2024년도 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) 문제지와 채점지에 있는 <> 는 변수입니다. 해당 부분을 변경해 입력합니다.
6) 채점은 문항 순서대로 진행해야 합니다.
7) 삭제된 채점 자료는 되돌릴 수 없으므로 유의하여 진행하며, 이의신청까지 완료 이후 선수가 생성한 클라우드 리소스를 삭제합니다.
8) 부분 점수가 있는 문항은 채점 항목에 부분 점수가 적혀져 있습니다.
9) 부분 점수가 따로 없는 문항은 모두 맞아야 점수로 인정됩니다.
10) 리소스의 정보를 읽어오는 채점항목은 기본적으로 스크립트 결과를 통해 채점을 진행하며, 만약 선수가 이의가 있다면 명령어를 직접 입력하여 확인해볼 수 있습니다.
11) [ ] 기호는 채점에 영향을 주지 않습니다.
12) 채점 내용의 $ 기호는 명령어에 포함되는 것이 아니라 쉘을 의미합니다.

2. 채점기준표

1) 주요항목별 배점			직 종 명			클라우드컴퓨팅		
과제 번호	일련 번호	주요항목	배점	채점방법		채점시기		비고
				독립	합의	경기 진행중	경기 종료후	
제2과제	1	Cloud governance	3.75		○		○	
합 계								

2) 채점방법 및 기준

(경기종료 후 채점)

순번	사전준비
0	1) Bastion 서버에 SSH를 통해 접근합니다. (별도 명시가 없는 경우 모든 채점은 Bastion 서버에서 진행합니다.) 2) Bastion 명령어 및 권한을 확인합니다. (awscli permission, jq, curl, awscli region) 3) 아래 파일들을 Bastion 서버의 /root/marking 디렉터리로 복사합니다. - marking-gover.sh 4) 채점을 진행하는 Bastion 서버의 쉘을 초기 실행할 때 다음 명령어을 실행하여 환경 변수를 초기화합니다. (**채점 스크립트로 진행 시 생략**) export instance_name=wsi-test export sg_id=$(aws ec2 describe-instances --filters "Name=tag:Name,Values=$instance_name" --query "Reservations[].Instances[].SecurityGroups[].GroupId" --output json | jq -r '.[0]') export instance_id=$(aws ec2 describe-instances --filters "Name=tag:Name,Values=wsi-test" --query "Reservations[].Instances[].InstanceId" --output json \| jq -r '.[0]') export vpc_id=$(aws ec2 describe-instances --instance-ids $instance_id --query "Reservations[].Instances[].VpcId" --output json | jq –r '.[0]') aws configure set default.region ap-northeast-2 aws configure set default.output json

과제 번호	일련 번호	주요항목	일련 번호	세부항목(채점방법)	배점
2과제	1	자동 복구	1	connected SG	0.5
			2	not connected SG	0.75
			3	new connection SG	1.5
			4	port	1
	총점				3.75

3) 채점내용

<table>
<tr><th>순번</th><th colspan="2">채점 항목</th></tr>
<tr><td rowspan="2">1-1</td><td>1-1-A
(명령어 입력)</td><td>aws ec2 authorize-security-group-ingress --group-id $sg_id --protocol tcp
--port 443 --cidr 0.0.0.0/0 > /dev/null
sleep 180
aws ec2 describe-security-groups --group-ids $sg_id --query
"SecurityGroups[].IpPermissions[]" | jq '.[] | select(.ToPort == 443)'</td></tr>
<tr><td>1-1-A
(예상 출력)
<u>출력 없음</u></td><td></td></tr>
<tr><td rowspan="2">1-2</td><td>1-2-A
(명령어 입력)</td><td>export new_sg_id=$(aws ec2 create-security-group --group-name last
--description "New security group" --vpc-id $vpc_id --output text)
aws ec2 authorize-security-group-ingress --group-id $new_sg_id --protocol tcp
--port 22 --cidr 0.0.0.0/0 > /dev/null
aws ec2 authorize-security-group-ingress --group-id $new_sg_id --protocol tcp
--port 443 --cidr 0.0.0.0/0 > /dev/null
sleep 180
aws ec2 describe-security-groups --group-ids $new_sg_id --query
"SecurityGroups[].IpPermissions[]" | jq '.[] | select(.ToPort == 22)'</td></tr>
<tr><td>1-2-A
(예상 출력)
<u>정확히 일치</u></td><td>{
"FromPort": 22,
"IpProtocol": "tcp",
"IpRanges": [
{
"CidrIp": "0.0.0.0/0"
}
],
"Ipv6Ranges": [],
"PrefixListIds": [],
"ToPort": 22,
"UserIdGroupPairs": []
}</td></tr>
</table>

순번	채점 항목	
1-3	1-3-A (명령어 입력)	aws ec2 modify-instance-attribute --instance-id $instance_id --groups $new_sg_id sleep 180 aws ec2 describe-security-groups --group-ids $new_sg_id --query "SecurityGroups[].IpPermissions[]" | jq '.[] | select(.ToPort == 443)'
	1-3-A (예상 출력) **출력 없음**	
1-4	1-4-A (명령어 입력)	aws ec2 describe-security-groups --group-ids $sg_id --query "SecurityGroups[].IpPermissions[].ToPort" | jq '.[]'
	1-4-A (예상 출력) **순서 상관 X**	80 22 3306