Using Deploy Command to Create your Stacks

The process involves using the aws cloudformation deploy command to create and update AWS stacks. First, ensure the CloudFormation template is in the correct folder. Use the deploy command with options like --stack-name, --template-file, and --parameter-overrides to create a stack. Check the stack's status and resources with describe-stacks and describe-stack-resources. To update the stack, modify parameters with deploy and confirm updates via describe-stacks. Finally, delete the stack using delete-stack and verify its deletion progress. The goal is to automate stack creation, updates, and cleanup via AWS CLI.

Activity

1. Find the template files in our GitHub repository under the same name as the heading for easy access and edits. Find and Save the attached template locally, open it in VS Code for edits.
2. Open the terminal and ensure you are in the folder containing your CloudFormation template.

```
PS C:\Users\Ishika> aws s3 mb s3://stack-general
make_bucket: stack-general
PS C:\Users\Ishika> aws s3 ls
>>
2024-11-21 01:12:20 aws500demotest0911
2024-12-01 10:51:22 awsbackupfeaturerds1
2024-11-21 01:12:27 awsbucketdkrecrdemo
2024-11-21 01:12:42 awsdemoapplatest0910
2024-11-21 01:12:42 awsdemocloudformation11
2024-11-21 01:12:42 awsdeploymentconfigbucket1
2024-12-01 20:31:06 awsfiletransferdemo1
2024-11-29 07:44:43 awspurpledemobucket1
2024-11-21 01:13:12 awssimplilearnd1demo0911
2024-11-23 15:19:59 bucket-1-general
2024-12-09 12:16:17 bucket-database-stack
2024-11-21 01:37:38 cf-templates-kmi81w4ukk76-ap-south-1
2024-11-28 14:57:14 cf-templates-kmi81w4ukk76-eu-west-1
2024-12-09 10:06:36 databse-stack-bucket
2024-11-29 10:10:14 nestedstackbuckets3
2024-12-06 21:51:07 riteshrotarys3cli
2024-11-19 16:29:17 s3-bucket-12010321
2024-12-01 20:03:00 s3demosgbucketdemo
2024-12-09 19:05:19 stack-general
```

3. Use aws cloudformation deploy to create a stack, specifying the stack name with --stack-name (e.g., DatabaseStack). Provide the template path using --template-file (e.g., database-stack-template.yaml). Optionally, provide an S3 bucket with

--s3-bucket to upload the template if its size exceeds the limit. Use --s3-prefix (optional) to organize templates within the S3 bucket. Provide parameters using --parameter-overrides with a simplified format (e.g., VpcId=<value> and DbSubnets=<value1>,<value2>). Execute the command; it will automatically create a change set, execute it, and poll the stack status.

```
PS C:\Users\Ishika> aws cloudformation deploy-stack-name Stack2-template-file "C:\internship Section 717.4. USING DEPLOY
COMMAND TO CREATE YOUR STACKS database-stack-template.yanl"--s3-bucket perikiz-parameter-overrides VpcId=vpc-ebec8badfef0
824a4 DbSubnets-subnet-061947966364884af, subnet-08c772de46d053876, subnet-Ba1bbb2feaa84bb95
```

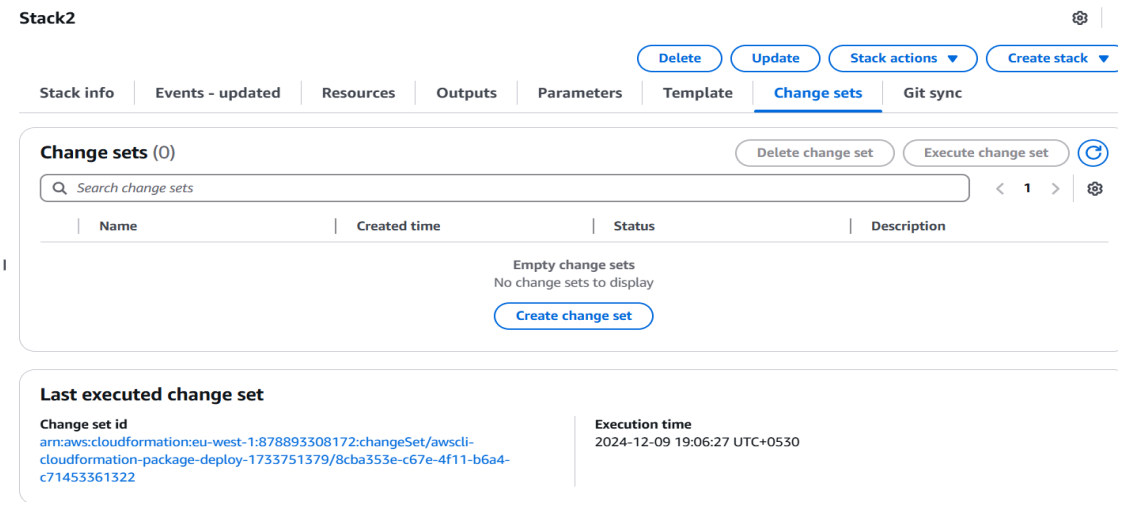
4. Confirm the stack creation is complete with aws cloudformation describe-stacks using the stack name.

```
PS C:\Users\Ishika> aws cloudformation describe-stacks --stack-name Stack2
{
    "Stacks": [
        {
            "StackId": "arn:aws:cloudformation:eu-west-1:878893308172:stack/Stack2/93f34b30
-b632-11ef-8b62-0254da007831",
            "StackName": "Stack2",
            "ChangeSetId": "arn:aws:cloudformation:eu-west-1:878893308172:changeSet/awscli-
cloudformation-package-deploy-1733751379/8cba353e-c67e-4f11-b6a4-c71453361322",
            "Description": "Sample database stack template for AWS CLI section.  It creates
 an Amazon RDS instance. ",
            "Parameters": [
                {
                    "ParameterKey": "VpcId",
                    "ParameterValue": "vpc-0b0c8badfef0024a4"
                },
                {
                    "ParameterKey": "DbSubnets",
                    "ParameterValue": "subnet-0619479663b4084af,subnet-08c772de46d053876,su
bnet-0a1bbb2feaa84bb95"
                },
                }
            ],
            "CreationTime": "2024-12-09T13:36:21.976000+00:00",
            "LastUpdatedTime": "2024-12-09T13:36:27.985000+00:00",
            "RollbackConfiguration": {},
            "StackStatus": "CREATE_COMPLETE",
            "DisableRollback": false,
            "NotificationARNs": [],
```

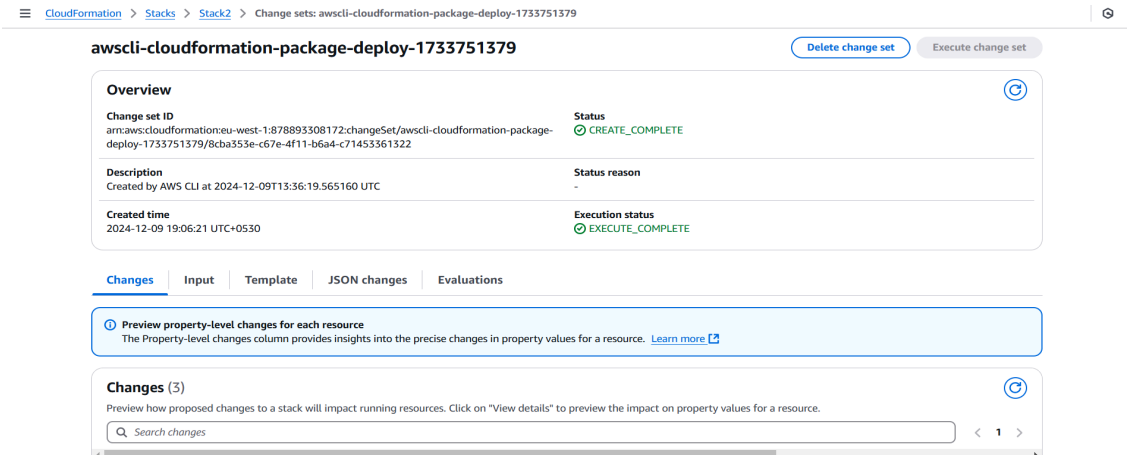
5. View stack resources using aws cloudformation describe-stack-resources with the stack name.

```
PS C:\Users\Ishika> aws cloudformation describe-stack-resources --stack-name Stack2
{
    "StackResources": [
        {
            "StackName": "Stack2",
            "StackId": "arn:aws:cloudformation:eu-west-1:878893308172:stack/Stack2/93f34b30
-b632-11ef-8b62-0254da007831",
            "LogicalResourceId": "DbInstance",
            "PhysicalResourceId": "stack2-dbinstance-g6am4tguquj4",
            "ResourceType": "AWS::RDS::DBInstance",
            "Timestamp": "2024-12-09T13:39:43.454000+00:00",
            "ResourceStatus": "CREATE_COMPLETE",
            "DriftInformation": {
        },
        {
        },
```

6. Check the change set created by the deploy command in the AWS Management Console under the stack's "Change sets" tab.



7. Confirm the change set's execution status as EXECUTE_COMPLETE and view details of added resources.



Using Deploy Commands to Update your Stacks

1. Find the template files in our GitHub repository under the same name as the heading for easy access and edits. Find and Save the attached template locally, open it in VS Code for edits.
2. Clear the screen and navigate to the folder with the template.
3. Use aws cloudformation deploy to update the stack with the --stack-name option.
4. Provide the --template-file option with the template path (no need for S3 options if the template is small). Use --parameter-overrides to specify only the parameters being

modified, e.g., AllocatedStorage=10. Execute the command to update the stack and

```
PS C:\Users\Ishika> aws cloudformation deploy --stack-name Stack2 --template-file "C:\inter
nship\Section 7\7.4. USING DEPLOY COMMAND TO CREATE YOUR STACKS\database-stack-template.yam
l" --s3-bucket stack-general --parameter-overrides AllocatedStorage=10

Waiting for changeset to be created..
Waiting for stack create/update to complete
Successfully created/updated stack - Stack2
```

5. Use aws cloudformation describe-stacks to verify the update status as UPDATE_COMPLETE.

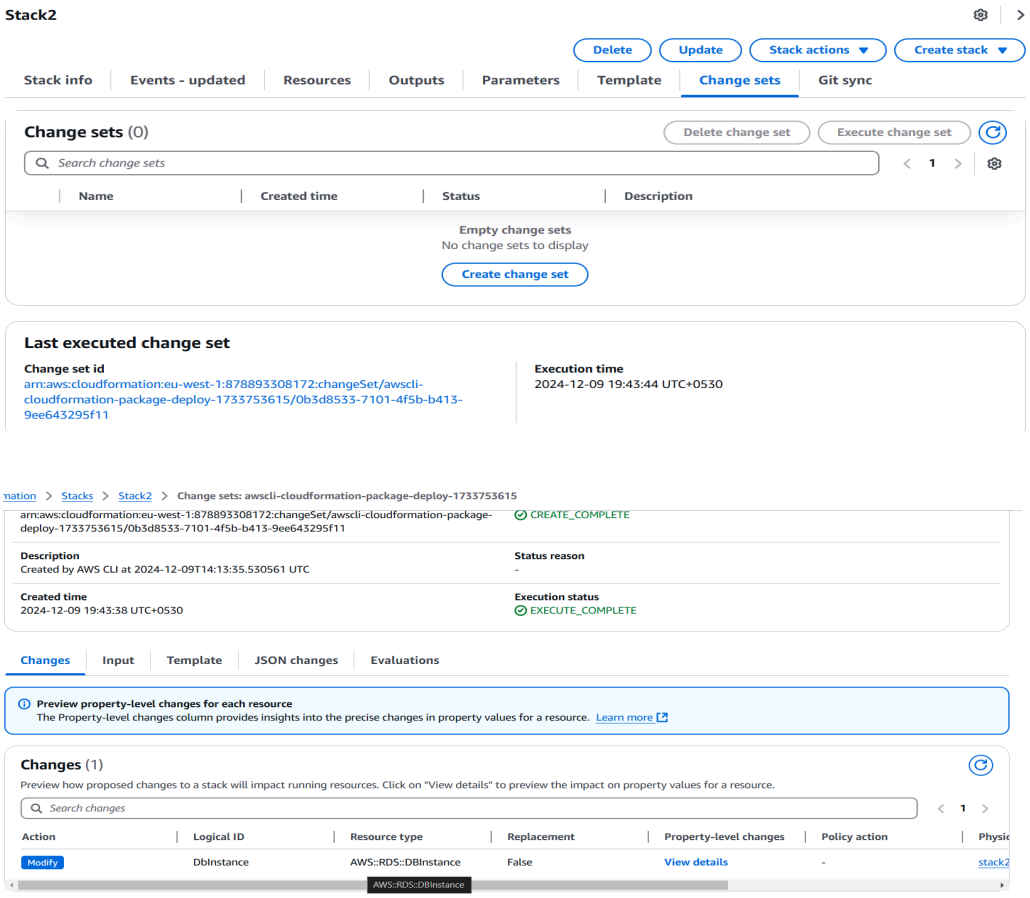
```
PS C:\Users\Ishika> aws cloudformation describe-stacks --stack-name Stack2
{
    "Stacks": [
        {
            "StackId": "arn:aws:cloudformation:eu-west-1:878893308172:stack/Stack2/93f34b30
-b632-11ef-8b62-0254da007831",
            "StackName": "Stack2",
            "ChangeSetId": "arn:aws:cloudformation:eu-west-1:878893308172:changeSet/awscli-
cloudformation-package-deploy-1733753615/0b3d8533-7101-4f5b-b413-9ee643295f11",
            "Description": "Sample database stack template for AWS CLI section.  It creates
 an Amazon RDS instance. ",
            "Parameters": [
                {
                    "ParameterKey": "VpcId",
                    "ParameterValue": "vpc-0b0c8badfef0024a4"
```

```
        "LastUpdatedTime": "2024-12-09T14:13:44.409000+00:00",
        "RollbackConfiguration": {},
        "StackStatus": "UPDATE_COMPLETE",
        "DisableRollback": false,
```

6. Use aws cloudformation describe-stack-resources to view the resources' statuses and confirm updates.

```
PS C:\Users\Ishika> aws cloudformation describe-stack-resources --stack-name Stack2
{
    "StackResources": [
        {
            "StackName": "Stack2",
            "StackId": "arn:aws:cloudformation:eu-west-1:878893308172:stack/Stack2/93f34b30
-b632-11ef-8b62-0254da007831",
            "LogicalResourceId": "DbInstance",
            "PhysicalResourceId": "stack2-dbinstance-g6am4tguquj4",
            "ResourceType": "AWS::RDS::DBInstance",
            "Timestamp": "2024-12-09T14:33:31.592000+00:00",
            "ResourceStatus": "UPDATE_COMPLETE",
            "DriftInformation": {
                "StackResourceDriftStatus": "NOT_CHECKED"
            }
        },
        {
            "StackName": "Stack2",
            "StackId": "arn:aws:cloudformation:eu-west-1:878893308172:stack/Stack2/93f34b30
-b632-11ef-8b62-0254da007831",
            "LogicalResourceId": "DbSecurityGroup",
            "PhysicalResourceId": "sg-0eb2b585beb040af6",
            "ResourceType": "AWS::EC2::SecurityGroup",
            "Timestamp": "2024-12-09T13:36:38.196000+00:00",
            "ResourceStatus": "CREATE_COMPLETE",
            "DriftInformation": {
                "StackResourceDriftStatus": "NOT_CHECKED"
```

7. Check the AWS CloudFormation Console for the change set details, showing only modified resources.



8. Use aws cloudformation delete-stack to clean up the stack and its resources.

```
PS C:\Users\Ishika> aws cloudformation delete-stack --stack-name Stack2
PS C:\Users\Ishika> aws cloudformation describe-stacks --stack-name Stack2
{
    "Stacks": [
        {
            "StackId": "arn:aws:cloudformation:eu-west-1:878893308172:stack/Stack2/93f34b30
-b632-11ef-8b62-0254da007831",
            "StackName": "Stack2",
            "Description": "Sample database stack template for AWS CLI section.  It creates
 an Amazon RDS instance. ",
            "Parameters": [
```

```
        "DeletionTime": "2024-12-09T14:47:03.051000+00:00",
        "LastUpdatedTime": "2024-12-09T14:13:44.409000+00:00",
        "RollbackConfiguration": {},
        "StackStatus": "DELETE_IN_PROGRESS",
        "DisableRollback": false,
        "NotificationARNs": [],
        "Tags": [],
        "EnableTerminationProtection": false,
        "DriftInformation": {
```

9. Confirm deletion progress with the describe-stacks command.