이동하여 hello.py 코드를 작성한 다음 위의 명령어를 실행해 주었습니다.    <그림 실습#2-2>  그림 실습#2-2 는 사용중이던 컨테이너에서 나가고 생성되어 있는 컨테이너를 확인하는 명령어 docker ps –a를 통해서 컨테이너 정보를 확인해 주었습니다. 생성되어 있는 컨테이너를 다시 실행하는 경우는 docker start 명령어와 docker attach 명령어를 통해 실행, 컨테이너에 잘 들어가지는 것을 확인할 수 있었습니다. |

|  |
| --- |
| **실습 #3 도커 환경을 배포**      <그림 실습#3-1>  그림 실습#3-1은 ops 라는 이름의 컨테이너를 실행해주고 docker exec 명령어를 통해서 ops 내에 있는 hello.py 소스코드를 실행하는 명령어입니다. 실행이 잘되는 것을 확인하고 컨테이너를 종료해주는 docker stop 명령어를 통해 종료해주었습니다.      <그림 실습#3-2>  그림 실습#3-2는 사용중인 컨테이너를 이미지로 만드는 것입니다. Docker commit 명령어를 통해서 ops 이름의 컨테이너를 seunghoe/ops:0.01 이라는 이미지로 만들어 주었습니다. 앞에 seunghoe를 붙힌 이유는 도커홈페이지에 올리기 위해서는 닉네임과 동일해야하기 때문입니다. Commit 이후 이미지가 잘 생성된 것을 확인할 수 있습니다.      <그림 실습#3-3>  그림 실습#3-3에서는 docker push 명령어를 사용하여 저의 docker 계정에 seunghoe/ops:0.01 이미지를 올려주는 화면입니다.      <그림 실습#3-4>  그림 실습#3-4 docker push 이후 docker 홈페이지에 들어가보면 이미지가 public으로 잘 올라가진 모습을 확인할 수 있습니다. |

|  |
| --- |
| **실습 #4 배포한 환경을 검증**      <그림 실습#4-1>  그림 실습#4-1 화면은 현재 이미지를 확인후 실습3에서 올린 이미지 파일을 다운로드하고 이미지를 확인하는 화면입니다. 이전에 사용중이던 docker 이미지는 docker rmi 명령어를 이용, 컨테이너는 docker rm 을 이용하여 제거해주었습니다.      <그림 실습#4-2>  그림 실습#4-2 는 seunghoe/ops:0.01 이라는 이미지를 이용하여 copy\_ops 라는 이름의 컨테이너를 생성해 주었고, 바로 종료후 docker ps –a 를 통해서 컨테이너가 잘 생성된 것을 확인하였습니다. 다시 docker start 명령어를 통해 컨테이너를 실행, docker exec 명령어를 통해서 copy\_ops 안에 있는 hello.py 파일을 실행해 보았을 때, 잘 작동하는 것을 확인 할 수있었습니다. |

|  |
| --- |
| **실습 #5 도커 파일 작성 실습**      <그림 실습#5-1>  그림 실습#5-1 은 docker 파일을 생성해 주는 것입니다. Dockerfile 내부는 그림과 같이 작성해 주었습니다. 따라서 seunghoe/ops:0.01 이미지를 실행하면 hello docker 라는 결과가 나오는 것을 알 수 있습니다. FROM 은 사용할 이미지, RUN 과 CMD 를 이용해서 실행주었습니다. |