Creating Stacks With AWS CLI

The process involves creating and updating AWS CloudFormation stacks using the AWS CLI. First, download the template file and place it in your working folder. Use aws s3 mb to create a bucket and aws s3 cp to upload the template. Then, create a stack with aws cloudformation create-stack, specifying the template's S3 URL and required parameters (e.g., VPC ID, DB subnets). Check the stack status with describe-stacks and monitor progress using wait stack-create-complete. For updates, use update-stack and adjust parameters, then monitor the update with wait stack-update-complete.

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. Download the database stack template from the resources and place it in your working folder.
3. Open a terminal and ensure you're in the folder containing the template file.
4. Check for the available buckets.

```
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
2024-11-28 14:57:14 cf-templates-kmi81w4ukk76-eu-west-1
2024-11-28 14:57:14 cf-templates-kmi81w4ukk76-eu-west-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
```

5. Use aws s3 mb to create a new S3 bucket in the eu-west-1 region (or use an existing bucket).

```
PS C:\Users\Ishika> aws s3 mb s3://generals-buckets-stacks
make_bucket: generals-buckets-stacks
```

6. Upload the template file to the S3 bucket using aws s3 cp.

```
PS C:\Users\Ishika> aws s3 cp "C:\internship\Section 7\7.2. Creating Stacks With AWS
CLI\database-stack-template.yaml" s3://generals-buckets-stacks
upload: ..\..\internship\Section 7\7.2. Creating Stacks With AWS CLI\database-stack-t
emplate.yaml to s3://generals-buckets-stacks/database-stack-template.yaml
```

7. Use the aws cloudformation create-stack command with the --stack-name option (e.g., DatabaseStack). Provide the template's S3 URL via the --template-url option in the create-stack command. Add the required parameters (e.g., VPC ID and DB subnets) using the --parameters option. Escape any commas in parameter values with double backslashes. Execute the command to create the stack and note the stack ID returned by AWS CLI.

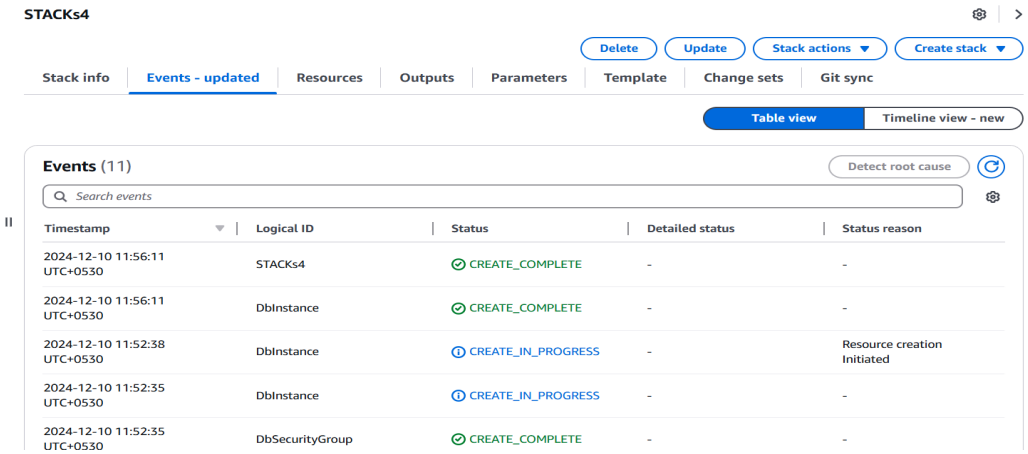
```
PS C:\Users\Ishika> aws cloudformation create-stack --stack-name STACKs4 --template-u
rl http://s3-eu-west-1.amazonaws.com/generals-buckets-stacks/database-stack-template.
yaml --parameters ParameterKey=VpcId,ParameterValue=vpc-0b0c8badfef0024a4 ParameterKe
y=DbSubnets,ParameterValue="subnet-0619479663b4084af\,subnet-08c772de46d053876\,subne
t-0a1bbb2feaa84bb95"
{
    "StackId": "arn:aws:cloudformation:eu-west-1:878893308172:stack/STACKs4/1eea6020-
b6bf-11ef-9485-02230301ef13"
}
```

8. Check the stack's creation status using the aws cloudformation describe-stacks command with the stack name or ID.

```
PS C:\Users\Ishika> aws cloudformation describe-stacks --stack-name arn:aws:cloudform
ation:eu-west-1:878893308172:stack/STACKs4/1eea6020-b6bf-11ef-9485-02230301ef13
{
    "Stacks": [
        {
            "StackId": "arn:aws:cloudformation:eu-west-1:878893308172:stack/STACKs4/1
eea6020-b6bf-11ef-9485-02230301ef13",
            "StackName": "STACKs4",
            "Description": "Sample database stack template for AWS CLI section.  It c
reates an Amazon RDS instance. ",
            "Parameters": [
                {
                    "ParameterKey": "VpcId",
                    "ParameterValue": "vpc-0b0c8badfef0024a4"
                },
                {
                    "ParameterKey": "DbSubnets",
                    "ParameterValue": "subnet-0619479663b4084af,subnet-08c772de46d053
876,subnet-0a1bbb2feaa84bb95"
```

```
PS C:\Users\Ishika> aws cloudformation describe-stacks --stack-name STACKs4
{
    "Stacks": [
        {
            "StackId": "arn:aws:cloudformation:eu-west-1:878893308172:stack/STACKs4/1
eea6020-b6bf-11ef-9485-02230301ef13",
            "StackName": "STACKs4",
            "Description": "Sample database stack template for AWS CLI section.  It c
reates an Amazon RDS instance. ",
            "Parameters": [
                {
                    "ParameterKey": "VpcId",
                    "ParameterValue": "vpc-0b0c8badfef0024a4"
                },
                {
                    "ParameterKey": "DbSubnets",
                    "ParameterValue": "subnet-0619479663b4084af,subnet-08c772de46d053
876,subnet-0a1bbb2feaa84bb95"
                },
```

9. Optionally, use the aws cloudformation wait stack-create-complete command to monitor the stack creation until completion.
10. Verify the stack status as "CREATE_COMPLETE" via CLI or CloudFormation Console.



Updating Stacks with AWS CLI

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. Use the aws cloudformation describe-stack-resources command to list all resources in the stack by providing the stack name.

```
PS C:\Users\Ishika> aws cloudformation describe-stacks --stack-name STACKs4
{
    "Stacks": [
        {
            "StackId": "arn:aws:cloudformation:eu-west-1:878893308172:stack/STACKs4/1
eea6020-b6bf-11ef-9485-02230301ef13",
            "StackName": "STACKs4",
            "Description": "Sample database stack template for AWS CLI section.  It c
reates an Amazon RDS instance. ",
            "Parameters": [
                {
                    "ParameterKey": "VpcId",
                    "ParameterValue": "vpc-0b0c8badfef0024a4"
                },
                {
                    "ParameterKey": "DbSubnets",
                    "ParameterValue": "subnet-0619479663b4084af,subnet-08c772de46d053
876,subnet-0a1bbb2feaa84bb95"
```

3. Filter resources to a specific one using the --logical-resource-id option to describe only that resource.

```
PS C:\Users\Ishika> aws cloudformation describe-stack-resources --stack-name STACKs4
--logical-resource-id DbInstance
{
    "StackResources": [
        {
            "StackName": "STACKs4",
            "StackId": "arn:aws:cloudformation:eu-west-1:878893308172:stack/STACKs4/1
eea6020-b6bf-11ef-9485-02230301ef13",
            "LogicalResourceId": "DbInstance",
            "PhysicalResourceId": "stacks4-dbinstance-bkdpxcxfqde5",
            "ResourceType": "AWS::RDS::DBInstance",
            "Timestamp": "2024-12-10T06:26:11.297000+00:00",
            "ResourceStatus": "CREATE_COMPLETE",
            "DriftInformation": {
                "StackResourceDriftStatus": "NOT_CHECKED"
            }
        }
    ]
}
```

4. Copy the resource's physical ID and describe its details using aws rds describe-db-instances with the --db-instance-identifier option.

```
        "StackId": "arn:aws:cloudformation:eu-west-1:878893308172:stack/STACKs
6020-b6bf-11ef-9485-02230301ef13",
        "LogicalResourceId": "DbInstance",
        "PhysicalResourceId": "stacks4-dbinstance-bkdpxcxfqde5",
        "ResourceType": "AWS::RDS::DBInstance",
        "Timestamp": "2024-12-10T06:26:11.297000+00:00",
```

```
PS C:\Users\Ishika> aws rds describe-db-instances --db-instance-identifier stacks4-db
instance-bkdpxcxfqde5
{
    "DBInstances": [
        {
            "DBInstanceIdentifier": "stacks4-dbinstance-bkdpxcxfqde5",
            "DBInstanceClass": "db.t4g.micro",
            "Engine": "mysql",
            "DBInstanceStatus": "available",
            "MasterUsername": "dbadmin",
            "Endpoint": {
                "Address": "stacks4-dbinstance-bkdpxcxfqde5.c924886w4bji.eu-west-1.rd
s.amazonaws.com",
```

5. Update the stack using aws cloudformation update-stack, providing the --stack-name and a new template via --template-body or --template-url. Adjust parameters during the update using --parameters to specify the new AllocatedStorage value and other required values.

```
PS C:\Users\Ishika> aws s3 cp "C:\internship\Section 7\7.2. updating stacks with AWS
CLI\database-stack-template.yaml" s3://generals-buckets-stacks
upload: ..\..\internship\Section 7\7.2. updating stacks with AWS CLI\database-stack-t
emplate.yaml to s3://generals-buckets-stacks/database-stack-template.yaml
PS C:\Users\Ishika> aws cloudformation update-stack --stack-name STACKs4 --template-u
rl http://s3-eu-west-1.amazonaws.com/generals-buckets-stacks/database-stack-template.
yaml --parameters ParameterKey=AllocatedStorage,ParameterValue=10 ParameterKey=VpcId,
ParameterValue=vpc-0b0c8badfef0024a4 ParameterKey=DbSubnets,ParameterValue="subnet-06
19479663b4084af\,subnet-08c772de46d053876\,subnet-0a1bbb2feaa84bb95"
{
    "StackId": "arn:aws:cloudformation:eu-west-1:878893308172:stack/STACKs4/1eea6020-
b6bf-11ef-9485-02230301ef13"
}
```

6. Check the update progress with aws cloudformation describe-stacks to verify the stack status is UPDATE_IN_PROGRESS.

```
PS C:\Users\Ishika> aws cloudformation describe-stacks --stack-name STACKs4
{
    "Stacks": [
        {
            "StackId": "arn:aws:cloudformation:eu-west-1:878893308172:stack/STACKs4/1
eea6020-b6bf-11ef-9485-02230301ef13",
            "StackName": "STACKs4",
            "Description": "Sample database stack template for AWS CLI section.  It c
reates an Amazon RDS instance. ",
            "Parameters": [
                {
                    "ParameterKey": "VpcId",
                    "ParameterValue": "vpc-0b0c8badfef0024a4"
                },
                {
```

```
    ],
    "CreationTime": "2024-12-10T06:22:24.634000+00:00",
    "LastUpdatedTime": "2024-12-10T07:51:45.768000+00:00",
    "RollbackConfiguration": {},
    "StackStatus": "UPDATE_IN_PROGRESS",
    "DisableRollback": false,
    "NotificationARNs": [],
    "Tags": [],
```

7. Use aws cloudformation wait stack-update-complete to automate polling until the update completes. Verify the stack status again with aws cloudformation describe-stacks to ensure the status is UPDATE_COMPLETE.

```
PS C:\Users\Ishika> aws cloudformation wait stack-update-complete --stack-name STACKs
4
PS C:\Users\Ishika> aws cloudformation describe-stacks --stack-name STACKs4
{
    "Stacks": [
        {
            "StackId": "arn:aws:cloudformation:eu-west-1:878893308172:stack/STACKs4/1
eea6020-b6bf-11ef-9485-02230301ef13",
            "StackName": "STACKs4",
            "Description": "Sample database stack template for AWS CLI section.  It c
reates an Amazon RDS instance. ",
            "Parameters": [
```

```
    ],
    "CreationTime": "2024-12-10T06:22:24.634000+00:00",
    "LastUpdatedTime": "2024-12-10T07:51:45.768000+00:00",
    "RollbackConfiguration": {},
    "StackStatus": "UPDATE_COMPLETE",
    "DisableRollback": false,
    "NotificationARNs": [],
    "Tags": [],
```

8. Use aws cloudformation describe-stack-resources to confirm the resource states and identify updated resources. For a singular resource view, use aws cloudformation describe-stack-resource to see detailed attributes like LastUpdatedTimestamp.

```
PS C:\Users\Ishika> aws cloudformation describe-stack-resources --stack-name STACKs4
{
    "StackResources": [
        {
            "StackName": "STACKs4",
            "StackId": "arn:aws:cloudformation:eu-west-1:878893308172:stack/STACKs4/1
eea6020-b6bf-11ef-9485-02230301ef13",
            "LogicalResourceId": "DbInstance",
            "PhysicalResourceId": "stacks4-dbinstance-bkdpxcxfqde5",
            "ResourceType": "AWS::RDS::DBInstance",
            "Timestamp": "2024-12-10T08:12:33.036000+00:00",
            "ResourceStatus": "UPDATE_COMPLETE",
            "DriftInformation": {
                "StackResourceDriftStatus": "NOT_CHECKED"
            }
        }
```