

EMR Serverless Workshop

[Cloud9 대체 방법: 스택 생성시 오류 문제 해결]

<https://catalog.us-east-1.prod.workshops.aws/workshops/e8e8fbb5-c3fb-4f86-bf77-0ba1fe402c55/en-US>

원본 템플릿을 다운로드한다

<https://ws-assets-prod-iad-r-iad-ed304a55c2ca1aee.s3.us-east-1.amazonaws.com/e8e8fbb5-c3fb-4f86-bf77-0ba1fe402c55/emr-serverless-cfn-v2-self-paced.yml>

메모장으로 열어서 아래 cloud9 부분을 삭제하고 저장한다

Resources:

Cloud9EMRServerless:

Type: 'AWS::Cloud9::EnvironmentEC2'

Properties:

AutomaticStopTimeMinutes: 30

Description: Cloud9 for EMRServerless Workshop

InstanceType: t2.micro

ImageId: amazonlinux-2023-x86_64

Name: Cloud9 for EMRServerless Workshop

S3Bucket:

Type: 'AWS::S3::Bucket'

Properties:

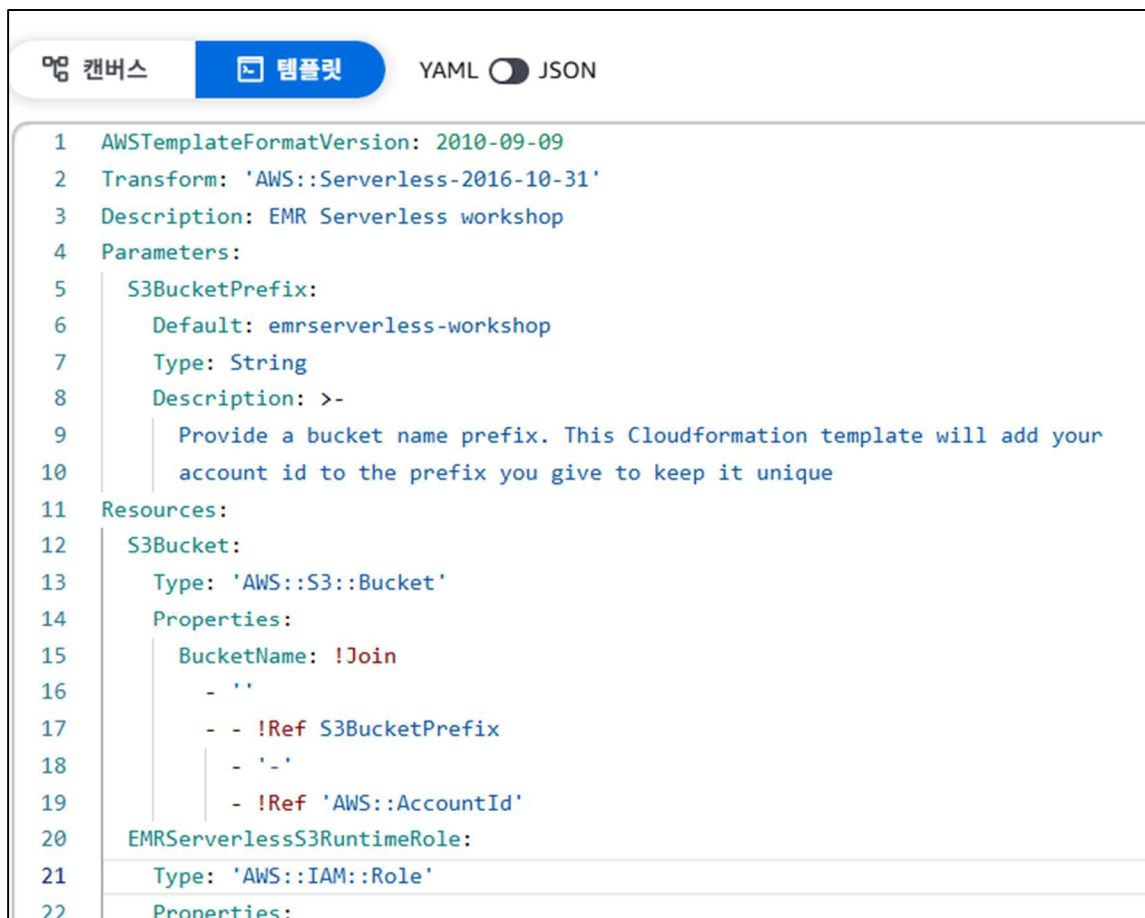
템플릿 파일 업로드 하영 수정된 파일을 업로드하고 다음 버튼을 누르면

파서 오류가 발생한다

아래의 인프라 컴포저에서 보기를 클릭한다



템플릿을 누르고 PC에서 템플릿 파일을 메모장으로 열어서 내용을 복사해서
인프라컴포저에 붙여 넣는다



우측 상단의 검증 버튼을 누르고 유효함으로 바뀌면 템플릿 생성 버튼을 누른다
확인후 CloudFormation 계속 진행을 누른다

CloudFormation 계속하기

i 이 기존 버킷 cf-templates--1jatso93tzxf-us-east-1에 템플릿을 배치하고 있습니다.
자세한 내용은 [CloudFormation 콘솔 모드에서 Infrastructure Composer 사용](#)을
참조하세요.

▶ 다른 버킷 사용

취소확인 후 CloudFormation 계속 진행

다음을 누른다

스택 생성

사전 조건 - 템플릿 준비

[IaC 생성기](#)에서 기존 리소스를 스캔하여 템플릿을 만들 수도 있습니다.

템플릿 준비

모든 스택은 템플릿을 기반으로 합니다. 템플릿은 JSON 또는 YAML 텍스트 파일로, 스택에 포함하려는 AWS 리소스에 대한 구성 정보가 들어 있습니다.

☐ 기존 템플릿 선택
기존 템플릿을 업로드하거나 선택합니다.

☒ 인프라 컴포저에서 빌드
비주얼 빌더를 사용하여 템플릿을 생성합니다.

인프라 컴포저에서 템플릿 생성

인프라 컴포저를 사용하여 간단한 드래그 앤 드롭 인터페이스에서 스택을 시각적으로 설계할 수 있습니다. 인프라 컴포저는 템플릿을 자동으로 업데이트하고 유효성을 검사합니다.

☒ 인프라 컴포저에서 템플릿을 가져왔습니다.

Amazon S3 URL

<https://s3.us-east-1.amazonaws.com/cf-templates--1jatso93tzxf-us-east-1/template-1744645052028.yaml>

[인프라 컴포저에서 편집](#)

취소다음

스택 이름을 임의로 (예. emr-temaplate-0415) 입력하고 이후는 스택 생성 과정 동일하
게 진행한다

5분 정도 기다리면 다음과 같이 생성 성공한 결과를 볼수 있다

스택 정보

이벤트

리소스

출력

파라미터

템플릿

변경 세트

Git Sync

출력 (9)

키	▲ 값	▼ 설명	▼ 내보내기 이름
EMRServerlessS3RuntimeRoleARN	arn:aws:iam::845992681665:role/EMRServerlessS3RuntimeRole	Amazon Resource Name (ARN) for EMRServerlessS3RuntimeRole: Substitute job-role-arn with this runtime role ARN	-
EMRServerlessSecurityGroup	sg-02d6e49790b9b6434	EMR Serverless Security Group	-
PrivateSubnet1	subnet-001e2b62303419f39	Private Subnet 1	-
PrivateSubnet2	subnet-0de78eca944f8f88f	Private Subnet 2	-
PrivateSubnet3	subnet-00c63c544943cfd12	Private Subnet 3	-
RedshiftServerlessNamespace	redshift-spark-nm	Redshift Serverless Namespace	-
RedshiftServerlessWorkgroup	redshift-spark-wg	Redshift Sereverless Workgroup	-
S3Bucket	emrserverless-workshop-845992681665	S3Bucket	-
VpcId	vpc-0745df7dabcfdc71a	VPC for redshift spark connector	-

[EC2 인스턴스 생성]

AWS 콘솔 EC2 → 인스턴스로 이동한다

이름:

emr-work-intance

애플리케이션 및 OS 이미지(Amazon Machine Image):

Amazon linux (기본값) 선택

인스턴스 유형 :

t3.small

키 페어(로그인) :

키 페어 없이 계속 진행

아래 세 개는 체크해준다

다음에서 SSH 트래픽 허용

인터넷에서 HTTPS 트래픽 허용

인터넷에서 HTTP 트래픽 허용

나머지는 기본설정으로 두고 인스턴스 시작을 클릭한다

EC2 인스턴스 생성 결과를 확인한다

옵션 2 - CLI에서 Spark 작업 제출 (약 4분 36초 소요)

Cloud shell 에서 실행해도 된다

```
export JOB_ROLE_ARN=arn:aws:iam::845992681665:role/EMRServerlessS3RuntimeRole
export S3_BUCKET=s3://emrserverless-workshop-845992681665
export APPLICATION_ID=00fro94g2q5nj409

aws emr-serverless start-job-run --application-id ${APPLICATION_ID} --execution-role-arn ${JOB_ROLE_ARN} --name "Spark-WordCount-CLI" --job-driver '{
    "sparkSubmit": {
        "entryPoint": "s3://us-east-1.elasticmapreduce/emr-containers/samples/wordcount/scripts/wordcount.py",
        "entryPointArguments": [
            ""$S3_BUCKET"/wordcount_output/"
        ],
        "sparkSubmitParameters": "--conf spark.executor.cores=1 --conf spark.executor.memory=4g --conf spark.driver.cores=1 --conf spark.driver.memory=4g --conf spark.executor.instances=1 --conf spark.hadoop.hive.metastore.client.factory.class=com.amazonaws.glue.catalog.metastore.AWSGlueDataCatalogHiveClientFactory"
    }
}'
```