1. Explain what CloudFormation is, why / when to use it, and some example use cases.

AWS **CloudFormation** is a service that gives us developers an easy way to create a collection of related AWS and third-party resources, and provision and manage them in an orderly and predictable fashion.

I have used to create a ec2 instance roles, policies, lambda functions etc, It can be written in both yaml and json format

Example

AWSTemplateFormatVersion: "2010-09-09"

Description: A sample template

Resources:

MyEC2Instance: *#An inline comment*

Type: "AWS::EC2::Instance"

Properties:

ImageId: "ami-0ff8a91507f77f867" *#Another comment -- This is a Linux AMI*

InstanceType: t2.micro

KeyName: testkey

BlockDeviceMappings:

-

DeviceName: /dev/sdm

Ebs:

VolumeType: io1

Iops: 200

DeleteOnTermination: false

VolumeSize: 20

1. Explain what SNS is and what are some use cases for it.

Amazon Simple Notification Service (Amazon SNS) is a managed service that provides message delivery from publishers to subscribers (also known as producers and consumers). SNS subscribes to topic for sending a message

Some used cases where I have used sns topic is create a notification to or DL group if any alert is raised with an error in production logs.

1. Assume our Lambda is going to be called thousands or even millions of times per second. What scaling challenges would you anticipate? Where are the bottlenecks?

Lambdas can handle the scaling functionality request up to 1000 per region after that any request would be a 504 error, the solution it is access the cross regions access to maintain the scalability.

1. Assume our Lambda becomes more computationally expensive, but some results can be predicted from past events. What steps might you take to make our application more efficient and cheaper to maintain?

A lambda can be easily triggered and can invoke at anytime so a cronjob can used efficiently to turn off the lambda resources immediately after use.

1. What are the tradeoffs involved in using Lambdas (serverless) versus traditional monolithic development?

Traditional monolith deployment involves the maintenance, costing and resources and uptime for a specific task which can be easily achieved by lambdas and they cost less and provide high latency.