2021 대전광역시 제56회 전국기능경기대회 채점기준

1. 채점상의 유의사항	직 종 명	클라우드컴퓨팅

※ 다음 사항을 유의하여 채점하시오.

1) AWS의 지역은 ap-northeast-2을 사용합니다.
2) 웹페이지 접근은 크롬이나 파이어폭스를 이용합니다.
3) 웹페이지에서 언어에 따라 문구가 다르게 보일 수 있습니다.
4) shell에서의 명령어의 출력은 버전에 따라 조금 다를 수 있습니다.
5) 채점 진행 전 환경 셋업을 위해 다음 사항을 확인해야 합니다.
   - 각 1,2,3,4 채점 항목에 사용되는 VM(이하 VM)에 SSH로 접근 가능한지 확인합니다.
   - VM에서 curl, jq, awscli 가 설치 되었는지 확인합니다.
   - VM에서 IAM Role이 매핑되어 awscli로 AWS 모든 리소스에 접근 가능한지 확인 합니다.
   - aws sts get-caller-identity 명령을 통해 선수의 계정이 아닌 다른 계정에 접근하고 있는지 확인합니다. 만약 다른 계정이라면 부정행위를 의심할 수 있습니다.
6) 문제지와 채점지에 있는 <> 는 변수입니다. 해당 부분을 변경해 입력합니다.
7) 채점은 문항 순서대로 진행해야 합니다.
8) 삭제된 내용은 되돌릴 수 없음으로 유의하여 채점을 진행합니다.
9) 이의신청까지 종료된 이후 선수가 생성한 클라우드 리소스를 삭제합니다.
10) 채점내용은 영문기준으로 되어 있음으로 시작 전 언어를 영어로 변경하면 조금 더 채점내용과 동일한 결과를 볼 수 있습니다.

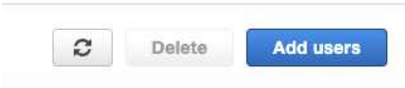
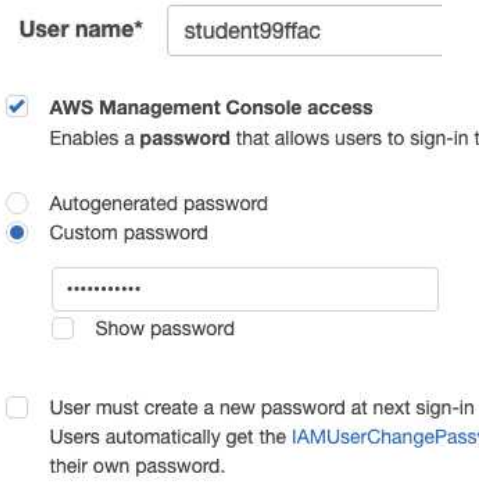
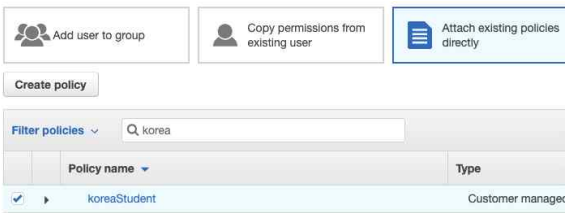
2. 채점기준표

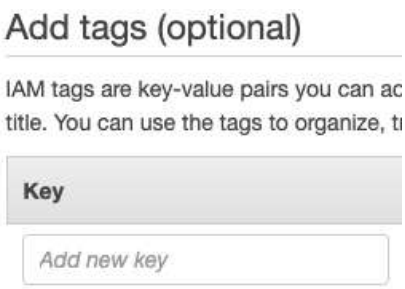
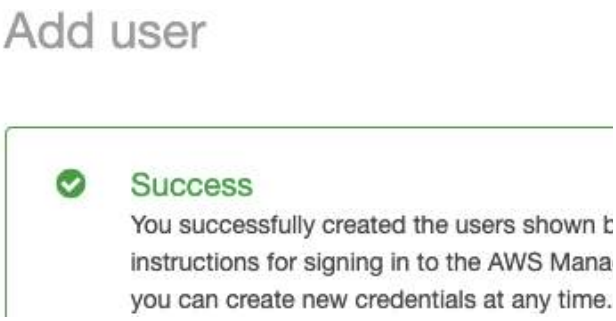
1) 주요항목별 배점			직 종 명			클라우드컴퓨팅		
과제 번호	일련 번호	주요항목	배점	채점방법		채점시기		비고
				독립	합의	경기 진행중	경기 종료후	
제3과제	1	IAM Policy	5		○		○	
	2	Lambda	5		○		○	
	3	Docker	5		○		○	
	4	Log data	5		○		○	
합 계			20					

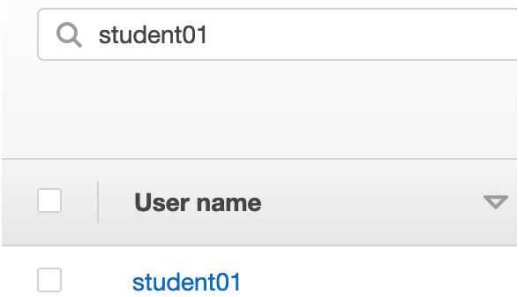
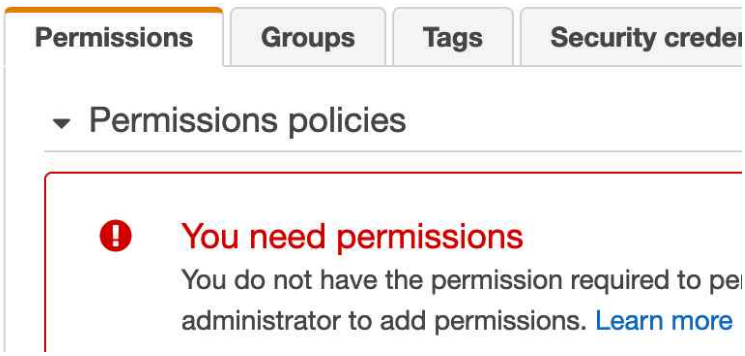
2) 채점방법 및 기준

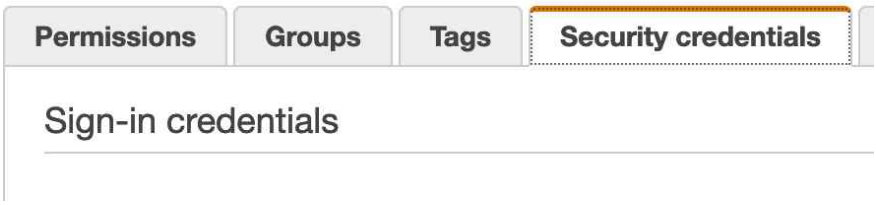
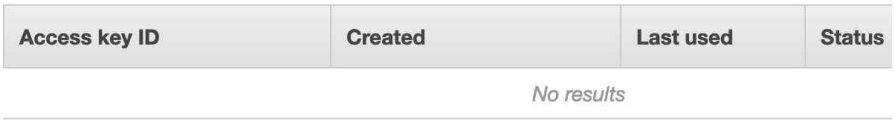
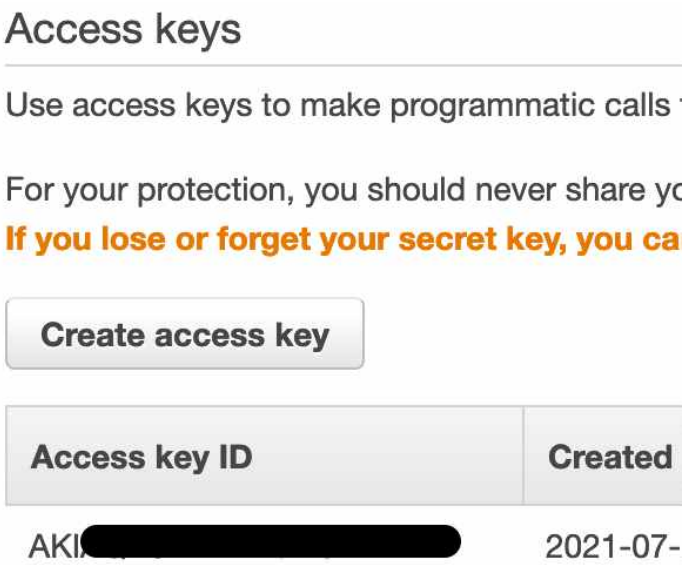
과제 번호	일련 번호	주요항목	일련 번호	세부항목(채점방법)	배점
제3과제	1	IAM Policy	1	IAM 차단 확인	0.7
			2	Access Key 수정 확인	1.2
			3	EC2 수정 확인	0.9
			4	EC2 차단 확인	1.1
			5	Access Key 차단 확인	1.1
	2	Lambda	1	Lambda 생성 확인	0.8
			2	버킷 생성 동작 확인	1.3
			3	파일 생성 확인	1.4
			4	파일 내용 확인	1.5
	3	Docker	1	ECR 생성 확인	0.9
			2	도커 이미지 업로드 확인	1.1
			3	API 호출 확인	1.5
			4	도커 프로세스 실행 확인	1.5
	4	Log data	1	1번 문제 확인	0.9
			2	2번 문제 확인	1.2
			3	3번 문제 확인	1.4
			4	4번 문제 확인	1.5
	총점				20

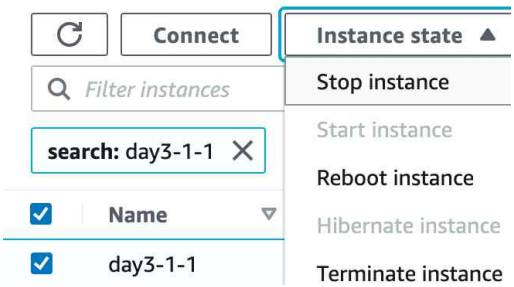
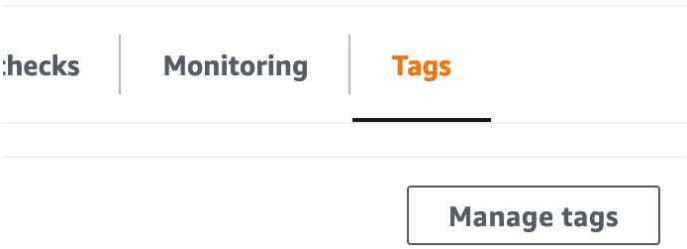
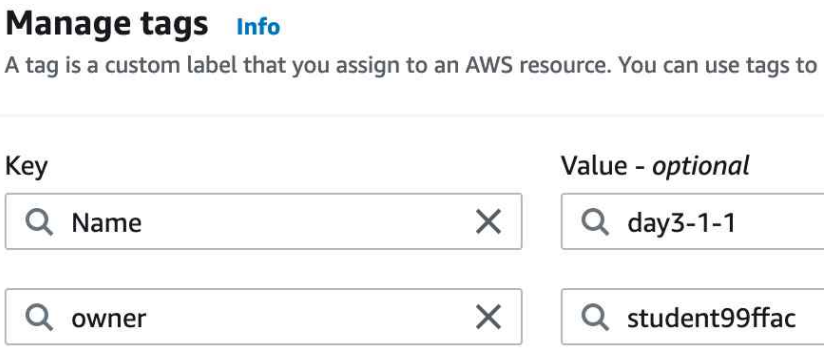
3) 채점 내용

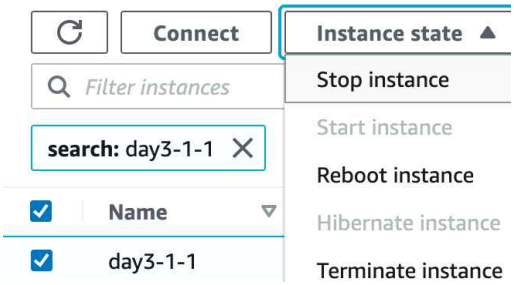
순번	채점 항목
사전 준비 IAM 계정 생성	1) 루트 계정으로 AWS 웹콘솔에 로그인하고, AWS IAM 페이지로 접근합니다. 2) IAM 페이지 왼쪽 Access management - Users에 접근합니다. 3) 오른편에 Add users를 클릭 합니다.  4) 아래와 같이 계정을 생성 합니다. 계정이름 = student99ffac / AWS 콘솔 엑세스 / 다음 로그인 시 비밀번호 변경 안함 해당 설정 후 권한 할당을 위해 다음 버튼을 누릅니다.  5) 아래와 같이 koreaStudent 라는 이름의 Policy 권한을 할당 합니다. 매우 중요한 작업입니다. 잘못하여 admin 등의 policy를 할당 하면 1-4, 1-5 채점 항목에 영향을 줄 수 있습니다. 선수가 추가하지 않아 Policy가 존재하지 않는다면 Policy를 할당하지 않고 다음으로 넘어갑니다.  뒷장에 계속

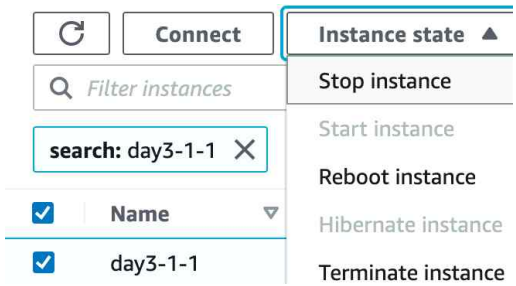
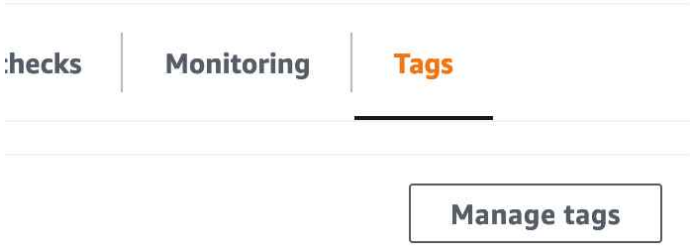
순번	채점 항목
사전 준비 IAM 계정 생성	6) Tag 부분은 빈값으로 유지하고 넘어 갑니다.  7) 최종적으로 생성 시 아래와 같은 Success 메시지를 볼 수 있습니다. 

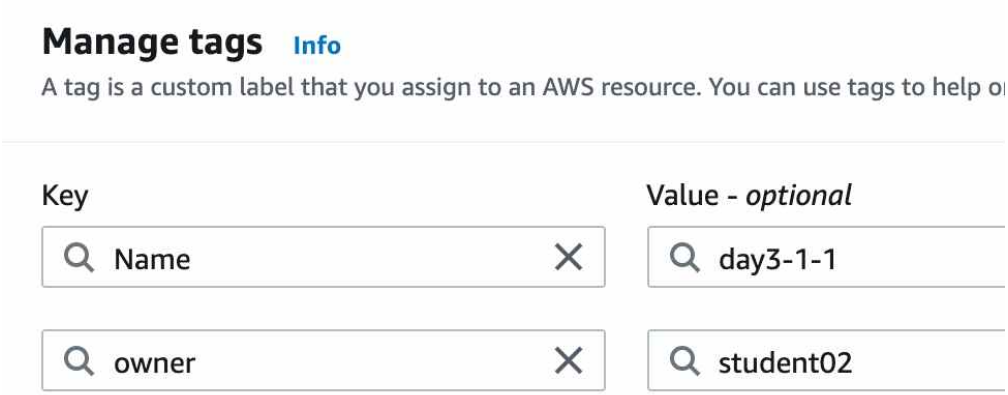
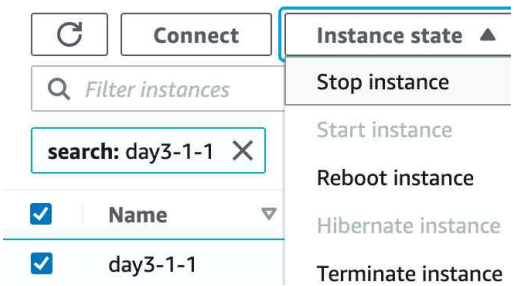
순번	채점 항목
1-1	1) student01 IAM 계정으로 AWS 웹페이지에 접근합니다. 2) AWS IAM 페이지에 접근합니다. 3) 웹페이지 왼쪽편 Access management 기능 아래에 Users 페이지를 클릭합니다. 4) Users 페이지 접근 후 student01 유저를 검색하여 클릭합니다.  5) Permissions 탭 접근 시 아래와 같이 권한 에러로 아무것도 보이지 않는지 확인합니다. 권한 차단에 대한 채점임으로 접근 불가, 권한 없음 등의 에러 메시지만 출력되어야 합니다. 

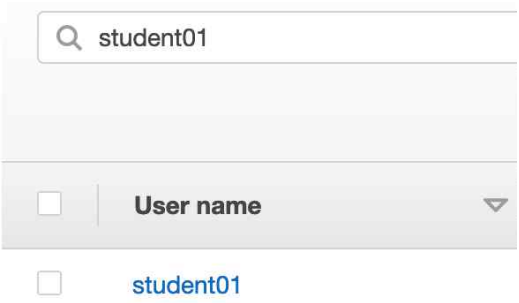
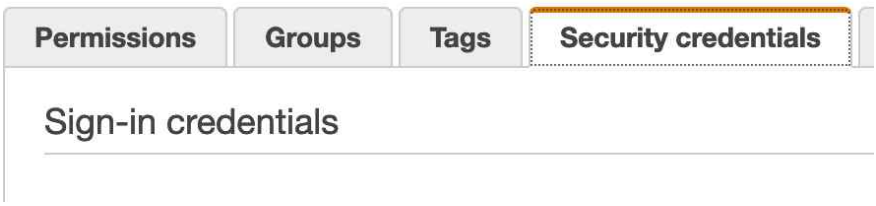
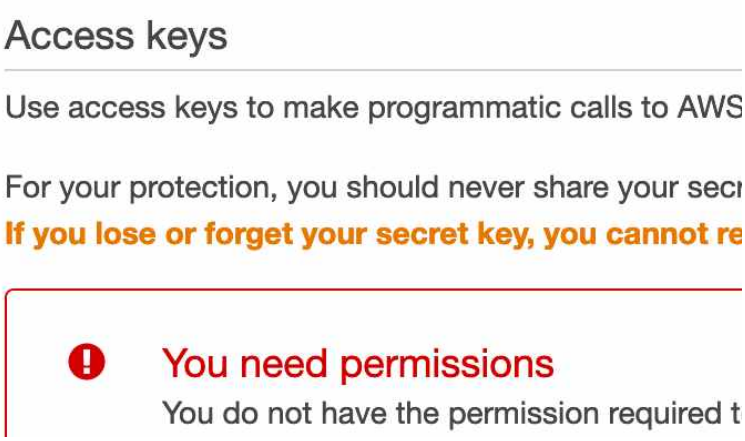
순번	채점 항목
1-2	1) 1-1 채점을 참고하여 student01 유저로 로그인 한 후 해당 유저의 IAM 페이지로 접근합니다. 2) student01 유저의 Security credentials 탭을 클릭합니다.  3) 계정의 Access Key가 생성 되어 있다면 모두 삭제하여 아래와 같은 상태로 구성합니다.  4) Create access key 버튼을 눌러 새로운 키 하나를 생성합니다. 5) 아래와 같이 새로운 키가 정상적으로 생성 되는지 확인합니다. 본인 계정의 Key 생성에 대한 테스트임으로 반드시 새로 생성하여 키가 정상적으로 생기는지 확인합니다. 

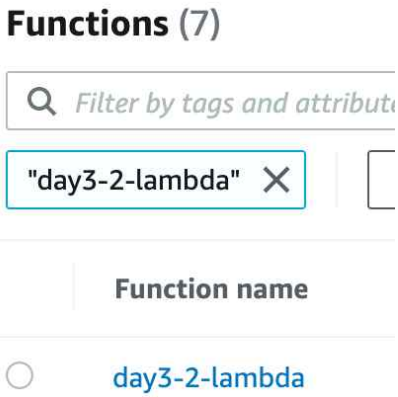
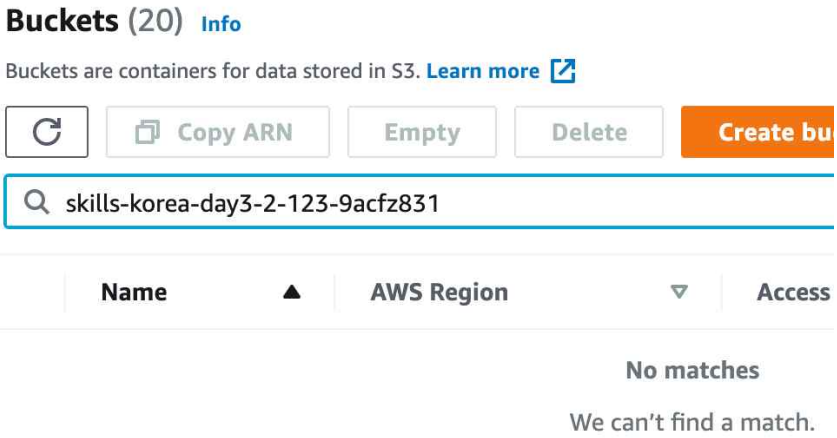
순번	채점 항목
1-3	1) 1-1 채점을 참고하여 student01 유저로 로그인 한 후 AWS EC2 페이지로 이동합니다. 2) EC2 페이지 접근 후 페이지의 왼쪽편 Instances 아래의 Instances를 클릭합니다. 3) day3-1-1 EC2를 검색하고, "Instance state" 버튼을 클릭합니다. EC2가 실행 중 이라면 Stop 하고, EC2가 중지되어 있으면 Start를 합니다.  4) 정상적으로 시작되거나 중지되는지 확인합니다. 5) day3-1-1 EC2의 "Manage tags" 버튼을 클릭합니다.  6) owner Tag 값을 student99ffac로 변경합니다. student01에서 student99ffac로 정상적으로 변경 되는지 확인합니다.  뒷장에 1-3 채점 계속

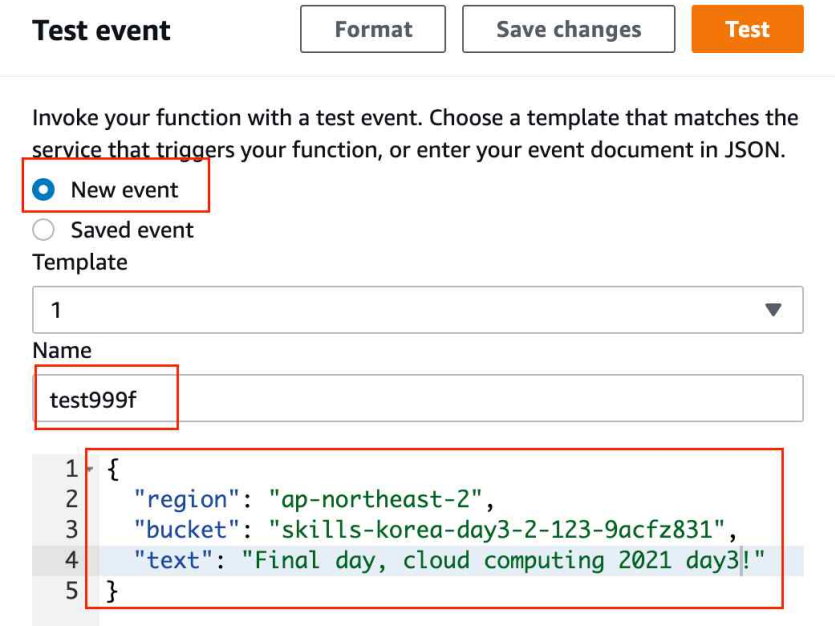
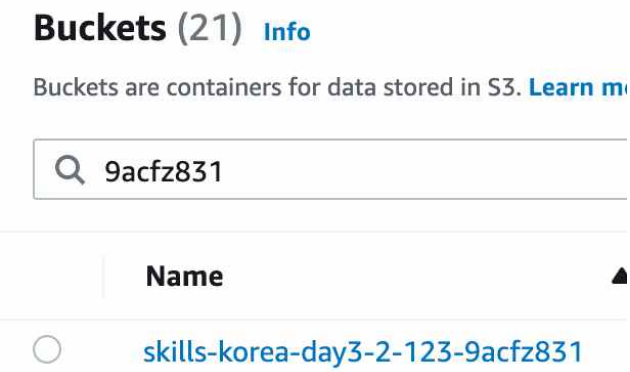
순번	채점 항목
1-3	7) day3-1-1 EC2의 상태를 다시 변경합니다. EC2가 실행 중 이라면 Stop 하고, EC2가 중지되어 있으면 Start를 합니다.  8) owner 정보 변경 후 해당 인스턴스 시작/중지 시 권한 에러가 나오는지 확인합니다.

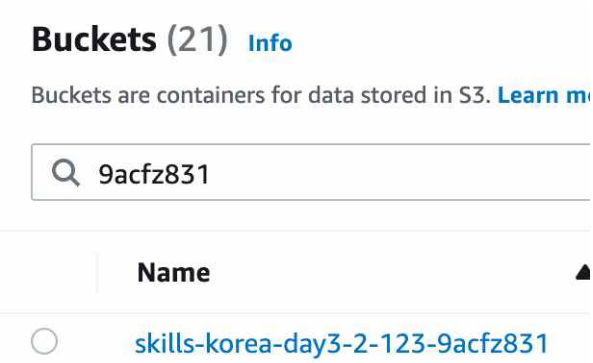
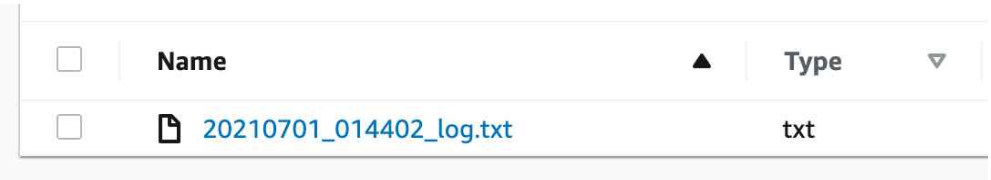
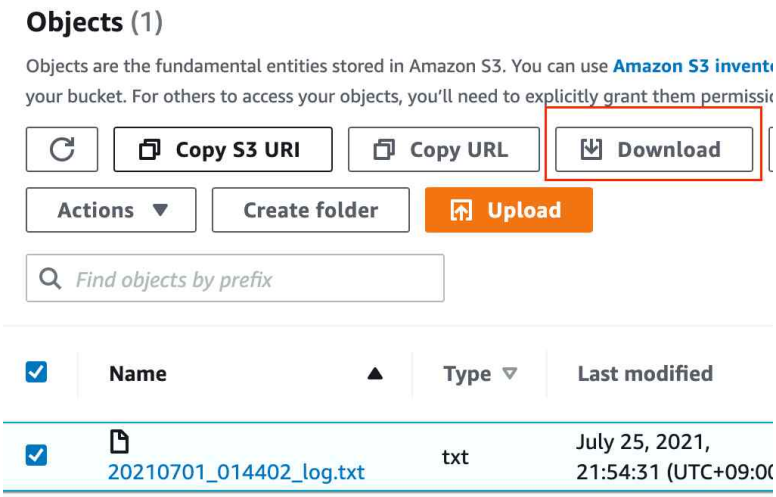
순번	채점 항목
1-4	1) student99ffac IAM 계정으로 AWS 웹페이지에 접근 후 EC2 페이지로 이동합니다. 2) 왼쪽 편 Instances에서 Instances를 클릭합니다. 3) day3-1-1 EC2를 검색하고, "Instance state" 버튼을 클릭합니다. EC2가 실행중 이라면 Stop 하고, EC2가 중지 되어 있으면 Start를 합니다.  4) 정상적으로 시작 되거나 중지 되는지 확인합니다. 5)day3-1-1 EC2의 "Manage tags" 버튼을 클릭 합니다.  뒷장에서 1-4 채점 계속

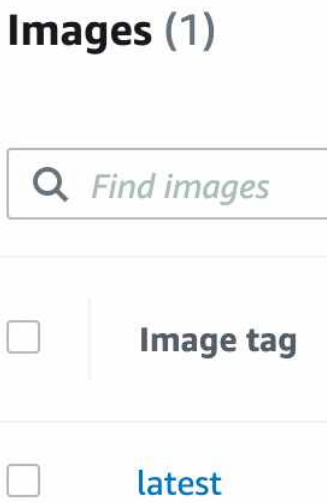
순번	채점 항목
	6) owner Tag 값을 student02로 변경합니다. student99ffac에서 student02로 정상적으로 변경 되는지 확인합니다. 
1-4	7) day3-1-1 EC2의 상태를 다시 변경 합니다. EC2가 실행중 이라면 Stop 하고, EC2가 중지 되어 있으면 Start를 합니다.  8) owner 정보 변경 후 해당 인스턴스 시작/중지시 권한 에러가 나오는지 확인합니다.

순번	채점 항목
1-5	1) 1-4 채점을 참고하여 student99ffac IAM 계정으로 AWS 웹페이지에 접근 합니다.. 2) AWS IAM 페이지에 접근 합니다. 3) 왼쪽편 Access management 기능 아래에 Users 페이지를 클릭합니다. 4) student01 유저를 검색하여 해당 유저를 클릭 합니다.  5) student01 유저의 Security credentials 탭을 클릭합니다.  6) 아래와 같이 Access keys 부분에 Key 리스트는 보이지 않고 에러가 보이는지 확인합니다. 권한 차단에 대한 채점으로 Key 리스트가 보이면 틀린 것으로 간주 합니다. 

순번	채점 항목
2-1	1) AWS 웹 콘솔에 IAM 유저가 아닌 루트 계정으로 로그인 합니다. 2) AWS Lambda 페이지로 이동하여 day3-2-lambda를 검색합니다. 해당 lambda 함수가 존재하는지 확인합니다. 
2-2	1) AWS 웹 콘솔에 IAM 유저가 아닌 루트 계정으로 로그인 합니다. 2) skills-korea-day3-2-<비번호>-<랜덤8자리글자> 형식의 버킷이름 하나를 지정합니다. 선수가 작업한 내용에 대한 채점이 아니라 생성할 버킷 이름 하나를 정하는 작업입니다. 타 계정에 존재할 수도 있으니 같은 이름의 버킷이 없도록 랜덤 8자리는 복잡하게 지정합니다. S3 페이지로 이동한 후 생성할 버킷이름 하나를 지정하여 현재 존재하지 않는지 확인합니다. 지정한 버킷이름을 메모하여 둡니다.  3) AWS Lambda 페이지로 이동하여 day3-2-lambda 함수를 클릭합니다. 뒷장에서 2-2 채점 계속

순번	채점 항목
2-2	4) Test 탭을 클릭합니다. 5) 아래와 같이 정보를 입력합니다. 제일 아래 박스에 있는 "bucket"에 입력하는 값은 2) 번에서 메모했던 버킷이름으로 변경하여 입력합니다. 내용 입력이 완료되면 우측상단에 있는 주황색 "Test" 버튼을 클릭합니다.  6) AWS S3 페이지로 이동해 버킷 이름을 검색하고, 해당 이름의 버킷이 생성되었는지 확인합니다. 

순번	채점 항목
2-3	1) AWS 웹 콘솔에 IAM 유저가 아닌 루트 계정으로 로그인 합니다. 2) AWS S3 페이지로 이동해 2-2 채점에서 생성한 버킷을 선택합니다.  3) 버킷 안에 존재하는 파일이름이 2-2 채점을 실행한 시간인지 확인합니다. <년도월일>_<시분초>_log.txt 형식의 이름을 가져야 합니다. 아래 그림에 있는 파일은 2021년도 07월 01일 01시 44분 02초에 생성한 파일입니다. 문제에 한국시간으로 명시 되어 있음으로 UTC 시간 기준이 아닌 현재 시간과 일치해야 합니다. 
2-4	1) AWS 웹 콘솔에 IAM 유저가 아닌 루트 계정으로 로그인 합니다. 2) 2-3번 채점항목에서 확인한 파일을 선택하여 다운로드 합니다.  3) 다운로드 받은 파일 내용이 Final day, cloud computing 2021 day3! 로 되어 있는지 확인합니다.

순번	채점 항목
3-1	1) AWS 웹 콘솔에 IAM 유저가 아닌 루트 계정으로 로그인 합니다. 2) AWS ECR 페이지로 이동한 뒤, 페이지의 왼쪽편 Amazon ECR - Repositories를 클릭합니다. 3) day3 repository를 찾아 클릭합니다. 4) 해당 repository에 아래와 같이 latest 라는 이미지가 존재 하는지 확인합니다. 

순번	채점 항목

3-2	1) 선수 AWS 계정에 있는 day3-3-ec2 라는 이름을 가진 EC2로 SSH 접속합니다.
	
	2) 아래 명령어를 차례로 입력하여 도커 컨테이너 및 이미지를 모두 삭제합니다.
	$ docker ps -a | grep -v CONTAINER | awk '{print "docker rm " $1}' | sh -x
	$ docker images | grep -v IMAGE | awk '{print "docker rmi " $3}' | sh -x
	
	3) 버전 확인 후 아래 명령어 중 하나를 통하여 AWS ECR에 로그인 합니다.
	(AWSCLI v2 기준 명령어)
	$ aws ecr get-login-password --region ap-northeast-2 | docker login --username AWS ₩
	--password-stdin < 선수 AWS 계정번호 >.dkr.ecr.ap-northeast-2.amazonaws.com
	
	(AWSCLI v1 기준 명령어)
	$ aws ecr get-login --region ap-northeast-2 --no-include-email | sh
	
	4) 아래 명령어를 입력해 이미지를 다운로드 합니다.
	$ docker pull < 선수 AWS 계정번호>.dkr.ecr.ap-northeast-2.amazonaws.com/day3:latest
	
	5) 아래 명령어를 입력해 컨테이너를 실행합니다.
	$ docker run -d -p 18888:8080 < AWS계정번호
	>.dkr.ecr.ap-northeast-2.amazonaws.com/day3:latest
	
	6) 아래 명령어를 통해 방금 실행한 컨테이너가 구동 되었는지 확인합니다.
	명령어 입력 후 선수 AWS 계정을 포함한 이미지 컨테이너의 STATUS가 Up이면 구동중입니다.
	$ docker ps
	
	CONTAINER ID IMAGE COMMAND CREATED STATUS
	5b6b5bbbb647 <선수계정>/day:latest "/bin/sh 3 minutes ago Up 3 minutes

3-3	1) 선수 AWS 계정에 있는 day3-3-ec2 라는 이름을 가진 EC2로 SSH 접속합니다. 2) 아래 명령어를 입력하여 컨테이너 서버로 HTTP request를 호출합니다. $ curl http://localhost:18888/health 3) 아래와 같은 결과를 받아 오는지 확인합니다. {"code":200,"msg":"ok"}

순번	채점 항목

3-4	1) 선수 AWS 계정에 있는 day3-3-ec2 라는 이름을 가진 EC2로 SSH 접속합니다.
	
	2) 아래 명령어를 입력하여 나오는 도커 container id를 기록합니다.
	$ docker ps | grep 'day3:latest' | awk '{print $1}'
	
	3) 아래 명령을 통해 컨테이너에 접속합니다.
	$ docker exec -it < 2)번에서 확인한 container id> sh
	
	4) 아래 명령어를 입력하여 결과값에 NAME="Alpine Linux"가 포함되어 있는지 확인합니다.
	$ cat /etc/os-release
	NAME="Alpine Linux"
	ID=alpine
	...
	
	5) 아래 명령어를 입력하여 Python 3.7.x 의 버전을 가지는지 확인합니다.
	$ python3 -V
	
	6) 아래 명령어를 입력하여 LISTEN 상태의 8080 포트를 가지고 있는지 확인합니다.
	$ netstat -an | grep 8080
	tcp 0 0 0.0.0.0:8080 0.0.0.0:* LISTEN

4-1	5,685개
4-2	1,291개
4-3	IP : 10.0.11.123, 횟수 : 682개
4-4	1순위 IP : 10.0.20.127, 횟수 : 2,388개 2순위 IP : 10.0.1.122, 횟수 : 1418개 3순위 IP : 10.0.21.161, 횟수 : 1002개