QUIZ 1:

Efficiency:

Question 1:

You have designed an application architecture that uses microservices to deploy small, independent components. These components will need to quickly scale based upon application usage. How would you deploy these microservices to ensure process isolation with the leaset operational overhead?

1. Use amazon EC2 micro instances with auto scaling.
2. Deploy tasks to AWS fargate with service auto scaling
3. Launch containers using AWS lambda layers
4. Use AWS elastic beanstalk with pre-defined AMIs

Answer: 2. Deploy tasks to AWS fargate with service auto scaling

Databases:

Question 2:

A library runs a third party app that stores book borrowing information in dynamodb. You have been asked to write a microservice that will notify people when book they reserved has been returned to the library. However, you are not able to modify third party app.

How could this microservice be triggered when a book is returned?

1. Add a dynamodb user defined function that triggers on a book return.

2. Schedule an AWS lambda task to check the database every hour.

3. Use dynamodb streams to trigger an AWS lambda function.

4. Use AWS CloudTrail to trigger an AWS lambda function when the dynamodb log file contains a book return.

Answer: 3. Use dynamodb streams to trigger an AWS lambda function.

Code Deploy:

Question 3:

You have a codeDeploy deployment group with 10 healthy instances. A new deployment is started, with minimum healthy hosts set to 9, however the first instance fails to deploy.

What will codeDeploy do to the deployment?

1. It will continue the deployment one instance at a time.

2. It will immediately fail overall deployment.

3. It will launch replacement instance and attempt re-deployment.

4. It will reboot the failed instance and attempt re-deployment.

Answer: It will immediately fail overall deployment.

Performance:

Question 4:

You have written a web application in AWS elastic beanstalk that removes background from selfie pictures. Users have complained that the website is slow, so you wish to move to an asynchronous model to process pictures.

How can you do this using Elastic Beanstalk?

1. Activate long-polling on the web request while awaiting results.

2. Spawn a local background worker process to process pictures.

3. Create a worker environment utilizing an amazon SQS queue.

4.Configure a periodic task to queue messages on a schedule

Answer: 3. Create a worker environment utilizing an amazon SQS queue.

Amazon ECS:

Question 5:

Your web application is deployed via Amazon ECS containers frontend by load balancer. You wish to move to a new application version by using a blue/green deployment. The blue version is live, and the green version was deployed on a different port for application testing.

How can you now gradually swap the application to green version?

1. Use weighted target groups on the load balancer.

2. Swap the listener rules for the existing target group

3. Trigger an Amazon ECS rolling update

4. Force a new deployment of the amazon ECS cluster

Answer: 1. Use weighted target groups on the load balancer.

QUIZ 2:

CloudFormation:

Question 1:

A CloudFormation stack creates an Amazon S3 bucket. When the stack is deleted, an error occurs because the bucket is not empty.

How can the CloudFormation Stack be modified to delete the contents of bucket when the stack is deleted?

1. Add the ‘purgeBucket’ property to the bucket definition.

2. Add an amazon ECS task to the CloudFormation template that will run during stack deletion

3. Add a custom resource to the CloudFormation template and provide an AWS lambda function

4. Add an AWS CloudTrail event to trigger an AWS lambda function when the stack is deleted.

Answer: 3. Add a custom resource to the CloudFormation template and provide an AWS lambda function

AWS Lambda:

Question 2:

What functionality is provided by AWS lambda layers?

1. Layers enables chaining between lambda functions at multiple functional levels

2. Layers keep deployment packages small by pulling an additional code and content

3. Layers enforce security by defining public frontends and private backends

4. Layers enable reuse of lambda containers for future code execution.

Answer: 2. Layers keep deployment packages small by pulling an additional code and content

Storage:

Question 3:

Your application is deployed across multiple Amazon EC2 instances. You wish to implement a counter variable that can be accessed and updated by all running instances in real-time.

How can this be implemented with minimal effort?

1. Launch an Amazon RDS database and add a counter table. Use an update statement to increment the variable.
2. Launch an elasticCache cluster and add a counter value. Use the incr() operator to increment the variable.
3. Mount a shared Amazon EFS file system and track the value in a file.
4. Use and automic counter on a DynamoDB table.

Answer: 4. Use and automic counter on a DynamoDB table.

Amazon S3:

Question 4:

You are creating a website that sells software online. After a customer pays for a product, they should be able to download the software.

How can this be done in a highly-scaled, secured manner?

1. Generate an Amazon S3 pre-signed with URL with an expiry time.

2. Provide a link to API gateway, which can verify the URLand return the software from Amazon S3 as download

3. Use auto scaling and an elastic load balancer, with a secure web server retrieving and streaming the files from Amazon S3

4. Provide a unique, hard to guess URL on Amazon S3.

Answer: 1. Generate an Amazon S3 pre-signed with URL with an expiry time.

Serverless:

Question 5:

Your application is generating time-sensitive objects in an amazon S3 bucket. These objects must be deleted 30 minutes after creation.

How can this operation be automated?

1. Schedule an AWS lambda function to run every 30 minutes.
2. Use Amazon CloudWatch events to trigger a step function when an object is created
3. Configure amazon S3 object lifecycle with a 30 minute expiry policy.
4. Add an expires metadata tag to objects when they are created.

Answer: 2. Use Amazon CloudWatch events to trigger a step function when an object is created