AWS 구성 내역 사진

AWS CodePipeline

<u>개발자 도구</u> > CodePipeline > 파이프라인 > rmsoft-bookmanage-codepipeline

rmsoft-bookmanage-codepipeline

파이프라인 유형: V1











면수 네임스페이스 DeployVariables 입력 아티팩트 BuildArtifact
ApplicationName rmsoft-bookmanage-ec2
DeploymentGroupName rmsoft-bookmanage-deploygroup

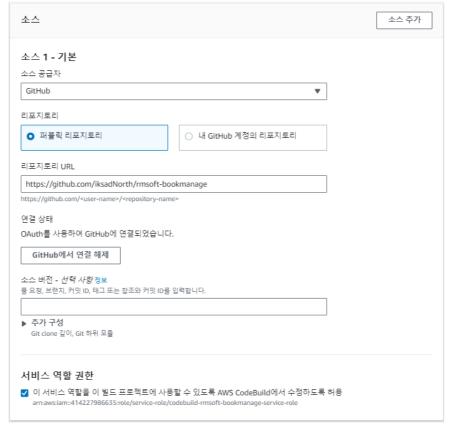
AWS CodeBuild

TEST_SPRING_DATASOURCE_USERNAME

TEST_SPRING_DATASOURCE_PASSWORD

a1w2e3r4





환경			
$0 \Pi X $ aws/codebuild/standard:7.0	환경 유형 Linux EC2	컴퓨팅 3GB 메모리, vCPU 2개	권한이 있음 false
서비스 역할 am:aws:iam::414227986635:role/service- role/codebuild-rmsoft-bookmanage-service-ro	제한 시간 1시간 0분 ste	대기 중인 제한 시간 8시간 0 분	인증서 -
▼ 환경 변수 이름	갔		유형
TEST_SPRING_DATASOURCE_URL	jdbc:mysqt://rmsoft-book-management.crmuxpc0pbxh.ap-northeast-2.rds.amazonaws.com:3306/bookmanage		PLAINTEXT

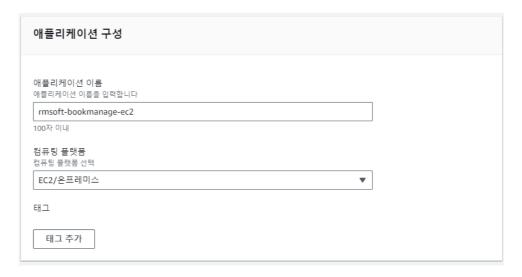
PLAINTEXT

PLAINTEXT

buildspec.yml

아티팩트	아티팩트 추가
아티팩트 1 – 기본	
유형	
Amazon S3 ▼	
테스트를 실행하거나 도커 이미지를 Amazon ECR에 넣는 경우 [아티팩트 영문]을 선택할 수 있습니다.	
버킷 이름	
Q rmsoft-bookmanage-artifact X	
이름 솔릭 아티젝트를 포함할 버킷에 있는 폴더 또는 압축 파일의 이름입니다. [추가 구성]에서 [아티젝트 패키징 압축 파일을 사용할지 선택합니다. 이름을 제공하지 않으면 기본적으로 프로젝트 이름으로 설정됩니다. rmsoft-bookmanage]]을 사용하여 폴더를 사용할지 아니면]
THISOTE-DUOKITIATIAGE	_
□ 의미 체계 버전 관리 활성화	
buildspec 파일에 지정된 아티팩트 이름 사용	
경로 - 선택 사항	
별드 출력 ZIP 파웬 또는 폴더의 경로입니다.	7
데: MyPath/MyArtifact.zip	
네임스페이스 유형 - 선택 <i>사항</i> 없음 ▼	7
BL을 [벨드 ID]를 선택하여 벨드 슐릭 ZIP 파일 또는 폴더의 경로(여: MyPath/MyBuildID/MyArtifact.zip)에 벨드	ID를 사이하니다. 또는 F에요1은 서태
합니다.	ive a de Te (maje e
아티팩트 패키징	
○ 없음 아티팩트 파일이 버킷에 업로드립니다. AWS CodeBuild가 아티팩 암숙 파일은 지정된 버킷	레트를 압축 파일로 업로드합니다(이 에 넣음).
□ 아티팩트 암호화 비활성화 아티팩트를 사용하여 정적 웹 사이트를 게시하거나 다른 사용자와 콘텐츠를 공유할 경우 암호화를 제기	계합니다.
서비스 역할 권한	
☑ 이 서비스 역할을 이 별도 프로젝트에 사용할 수 있도록 AWS CodeBuild에서 수정하도록 armaws.iam::414227986635:role/service-role/codebuild-rmsoft-bookmanage-service-role	· 허용
▶ 추가 구성 캐시, 압호화 기	

AWS CodeDeploy





환경 구성: Amazon EC2 인스턴스 키 값 Name rmsoft-bookmanage-ec2

개정 세부 정보

개정 위치 생성 날짜

s3://rmsoft-bookmanage-artifact/rmsoft-bookmanage 1일 전 설명

Application revision registered by Deployment ID: d-BP46NVTJ2

```
version: 0.0
os: linux
ifiles:
    - source: /
    destination: /home/ubuntu/
    overwrite: yes

ihooks:
    ApplicationStop:
    - location: scripts/kill_old_process.sh
        timeout: 60
        runas: root
    ApplicationStart:
    - location: scripts/deploy.sh
        timeout: 60
        runas: root
    - location: scripts/set_port_forwarding.sh
        timeout: 60
        runas: root
```

appspec.yml

```
ps -ef | grep jar | grep -v 'grep' | awk '{print $2}' | xargs --replace=@ sudo kill -9 @
```

scripts/kill_old_process.sh.sh

```
ROOT_DIR=/home/ubuntu;

sudo chmod +x $ROOT_DIR/build/libs/rmsoft-0.0.1-SNAPSHOT.jar

source $ROOT_DIR/scripts/set_env.sh
nohup java -jar $ROOT_DIR/build/libs/rmsoft-0.0.1-SNAPSHOT.jar >> $ROOT_DIR/nohup.out 2> $ROOT_DIR/nohup.err &
```

scripts/deploy.sh