**AWS Certified Developer Associate - Results**

65 questions | 2 hours 10 minutes | 72% correct required to pass

Attempt 1: Failed (72% required to pass)

67%correct(44/65)

1 hour 17 minutes

**Domains**

**AWS Compute (17 questions)**

71%

29%

**AWS Database (11 questions)**

82%

18%

**AWS Security, Identity, & Compliance (12 questions)**

58%

42%

**AWS Networking & Content Delivery (6 questions)**

33%

67%

**AWS Developer Tools (3 questions)**

100%

**AWS Storage (5 questions)**

60%

40%

**AWS Application Integration (6 questions)**

50%

50%

**AWS Management & Governance (3 questions)**

100%

## AWS Certified Developer Associate - Results

Back to result overview

**New**

[Share feedback](https://survey.alchemer.com/s3/7719142/Practice-tests-learner-csat/?user_id=52182326&course_id=2981704&quiz_id=4852744)

**Attempt 1**

All domains

* 65 all
* 44 correct
* 21 incorrect
* 0 skipped
* 16 marked

Collapse all questions

**Question 1Incorrect**

A Developer is deploying an Amazon EC2 update using AWS CodeDeploy. In the appspec.yml file, which of the following is a valid structure for the order of hooks that should be specified?

**BeforeBlockTraffic > AfterBlockTraffic > BeforeAllowTraffic > AfterAllowTraffic**

**BeforeAllowTraffic > AfterAllowTraffic**

**Your answer is incorrect**

**BeforeInstall > AfterInstall > AfterAllowTestTraffic > BeforeAllowTraffic > AfterAllowTraffic**

**Correct answer**

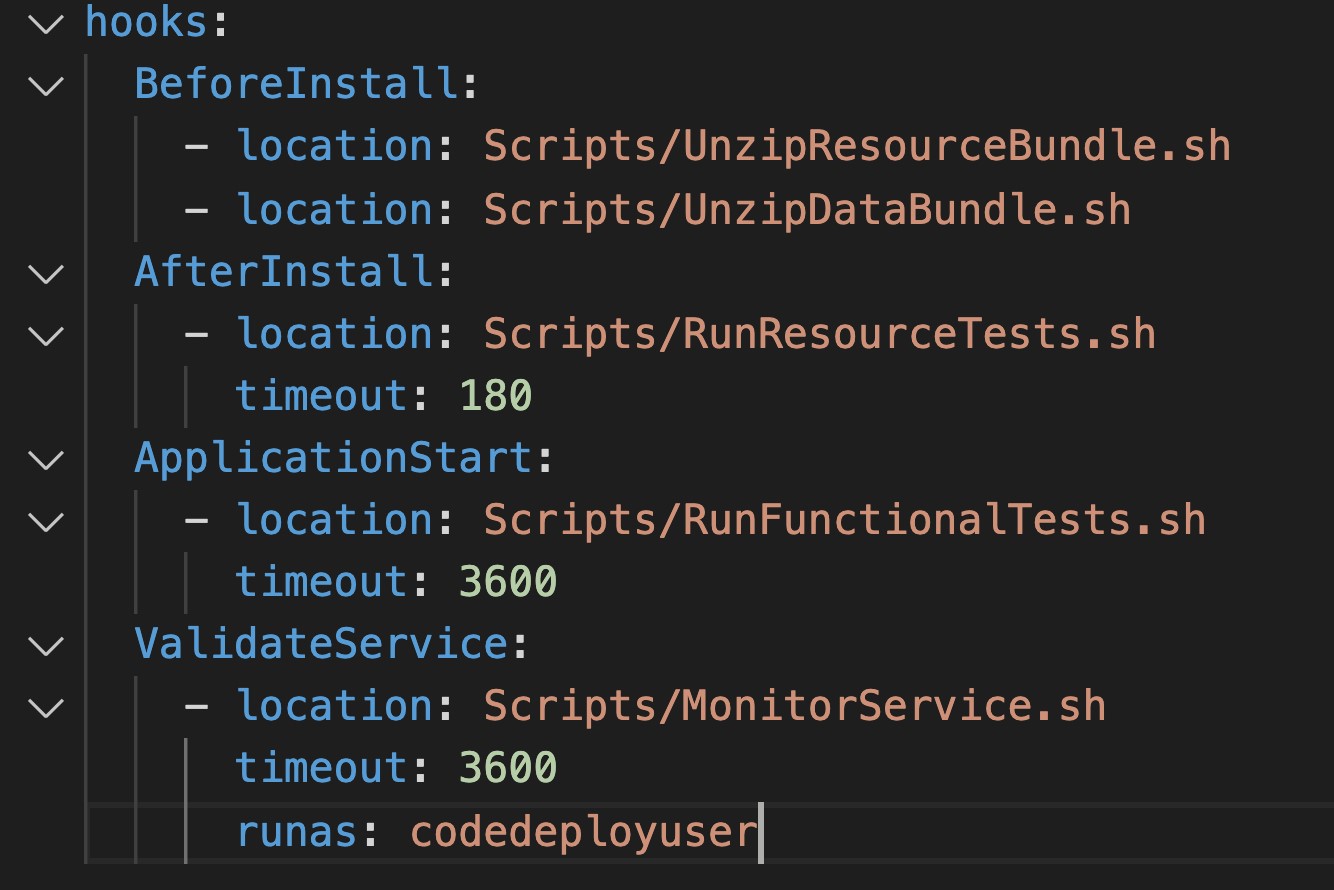
**BeforeInstall > AfterInstall > ApplicationStart > ValidateService**

Overall explanation

The content in the 'hooks' section of the AppSpec file varies, depending on the compute platform for your deployment. The 'hooks' section for an EC2/On-Premises deployment contains mappings that link deployment lifecycle event hooks to one or more scripts.

The 'hooks' section for a Lambda or an Amazon ECS deployment specifies Lambda validation functions to run during a deployment lifecycle event. If an event hook is not present, no operation is executed for that event. This section is required only if you are running scripts or Lambda validation functions as part of the deployment.

The following code snippet shows a valid example of the structure of hooks for an Amazon EC2 deployment:



Therefore, in this scenario a valid structure for the order of hooks that should be specified in the appspec.yml file is: BeforeInstall > AfterInstall > ApplicationStart > ValidateService

**CORRECT:**"BeforeInstall > AfterInstall > ApplicationStart > ValidateService" is the correct answer.

**INCORRECT:** "BeforeInstall > AfterInstall > AfterAllowTestTraffic > BeforeAllowTraffic > AfterAllowTraffic" is incorrect as this would be valid for Amazon ECS.

**INCORRECT:** "BeforeAllowTraffic > AfterAllowTraffic" is incorrect as this would be valid for AWS Lambda.

**INCORRECT:** "BeforeBlockTraffic > AfterBlockTraffic > BeforeAllowTraffic > AfterAllowTraffic" is incorrect as this is a partial listing of hooks for Amazon EC2 but is incomplete.

**References:**

<https://docs.aws.amazon.com/codedeploy/latest/userguide/reference-appspec-file-structure-hooks.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-developer-tools/>

**Domain**

AWS Compute

**Question 2Correct**

A developer is building a Docker application on Amazon ECS that will use an Application Load Balancer (ALB). The developer needs to configure the port mapping between the host port and container port. Where is this setting configured?

**Your answer is correct**

**Task definition**

**Service scheduler**

**Container instance**

**Host definition**

Overall explanation

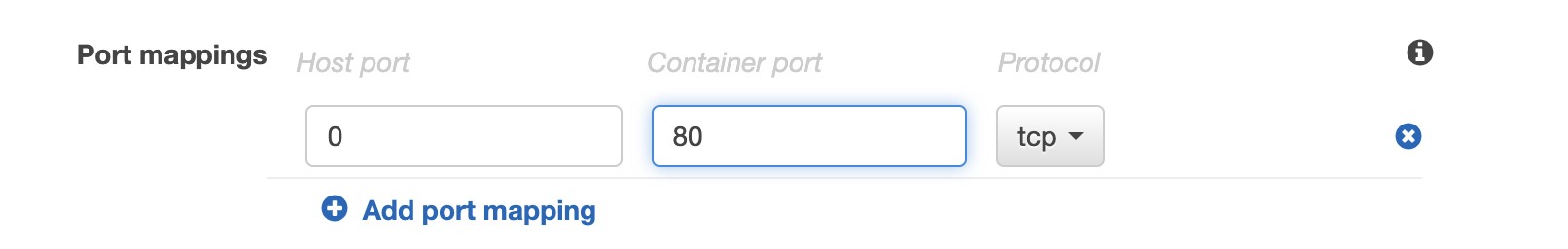
Port mappings allow containers to access ports on the host container instance to send or receive traffic. Port mappings are specified as part of the container definition.

The container definition settings are specified within the task definition. The relevant settings are:

containerPort - the port number on the container that is bound to the user-specified or automatically assigned host port.

hostPort - the port number on the container instance to reserve for your container.

With an ALB you can use Dynamic port mapping which makes it easier to run multiple tasks on the same Amazon ECS service on an Amazon ECS cluster. This is configured by setting the host port to 0, as in the image below:



**CORRECT:**"Task definition" is the correct answer.

**INCORRECT:** "Host definition" is incorrect as there’s no such thing.

**INCORRECT:** "Service scheduler" is incorrect as the service scheduler is responsible for scheduling tasks and placing those tasks.

**INCORRECT:** "Container instance" is incorrect as you don’t specify any settings on the container instance to control the host and container port mappings.

**References:**

<https://docs.aws.amazon.com/AmazonECS/latest/APIReference/API_PortMapping.html>

<https://aws.amazon.com/premiumsupport/knowledge-center/dynamic-port-mapping-ecs/>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-ecs-and-eks/>

**Domain**

AWS Compute

**Question 3Correct**

An Amazon ElastiCache cluster has been placed in front of a large Amazon RDS database. To reduce cost the ElastiCache cluster should only cache items that are actually requested. How should ElastiCache be optimized?

**Use a write-through caching strategy**

**Only cache database writes**

**Your answer is correct**

**Use a lazy loading caching strategy**

**Enable a TTL on cached data**

Overall explanation

There are two caching strategies available: Lazy Loading and Write-Through:

**Lazy Loading**

Loads the data into the cache only when necessary (if a cache miss occurs).

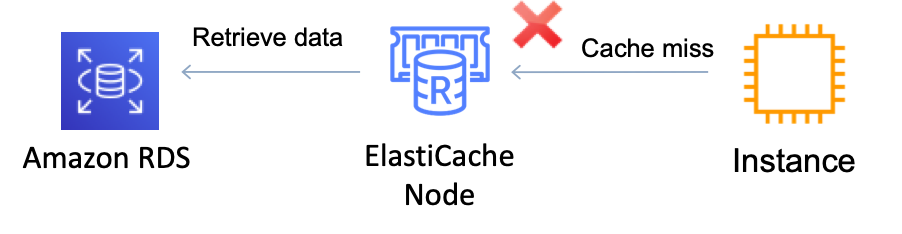
Lazy loading avoids filling up the cache with data that won’t be requested.

If requested data is in the cache, ElastiCache returns the data to the application.

If the data is not in the cache or has expired, ElastiCache returns a null.

The application then fetches the data from the database and writes the data received into the cache so that it is available for next time.

Data in the cache can become stale if Lazy Loading is implemented without other strategies (such as TTL).



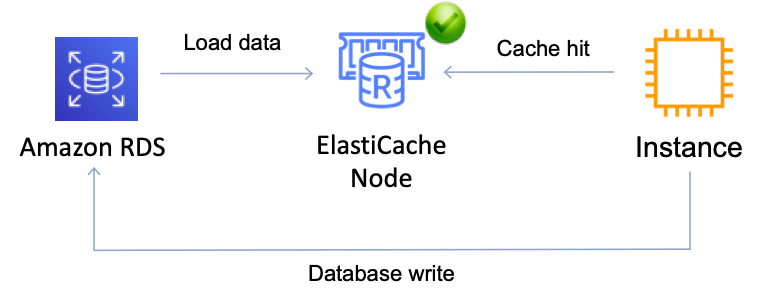
**Write Through**

When using a write through strategy, the cache is updated whenever a new write or update is made to the underlying database.

Allows cache data to remain up-to-date.

Can add wait time to write operations in your application.

Without a TTL you can end up with a lot of cached data that is never read.



**CORRECT:**"Use a lazy loading caching strategy" is the correct answer.

**INCORRECT:** "Use a write-through caching strategy" is incorrect as this will load all database items into the cache increasing cost.

**INCORRECT:** "Only cache database writes" is incorrect as you cannot cache writes, only reads.

**INCORRECT:** "Enable a TTL on cached data" is incorrect. This would help expire stale items but it is not a cache optimization strategy that will cache only items that are requested.

**References:**

<https://docs.aws.amazon.com/AmazonElastiCache/latest/mem-ug/Strategies.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-elasticache/>

**Domain**

AWS Database

**Question 4Incorrect**

A Developer needs to update an Amazon ECS application that was deployed using AWS CodeDeploy. What file does the Developer need to update to push the change through CodeDeploy?

**Correct answer**

**appspec.yml**

**ebextensions.config**

**Your answer is incorrect**

**buildspec.yml**

**dockerrun.aws.json**

Overall explanation

In CodeDeploy, a revision contains a version of the source files CodeDeploy will deploy to your instances or scripts CodeDeploy will run on your instances. You plan the revision, add an AppSpec file to the revision, and then push the revision to Amazon S3 or GitHub. After you push the revision, you can deploy it.

For a deployment to an Amazon ECS compute platform:

• The AppSpec file specifies the Amazon ECS task definition used for the deployment, a container name and port mapping used to route traffic, and optional Lambda functions run after deployment lifecycle events.

• A revision is the same as an AppSpec file.

• An AppSpec file can be written using JSON or YAML.

• An AppSpec file can be saved as a text file or entered directly into a console when you create a deployment.

**Therefore, the appspec.yml file needs to be updated by the Developer.**

**CORRECT:**"appspec.yml" is the correct answer.

**INCORRECT:** "dockerrun.aws.json" is incorrect. A Dockerrun.aws.json file describes how to deploy a remote Docker image as an Elastic Beanstalk application.

**INCORRECT:** "buildspec.yaml" is incorrect. A build spec is a collection of build commands and related settings, in YAML format, that CodeBuild uses to run a build using AWS CodeBuild.

**INCORRECT:** "ebextensions.config" is incorrect. The .ebextensions folder in the source code for an Elastic Beanstalk application is used for .config files that configure the environment and customize resources.

**References:**

<https://docs.aws.amazon.com/codedeploy/latest/userguide/application-revisions-appspec-file.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-developer-tools/>

**Domain**

AWS Compute

**Question 5Incorrect**

An application resizes images that are uploaded to an Amazon S3 bucket. Amazon S3 event notifications are used to trigger an AWS Lambda function that resizes the images. The processing time for each image is less than one second. A large amount of images are expected to be received in a short burst of traffic. How will AWS Lambda accommodate the workload?

**Your answer is incorrect**

**Lambda will collect and then batch process the images in a single execution**

**Correct answer**

**Lambda will scale out and execute the requests concurrently**

**Lambda will scale the memory allocated to the function to increase the amount of CPU available to process many images**

**Lambda will process the images sequentially in the order they are received**

Overall explanation

The first time you invoke your function, AWS Lambda creates an instance of the function and runs its handler method to process the event. When the function returns a response, it stays active and waits to process additional events. If you invoke the function again while the first event is being processed, Lambda initializes another instance, and the function processes the two events concurrently.

Your functions’ concurrency is the number of instances that serve requests at a given time. For an initial burst of traffic, your functions’ cumulative concurrency in a Region can reach an initial level of between 500 and 3000, which varies per Region.

Burst Concurrency Limits:

• 3000 – US West (Oregon), US East (N. Virginia), Europe (Ireland).

• 1000 – Asia Pacific (Tokyo), Europe (Frankfurt).

• 500 – Other Regions.

After the initial burst, your functions’ concurrency can scale by an additional 500 instances each minute. This continues until there are enough instances to serve all requests, or until a concurrency limit is reached.

The default account limit is up to 1000 executions per second, per region (can be increased).

**CORRECT:**"Lambda will scale out and execute the requests concurrently" is the correct answer.

**INCORRECT:** "Lambda will process the images sequentially in the order they are received" is incorrect as Lambda uses concurrency to process multiple events in parallel.

**INCORRECT:** "Lambda will collect and then batch process the images in a single execution" is incorrect as Lambda never collects requests and then processes them at a later time. Lambda always uses concurrency to process requests in parallel.

**INCORRECT:** "Lambda will scale the memory allocated to the function to increase the amount of CPU available to process many images" is incorrect as Lambda does not automatically scale memory/CPU and processes requests in parallel, not sequentially.

**References:**

<https://docs.aws.amazon.com/lambda/latest/dg/configuration-concurrency.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-lambda/>

**Domain**

AWS Storage

**Question 6Correct**

An application will ingest data at a very high throughput from several sources and stored in an Amazon S3 bucket for subsequent analysis. Which AWS service should a Developer choose for this requirement?

**Amazon S3 Transfer Acceleration**

**Your answer is correct**

**Amazon Kinesis Data Firehose**

**Amazon Kinesis Data Analytics**

**Amazon Simple Queue Service (SQS)**

Overall explanation

Amazon Kinesis Data Firehose is the easiest way to reliably load streaming data into data lakes, data stores and analytics tools. It can capture, transform, and load streaming data into Amazon S3, Amazon Redshift, Amazon Elasticsearch Service, and Splunk, enabling near real-time analytics with existing business intelligence tools and dashboards.

A destination is the data store where your data will be delivered. Firehose Destinations include:

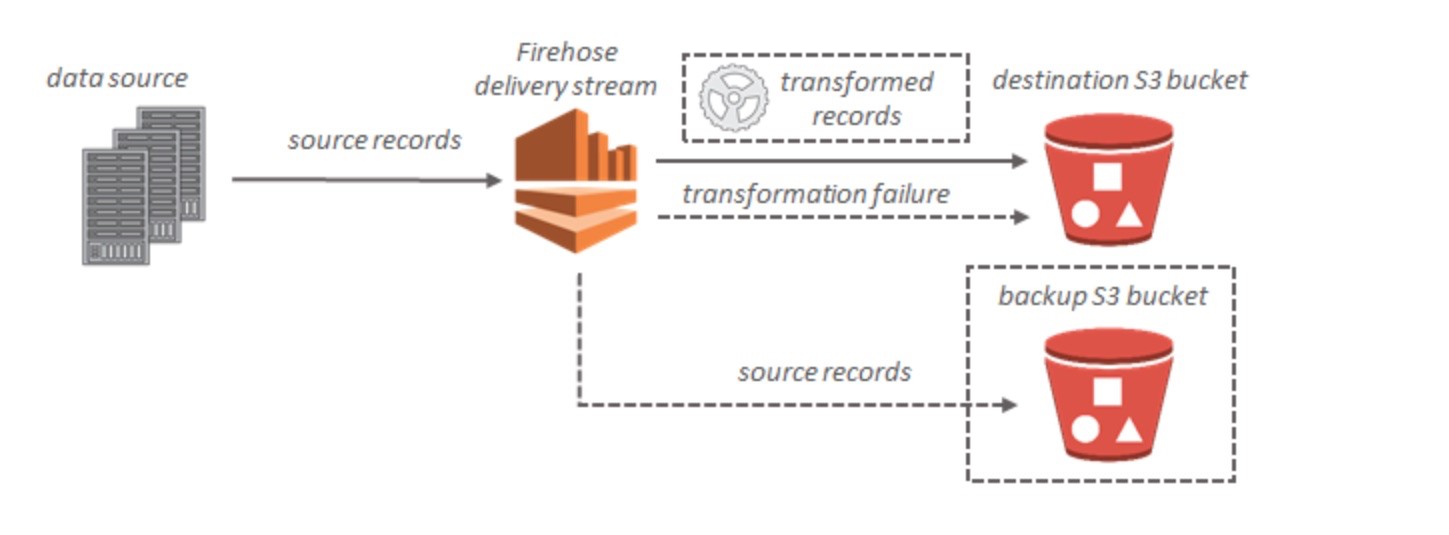
• Amazon S3.

• Amazon Redshift.

• Amazon Elasticsearch Service.

Splunk.

For Amazon S3 destinations, streaming data is delivered to your S3 bucket. If data transformation is enabled, you can optionally back up source data to another Amazon S3 bucket:



The best choice of AWS service for this scenario is to use Amazon Kinesis Data Firehose as it can ingest large amounts of data at extremely high throughput and load that data into an Amazon S3 bucket

**CORRECT:**"Amazon Kinesis Data Firehose" is the correct answer.

**INCORRECT:** "Amazon S3 Transfer Acceleration" is incorrect as this is a service used for improving the performance of uploads into Amazon S3. It is not suitable for ingesting streaming data.

**INCORRECT:** "Amazon Kinesis Data Analytics" is incorrect as this service is used for processing and analyzing real-time, streaming data. The easiest way to load streaming data into a data store for analysing at a later time is Kinesis Data Firehose

**INCORRECT:** "Amazon Simple Queue Service (SQS)" is incorrect as this is not the best solution for this scenario. With SQS you need a producer to place the messages on the queue and then consumers to process the messages and load them into Amazon S3. Kinesis Data Firehose can do this natively without the need for consumers.

**References:**

<https://aws.amazon.com/kinesis/data-firehose/>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-kinesis/>

**Domain**

AWS Application Integration

**Question 7Correct**

A company is building an application to track athlete performance using an Amazon DynamoDB table. Each item in the table is identified by a partition key (user\_id) and a sort key (sport\_name). The table design is shown below:

• Partition key: user\_id

• Sort Key: sport\_name

• Attributes: score, score\_datetime

A Developer is asked to write a leaderboard application to display the top performers (user\_id) based on the score for each sport\_name.

What process will allow the Developer to extract results MOST efficiently from the DynamoDB table?

**Use a DynamoDB query operation with the key attributes of user\_id and sport\_name and order the results based on the score attribute**

**Create a local secondary index with a primary key of sport\_name and a sort key of score and get the results based on the score attribute**

**Use a DynamoDB scan operation to retrieve scores and user\_id based on sport\_name, and order the results based on the score attribute**

**Your answer is correct**

**Create a global secondary index with a partition key of sport\_name and a sort key of score, and get the results**

Overall explanation

The Developer needs to be able to sort the scores for each sport and then extract the highest performing athletes. In this case BOTH the partition key and sort key must be different which means a Global Secondary index is required (as a Local Secondary index only has a different sort key). The GSI would be configured as follows:

· Partition key: sport\_name

· Sort Key: score

The results will then be listed in order of the highest score for each sport which is exactly what is required.

**CORRECT:**"Create a global secondary index with a partition key of sport\_name and a sort key of score, and get the results" is the correct answer.

**INCORRECT:** "Create a local secondary index with a primary key of sport\_name and a sort key of score and get the results based on the score attribute" is incorrect as an LSI cannot be created after table creation and also only has a different sort key, not a different partition key.

**INCORRECT:** "Use a DynamoDB query operation with the key attributes of user\_id and sport\_name and order the results based on the score attribute" is incorrect as this is less efficient compared to using a GSI.

**INCORRECT:** "Use a DynamoDB scan operation to retrieve scores and user\_id based on sport\_name, and order the results based on the score attribute" is incorrect as this is the least efficient option as a scan returns every item in the table (more RCUs).

**References:**

<https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/GSI.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-dynamodb/>

**Domain**

AWS Database

**Question 8Incorrect**

An application collects data from sensors in a manufacturing facility. The data is stored in an Amazon SQS Standard queue by an AWS Lambda function and an Amazon EC2 instance processes the data and stores it in an Amazon RedShift data warehouse. A fault in the sensors’ software is causing occasional duplicate messages to be sent. Timestamps on the duplicate messages show they are generated within a few seconds of the primary message.

How can a Developer prevent duplicate data being stored in the data warehouse?

**Your answer is incorrect**

**Send a ChangeMessageVisibility call with VisibilityTimeout set to 30 seconds after the receipt of every message from the queue**

**Use a FIFO queue and configure the Lambda function to add a message group ID to the messages generated by each individual sensor**

**Correct answer**

**Use a FIFO queue and configure the Lambda function to add a message deduplication token to the message body**

**Configure a redrive policy, specify a destination Dead-Letter queue, and set the maxReceiveCount to 1**

Overall explanation

FIFO (First-In-First-Out) queues are designed to enhance messaging between applications when the order of operations and events is critical, or where duplicates can't be tolerated.

In FIFO queues, messages are ordered based on message group ID. If multiple hosts (or different threads on the same host) send messages with the same message group ID to a FIFO queue, Amazon SQS stores the messages in the order in which they arrive for processing. To ensure that Amazon SQS preserves the order in which messages are sent and received, ensure that each producer uses a unique message group ID to send all its messages.

FIFO queue logic applies only per message group ID. Each message group ID represents a distinct ordered message group within an Amazon SQS queue. For each message group ID, all messages are sent and received in strict order. However, messages with different message group ID values might be sent and received out of order. You must associate a message group ID with a message. If you don't provide a message group ID, the action fails. If you require a single group of ordered messages, provide the same message group ID for messages sent to the FIFO queue.

Therefore, the Developer can use a FIFO queue and configure the Lambda function to add a message deduplication token to the message body. This will ensure that the messages are deduplicated before being picked up for processing by the Amazon EC2 instance.

**CORRECT:**"Use a FIFO queue and configure the Lambda function to add a message deduplication token to the message body" is the correct answer.

**INCORRECT:** "Use a FIFO queue and configure the Lambda function to add a message group ID to the messages generated by each individual sensor" is incorrect. The message group ID is the tag that specifies that a message belongs to a specific message group. Messages that belong to the same message group are always processed one by one, in a strict order relative to the message group.

**INCORRECT:** "Send a ChangeMessageVisibility call with VisibilityTimeout set to 30 seconds after the receipt of every message from the queue" is incorrect as this will just change the visibility timeout for the message which will prevent others from seeing it until it has been processed and deleted from the queue. This doesn’t stop a message with duplicate data being processed.

**INCORRECT:** "Configure a redrive policy, specify a destination Dead-Letter queue, and set the maxReceiveCount to 1" is incorrect as without a FIFO queue and a message deduplication ID duplicate messages will still enter the queue. The redrive policy only applies to individual messages for which processing has failed a number of times as specified in the maxReceiveCount.

**References:**

<https://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSDeveloperGuide/using-messagededuplicationid-property.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-application-integration-services/>

**Domain**

AWS Application Integration

**Question 9Incorrect**

An organization has encrypted a large quantity of data. To protect their data encryption keys they are planning to use envelope encryption. Which of the following processes is a correct implementation of envelope encryption?

**Encrypt plaintext data with a master key and then encrypt the master key with a top-level plaintext data key**

**Encrypt plaintext data with a master key and then encrypt the master key with a top-level encrypted data key**

**Your answer is incorrect**

**Encrypt plaintext data with a data key and then encrypt the data key with a top-level encrypted master key**

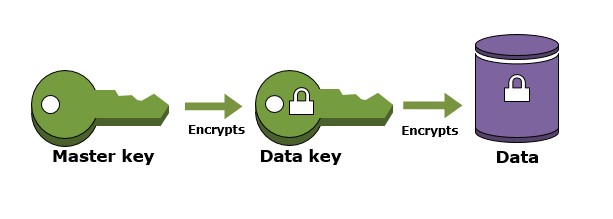
**Correct answer**

**Encrypt plaintext data with a data key and then encrypt the data key with a top-level plaintext master key.**

Overall explanation

When you encrypt your data, your data is protected, but you have to protect your encryption key. One strategy is to encrypt it. Envelope encryption is the practice of encrypting plaintext data with a data key, and then encrypting the data key under another key.

You can even encrypt the data encryption key under another encryption key and encrypt that encryption key under another encryption key. But, eventually, one key must remain in plaintext so you can decrypt the keys and your data. This top-level plaintext key encryption key is known as the master key.



Envelope encryption offers several benefits:

• **Protecting data keys**

When you encrypt a data key, you don't have to worry about storing the encrypted data key, because the data key is inherently protected by encryption. You can safely store the encrypted data key alongside the encrypted data.

• **Encrypting the same data under multiple master keys**

Encryption operations can be time consuming, particularly when the data being encrypted are large objects. Instead of re-encrypting raw data multiple times with different keys, you can re-encrypt only the data keys that protect the raw data.

• **Combining the strengths of multiple algorithms**

In general, symmetric key algorithms are faster and produce smaller ciphertexts than public key algorithms. But public key algorithms provide inherent separation of roles and easier key management. Envelope encryption lets you combine the strengths of each strategy.

As described above, the process that should be implemented is to encrypt plaintext data with a data key and then encrypt the data key with a top-level plaintext master key.

**CORRECT:**"Encrypt plaintext data with a data key and then encrypt the data key with a top-level plaintext master key" is the correct answer.

**INCORRECT:** "Encrypt plaintext data with a master key and then encrypt the master key with a top-level plaintext data key" is incorrect as the master key is the top-level key.

**INCORRECT:** "Encrypt plaintext data with a data key and then encrypt the data key with a top-level encrypted master key" is incorrect as the top-level master key must be unencrypted so it can be used to decrypt data.

**INCORRECT:** "Encrypt plaintext data with a master key and then encrypt the master key with a top-level encrypted data key" is incorrect as the master key is the top-level key.

**References:**

<https://docs.aws.amazon.com/kms/latest/developerguide/concepts.html#enveloping>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-kms/>

**Domain**

AWS Security, Identity, & Compliance

**Question 10Correct**

An application includes multiple Auto Scaling groups of Amazon EC2 instances. Each group corresponds to a different subdomain of example.com, including forum.example.com and myaccount.example.com. An Elastic Load Balancer will be used to distribute load from a single HTTPS listener.

Which type of Elastic Load Balancer MUST a Developer use in this scenario?

**Classic Load Balancer**

**Network Load Balancer**

**Your answer is correct**

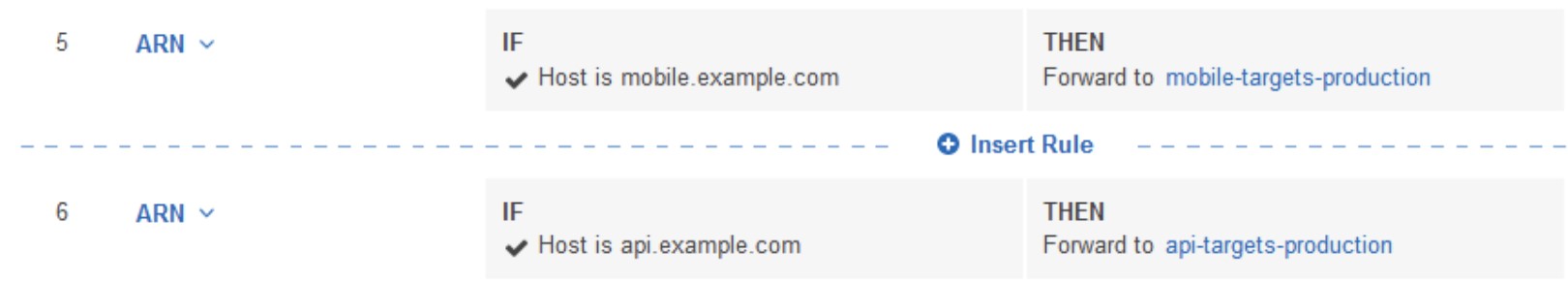
**Application Load Balancer**

**Task Load Balancer**

Overall explanation

With an Application Load Balancer it is possible to route requests based on the domain name specified in the Host header. This means you can route traffic coming in to forum.example.com and myaccount.example.com to different target groups.

You can see an example of a couple of similar rules depicted below:



The Application Load Balancer is the only Elastic Load Balancer provided by AWS that can perform host-based routing.

**CORRECT:**"Application Load Balancer" is the correct answer.

**INCORRECT:** "Network Load Balancer" is incorrect as this type of ELB routes traffic based on information at the connection layer (L4).

**INCORRECT:** "Classic Load Balancer" is incorrect as it doesn't support any kind of host or path-based routing or even target groups.

**INCORRECT:** "Task Load Balancer" is incorrect as this is not a type of ELB.

**References:**

<https://aws.amazon.com/blogs/aws/new-host-based-routing-support-for-aws-application-load-balancers/>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-elastic-load-balancing-aws-elb/>

**Domain**

AWS Compute

**Question 11Correct**

A company has a website that is developed in PHP and WordPress and is launched using AWS Elastic Beanstalk. There is a new version of the website that needs to be deployed in the Elastic Beanstalk environment. The company cannot tolerate having the website offline if an update fails. Deployments must have minimal impact and rollback as soon as possible.

What deployment method should be used?

**Snapshots**

**Your answer is correct**

**Immutable**

**Rolling**

**All at once**

Overall explanation

AWS Elastic Beanstalk provides several options for how deployments are processed, including deployment policies and options that let you configure batch size and health check behavior during deployments.

All at once:

• Deploys the new version to all instances simultaneously.

Rolling:

• Update a few instances at a time (bucket), and then move onto the next bucket once the first bucket is healthy (downtime for 1 bucket at a time).

Rolling with additional batch:

• Like Rolling but launches new instances in a batch ensuring that there is full availability.

Immutable:

• Launches new instances in a new ASG and deploys the version update to these instances before swapping traffic to these instances once healthy.

• Zero downtime.

Blue / Green deployment:

• Zero downtime and release facility.

• Create a new “stage” environment and deploy updates there.

For this scenario, the best choice is Immutable as this is the safest option when you cannot tolerate downtime and also provides a simple way of rolling back should an issue occur.

**CORRECT:**"Immutable" is the correct answer.

**INCORRECT:** "All at once" is incorrect as this will take all instances down and cause a total outage.

**INCORRECT:** "Snapshots" is incorrect as this is not a deployment method you can use with Elastic Beanstalk.

**INCORRECT:** "Rolling" is incorrect as this will reduce the capacity of the application and it is more difficult to roll back as you must redeploy the old version to the instances.

**References:**

<https://docs.aws.amazon.com/elasticbeanstalk/latest/dg/using-features.rolling-version-deploy.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-elastic-beanstalk/>

**Domain**

AWS Compute

**Question 12Correct**

A developer is creating a multi-tier web application. The front-end will place messages in an Amazon SQS queue for the back-end to process. Each job includes a file that is 1GB in size. What MUST the developer do to ensure this works as expected?

**Store the large files in DynamoDB and use the SQS Extended Client Library for Java to manage SQS messages**

**Create a FIFO queue that supports large files**

**Your answer is correct**

**Store the large files in Amazon S3 and use the SQS Extended Client Library for Java to manage SQS messages**

**Increase the maximum message size of the queue from 256KB to 1GB**

Overall explanation

You can use Amazon S3 and the Amazon SQS Extended Client Library for Java to manage Amazon SQS messages. This is especially useful for storing and consuming messages up to 2 GB in size. Unless your application requires repeatedly creating queues and leaving them inactive or storing large amounts of data in your queue, consider using Amazon S3 for storing your data.

You can use the Amazon SQS Extended Client Library for Java library to do the following:

• Specify whether messages are always stored in Amazon S3 or only when the size of a message exceeds 256 KB.

• Send a message that references a single message object stored in an Amazon S3 bucket.

• Get the corresponding message object from an Amazon S3 bucket.

• Delete the corresponding message object from an Amazon S3 bucket.

Note: Amazon SQS only supports messages up to 256KB in size. Therefore, the extended client library for Java must be used.

**CORRECT:**"Store the large files in Amazon S3 and use the SQS Extended Client Library for Java to manage SQS messages" is the correct answer.

**INCORRECT:** "Increase the maximum message size of the queue from 256KB to 1GB" is incorrect as you cannot increase the maximum message size above 256KB.

**INCORRECT:** "Store the large files in DynamoDB and use the SQS Extended Client Library for Java to manage SQS messages" is incorrect as you should store the files in Amazon S3.

**INCORRECT:** "Create a FIFO queue that supports large files " is incorrect as FIFO queues also have a maximum message size of 256KB.

**References:**

<https://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSDeveloperGuide/sqs-s3-messages.html>

<https://docs.aws.amazon.com/AWSSimpleQueueService/latest/SQSDeveloperGuide/working-java-example-using-s3-for-large-sqs-messages.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-application-integration-services/>

**Question 13Correct**

A company runs many microservices applications that use Docker containers. The company are planning to migrate the containers to Amazon ECS. The workloads are highly variable and therefore the company prefers to be charged per running task.

Which solution is the BEST fit for the company’s requirements?

**An Amazon ECS Service with Auto Scaling**

**An Amazon ECS Cluster with Auto Scaling**

**Your answer is correct**

**Amazon ECS with the Fargate launch type**

**Amazon ECS with the EC2 launch type**

Overall explanation

The key requirement is that the company should be charged per running task. Therefore, the best answer is to use Amazon ECS with the Fargate launch type as with this model AWS charge you for running tasks rather than running container instances.

The Fargate launch type allows you to run your containerized applications without the need to provision and manage the backend infrastructure. You just register your task definition and Fargate launches the container for you. The Fargate Launch Type is a serverless infrastructure managed by AWS.

**CORRECT:**"Amazon ECS with the Fargate launch type" is the correct answer.

**INCORRECT:** "Amazon ECS with the EC2 launch type" is incorrect as with this launch type you pay for running container instances (EC2 instances).

**INCORRECT:** "An Amazon ECS Service with Auto Scaling" is incorrect as this does not specify the launch type. You can run an ECS Service on the Fargate or EC2 launch types.

**INCORRECT:** "An Amazon ECS Cluster with Auto Scaling" is incorrect as this does not specify the launch type. You can run an ECS Cluster on the Fargate or EC2 launch types.

**References:**

<https://docs.aws.amazon.com/AmazonECS/latest/developerguide/AWS_Fargate.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-ecs-and-eks/>

**Domain**

AWS Compute

**Question 14Incorrect**

A solution requires a serverless service for receiving streaming data and loading it directly into an Amazon Elasticsearch datastore. Which AWS service would be suitable for this requirement?

**Correct answer**

**Amazon Kinesis Data Firehose**

**Amazon Kinesis Data Analytics**

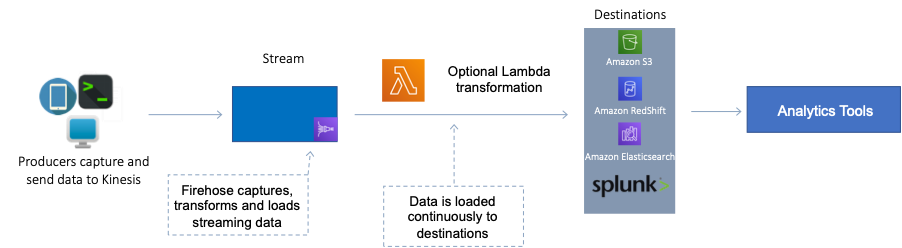
**Amazon Simple Queue Service (SQS)**

**Your answer is incorrect**

**Amazon Kinesis Data Streams**

Overall explanation

Amazon Kinesis Data Firehose is the easiest way to load streaming data into data stores and analytics tools. It can capture, transform, and load streaming data into Amazon S3, Amazon Redshift, Amazon Elasticsearch Service, and Splunk, enabling near real-time analytics with existing business intelligence tools and dashboards you’re already using today.



Firehose is a fully managed service that automatically scales to match the throughput of your data and requires no ongoing administration. It can also batch, compress, and encrypt the data before loading it, minimizing the amount of storage used at the destination and increasing security.

**CORRECT:**"Amazon Kinesis Data Firehose" is the correct answer.

**INCORRECT:** "Amazon Kinesis Data Streams" is incorrect as with Kinesis Data Streams you need consumers running on EC2 instances or AWS Lambda for processing the data from the stream. It therefore will not load data directly to a datastore.

**INCORRECT:** "Amazon Kinesis Data Analytics" is incorrect as this service is used for performing analytics on streaming data using Structured Query Language (SQL queries.

**INCORRECT:** "Amazon Simple Queue Service (SQS)" is incorrect as this is a message queueing service. You would need servers to place messages on the queue and then other servers to process messages from the queue and store them in Elasticsearch.

**References:**

<https://aws.amazon.com/kinesis/data-firehose/faqs/>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-kinesis/>

**Domain**

AWS Application Integration

**Question 15Incorrect**

A Developer is creating multiple AWS Lambda functions that will be using an external library that is not included in the standard Lambda libraries. What is the BEST way to make these libraries available to the functions?

**Store the files in Amazon S3 and reference them from your function code**

**Correct answer**

**Create a layer in Lambda that includes the external library**

**Include the external library with the function code**

**Your answer is incorrect**

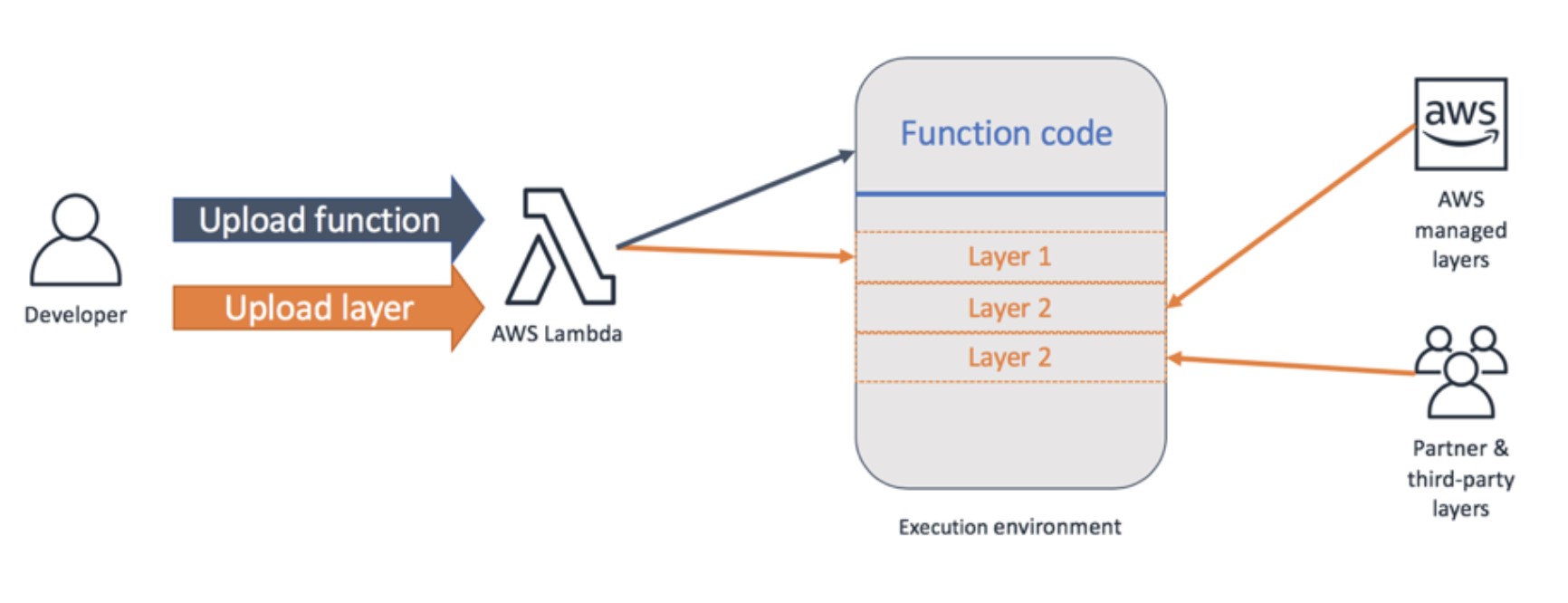
**Create a deployment package that includes the external library**

Overall explanation

You can configure your Lambda function to pull in additional code and content in the form of layers. A layer is a ZIP archive that contains libraries, a custom runtime, or other dependencies. With layers, you can use libraries in your function without needing to include them in your deployment package.

Layers let you keep your deployment package small, which makes development easier. You can avoid errors that can occur when you install and package dependencies with your function code.

When a Lambda function configured with a Lambda layer is executed, AWS downloads any specified layers and extracts them to the /opt directory on the function execution environment. Each runtime then looks for a language-specific folder under the /opt directory.



One of the best practices for AWS Lambda functions is to minimize your deployment package size to its runtime necessities in order to reduce the amount of time that it takes for your deployment package to be downloaded and unpacked ahead of invocation.

Therefore, it is preferable to use layers to store the external libraries to optimize performance of the function. Using layers means that the external library will also be available to all of the Lambda functions that the Developer is creating.

**CORRECT:**"Create a layer in Lambda that includes the external library" is the correct answer.

**INCORRECT:** "Include the external library with the function code" is incorrect as you should not include an external library within the function code. Even if possible this would result in bloated code that could slow down execution time.

**INCORRECT:** "Create a deployment package that includes the external library" is incorrect as the best practice is to minimize package sizes to runtime necessities. Also, this would require including the library in all function deployment packages whereas with layers we can create a single layer that is used by all functions.

**INCORRECT:** "Store the files in Amazon S3 and reference them from your function code" is incorrect as this would likely result in increased latency of your function execution. Instead you should either package the library in the deployment package for your function or use layers (preferable in this scenario).

**References:**

<https://docs.aws.amazon.com/lambda/latest/dg/configuration-layers.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-lambda/>

**Domain**

AWS Compute

**Question 16Incorrect**

A manufacturing company is creating a new RESTful API that their customers can use to query the status of orders. The endpoint for customer queries will be https://www.manufacturerdomain.com/status/customerID

Which of the following application designs will meet the requirements? (Select TWO.)

**Correct selection**

**Elastic Load Balancing; Amazon EC2**

**Your selection is correct**

**Amazon API Gateway; AWS Lambda**

**Amazon SQS; Amazon SNS**

**Amazon S3; Amazon CloudFront**

**Your selection is incorrect**

**Amazon ElastiCache; Amazon Elacticsearch Service**

Overall explanation

This scenario includes a web application that will use RESTful API calls to determine the status of orders and dynamically return the results back to the company’s customers. Therefore, the two best options are as per below:

• Amazon API Gateway; AWS Lambda – this choice includes API Gateway which is provides managed REST APIs and Lambda which can run the backend code for the application. This is a good solution for this scenario.

• Elastic Load Balancing; Amazon EC2 – with this choice the ELB can load balance to one or more EC2 instances which can run the RESTful APIs and compute functions. This is also a good choice but could be more costly (operationally and financially).

None of the other options provide a workable solution for this scenario.

**CORRECT:**"Elastic Load Balancing; Amazon EC2" is a correct answer.

**CORRECT:**"Amazon API Gateway; AWS Lambda" is a correct answer.

**INCORRECT:** "Amazon SQS; Amazon SNS" is incorrect as these services are used for queuing and sending notifications. They are not suitable for hosting a REST API.

**INCORRECT:** "Amazon ElastiCache; Amazon Elacticsearch Service" is incorrect as ElastiCache is an in-memory caching service and Elasticsearch is used for searching. These do not provide a suitable solution for this scenario.

**INCORRECT:** "Amazon S3; Amazon CloudFront" is incorrect as though you can host a static website on Amazon S3 with CloudFront caching the content, this is a static website only and you cannot host an API.

**References:**

<https://aws.amazon.com/ec2/features/>

<https://aws.amazon.com/elasticloadbalancing/>

<https://aws.amazon.com/api-gateway/features/>

<https://aws.amazon.com/lambda/features/>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-ec2/>

<https://digitalcloud.training/aws-elastic-load-balancing-aws-elb/>

<https://digitalcloud.training/aws-lambda/>

<https://digitalcloud.training/amazon-api-gateway/>

**Domain**

AWS Networking & Content Delivery

**Question 17Correct**

A mobile application runs as a serverless application on AWS. A Developer needs to create a push notification feature that sends periodic message to subscribers. How can the Developer send the notification from the application?

**Your answer is correct**

**Publish a notification to an Amazon SNS Topic**

**Publish a message to an Amazon SQS Queue**

**Publish a notification to Amazon CloudWatch Events**

**Publish a message to an Amazon SWF Workflow**

Overall explanation

With [Amazon SNS](https://aws.amazon.com/sns/), you have the ability to send push notification messages directly to apps on mobile devices. Push notification messages sent to a mobile endpoint can appear in the mobile app as message alerts, badge updates, or even sound alerts.

You send push notification messages to both mobile devices and desktops using one of the following supported push notification services:

• Amazon Device Messaging (ADM)

• Apple Push Notification Service (APNs) for both iOS and Mac OS X

• Baidu Cloud Push (Baidu)

• Firebase Cloud Messaging (FCM)

• Microsoft Push Notification Service for Windows Phone (MPNS)

• Windows Push Notification Services (WNS)

To send a notification to an Amazon SNS subscriber, the application needs to send the notification to an Amazon SNS Topic. Amazon SNS will then send the notification to the relevant subscribers.

**CORRECT:**"Publish a notification to an Amazon SNS Topic" is the correct answer.

**INCORRECT:** "Publish a message to an Amazon SQS Queue" is incorrect as SQS is a message queue service, not a notification service.

**INCORRECT:** "Publish a notification to Amazon CloudWatch Events" is incorrect as CloudWatch Events will not be able to send notifications to mobile app users.

**INCORRECT:** "Publish a message to an Amazon SWF Workflow" is incorrect as SWF is a workflow orchestration service and it is not used for publishing messages to mobile app users.

**References:**

<https://docs.aws.amazon.com/sns/latest/dg/sns-how-user-notifications-work.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-application-integration-services/>

**Domain**

AWS Application Integration

**Question 18Incorrect**

A Developer is creating an application that uses Amazon EC2 instances and must be highly available and fault tolerant. How should the Developer configure the VPC?

**Create an Internet Gateway for every availability zone**

**Your answer is incorrect**

**Create a cluster placement group for the EC2 instances**

**Correct answer**

**Create a subnet in each availability zone in the region**

**Create multiple subnets within a single availability zone in the region**

Overall explanation

To ensure high availability and fault tolerance the Developer should create a subnet within each availability zone. The EC2 instances should then be distributed between these subnets.

The Developer would likely use Amazon EC2 Auto Scaling which will automatically launch instances in each subnet and then Elastic Load Balancing to distributed incoming traffic.

**CORRECT:**"Create a subnet in each availability zone in the region" is the correct answer.

**INCORRECT:** "Create multiple subnets within a single availability zone in the region" is incorrect as this will not provide fault tolerance in the event that the AZ becomes unavailable.

**INCORRECT:** "Create an Internet Gateway for every availability zone" is incorrect as there is a single Internet Gateway per VPC.

**INCORRECT:** "Create a cluster placement group for the EC2 instances" is incorrect as this is used for ensuring low latency access between EC2 instances in a single availability zone.

**References:**

<https://d1.awsstatic.com/whitepapers/aws-building-fault-tolerant-applications.pdf>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-vpc/>

**Domain**

AWS Networking & Content Delivery

**Question 19Correct**

A serverless application uses an AWS Lambda function, Amazon API Gateway API and an Amazon DynamoDB table. The Lambda function executes 10 times per second and takes 3 seconds to complete each execution.

How many concurrent executions will the Lambda function require?

**Your answer is correct**

**30**

**3**

**10**

**12**

Overall explanation

Concurrency is the number of requests that your function is serving at any given time. When your function is invoked, Lambda allocates an instance of it to process the event. When the function code finishes running, it can handle another request. If the function is invoked again while a request is still being processed, another instance is allocated, which increases the function's concurrency.

To calculate the concurrency requirements for the Lambda function simply multiply the number of executions per second (10) by the time it takes to complete the execution (3).

Therefore, for this scenario the calculation is 10 x 3 = 30.

**CORRECT:**"30" is the correct answer.

**INCORRECT:** "10" is incorrect. Please use the formula above to calculate concurrency requirements.

**INCORRECT:** "12" is incorrect. Please use the formula above to calculate concurrency requirements.

**INCORRECT:** "3" is incorrect. Please use the formula above to calculate concurrency requirements.

**References:**

<https://docs.aws.amazon.com/lambda/latest/dg/invocation-scaling.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-lambda/>

**Domain**

AWS Compute

**Question 20Correct**

A company has sensitive data that must be encrypted. The data is made up of 1 GB objects and there is a total of 150 GB of data.

What is the BEST approach for a Developer to encrypt the data using AWS KMS?

**Make an Encrypt API call to encrypt the plaintext data as ciphertext using a customer master key (CMK)**

**Make an Encrypt API call to encrypt the plaintext data as ciphertext using a customer master key (CMK) with imported key material**

**Make a GenerateDataKeyWithoutPlaintext API call that returns an encrypted copy of a data key. Use the encrypted key to encrypt the data**

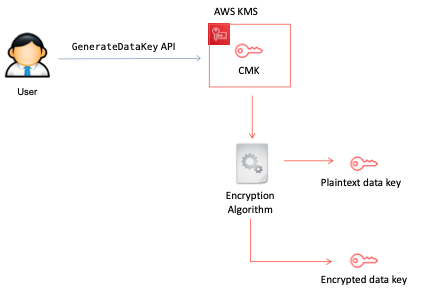
**Your answer is correct**

**Make a GenerateDataKey API call that returns a plaintext key and an encrypted copy of a data key. Use the plaintext key to encrypt the data**

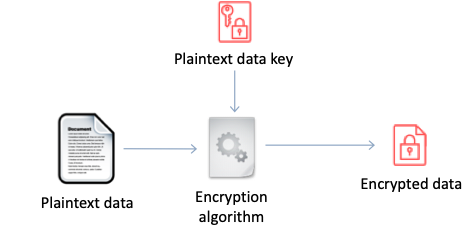
Overall explanation

To encrypt large quantities of data with the AWS Key Management Service (KMS), you must use a data encryption key rather than a customer master keys (CMK). This is because a CMK can only encrypt up to 4KB in a single operation and in this scenario the objects are 1 GB in size.

To create a data key, call the [GenerateDataKey](https://docs.aws.amazon.com/kms/latest/APIReference/API_GenerateDataKey.html) operation. AWS KMS uses the CMK that you specify to generate a data key. The operation returns a plaintext copy of the data key and a copy of the data key encrypted under the CMK. The following image shows this operation.



AWS KMS cannot use a data key to encrypt data. But you can use the data key outside of KMS, such as by using OpenSSL or a cryptographic library like the [AWS Encryption SDK](https://docs.aws.amazon.com/encryption-sdk/latest/developer-guide/). Data can then be encrypted using the plaintext data key as depicted below:



Therefore, the Developer should make a GenerateDataKey API call that returns a plaintext key and an encrypted copy of a data key, and then use the plaintext key to encrypt the data.

**CORRECT:**"Make a GenerateDataKey API call that returns a plaintext key and an encrypted copy of a data key. Use the plaintext key to encrypt the data" is the correct answer.

**INCORRECT:** "Make an Encrypt API call to encrypt the plaintext data as ciphertext using a customer master key (CMK)" is incorrect as you cannot use a CMK to encrypt objects over 4 KB in size.

**INCORRECT:** "Make an Encrypt API call to encrypt the plaintext data as ciphertext using a customer master key (CMK) with imported key material" is incorrect as you cannot use a CMK to encrypt objects over 4 KB in size.

**INCORRECT:** "Make a GenerateDataKeyWithoutPlaintext API call that returns an encrypted copy of a data key. Use the encrypted key to encrypt the data" is incorrect as you need to encrypt data with a plaintext data key.

**References:**

<https://docs.aws.amazon.com/kms/latest/developerguide/concepts.html#data-keys>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-kms/>

**Domain**

AWS Security, Identity, & Compliance

**Question 21Incorrect**

A company is reviewing their security practices. According to AWS best practice, how should access keys be managed to improve security? (Select TWO.)

**Your selection is correct**

**Use different access keys for different applications**

**Correct selection**

**Delete all access keys for the root account IAM user**

**Embed access keys directly into code**

**Your selection is incorrect**

**Rotate access keys daily**

**Use the same access key in all applications for consistency**

Overall explanation

When you access AWS programmatically, you use an access key to verify your identity and the identity of your applications. An access key consists of an access key ID (something like AKIAIOSFODNN7EXAMPLE) and a secret access key (something like wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY).

Anyone who has your access key has the same level of access to your AWS resources that you do. Steps to protect access keys include the following:

• Remove (or Don't Generate) Account Access Key – this is especially important for the root account.

• Use Temporary Security Credentials (IAM Roles) Instead of Long-Term Access Keys.

• Don't embed access keys directly into code.

• Use different access keys for different applications.

• Rotate access keys periodically.

• Remove unused access keys.

• Configure multi-factor authentication for your most sensitive operations.

**CORRECT:**"Delete all access keys for the root account IAM user" is the correct answer.

**CORRECT:**"Use different access keys for different applications" is the correct answer.

**INCORRECT:** "Embed access keys directly into code" is incorrect. This is not a best practice as this is something that should be avoided as much as possible.

**INCORRECT:** "Rotate access keys daily" is incorrect. Though this would be beneficial from a security perspective it may be hard to manage so this is not an AWS recommended best practice. AWS recommend you rotate access keys “periodically”, not “daily”.

**INCORRECT:** "Use the same access key in all applications for consistency" is incorrect. The best practice is to use different access keys for different applications.

**References:**

<https://docs.aws.amazon.com/general/latest/gr/aws-access-keys-best-practices.html#iam-user-access-keys>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-iam/>

**Domain**

AWS Security, Identity, & Compliance

**Question 22Correct**

A Development team wants to instrument their code to provide more detailed information to AWS X-Ray than simple outgoing and incoming requests. This will generate large amounts of data, so the Development team wants to implement indexing so they can filter the data.

What should the Development team do to achieve this?

**Install required plugins for the appropriate AWS SDK**

**Your answer is correct**

**Add annotations to the segment document**

**Add metadata to the segment document**

**Configure the necessary X-Ray environment variables**

Overall explanation

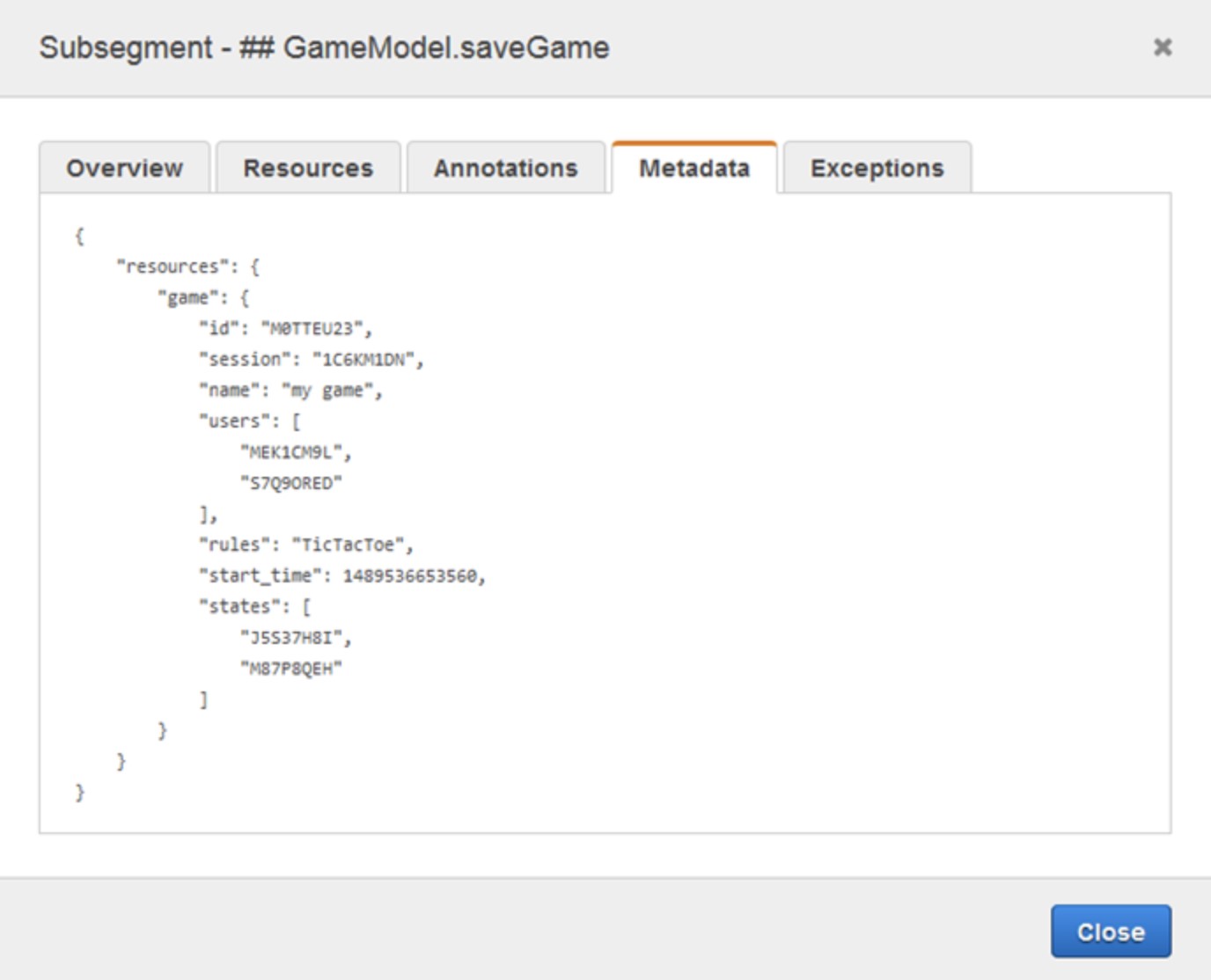
AWS X-Ray makes it easy for developers to analyze the behavior of their production, distributed applications with end-to-end tracing capabilities. You can use X-Ray to identify performance bottlenecks, edge case errors, and other hard to detect issues.

When you instrument your application, the X-Ray SDK records information about incoming and outgoing requests, the AWS resources used, and the application itself. You can add other information to the segment document as annotations and metadata. Annotations and metadata are aggregated at the trace level and can be added to any segment or subsegment.

**Annotations** are simple key-value pairs that are indexed for use with [filter expressions](https://docs.aws.amazon.com/xray/latest/devguide/xray-console-filters.html). Use annotations to record data that you want to use to group traces in the console, or when calling the [GetTraceSummaries](https://docs.aws.amazon.com/xray/latest/api/API_GetTraceSummaries.html) API. X-Ray indexes up to 50 annotations per trace.

**Metadata** are key-value pairs with values of any type, including objects and lists, but that are not indexed. Use metadata to record data you want to store in the trace but don't need to use for searching traces.

You can view annotations and metadata in the segment or subsegment details in the X-Ray console.



In this scenario, we need to add annotations to the segment document so that the data that needs to be filtered is indexed.

**CORRECT:**"Add annotations to the segment document" is the correct answer.

**INCORRECT:** "Add metadata to the segment document" is incorrect as metadata is not indexed for filtering.

**INCORRECT:** "Configure the necessary X-Ray environment variables" is incorrect as this will not result in indexing of the required data.

**INCORRECT:** "Install required plugins for the appropriate AWS SDK" is incorrect as there are no plugin requirements for the AWS SDK to support this solution as the annotations feature is available in AWS X-Ray.

**References:**

<https://docs.aws.amazon.com/xray/latest/devguide/xray-concepts.html#xray-concepts-annotations>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-developer-tools/>

**Domain**

AWS Developer Tools

**Question 23Correct**

An organization has an Amazon S3 bucket containing premier content that they intend to make available to only paid subscribers of their website. The objects in the S3 bucket are private to prevent inadvertent exposure of the premier content to non-paying website visitors.

How can the organization provide only paid subscribers the ability to download the premier content in the S3 bucket?

**Apply a bucket policy that grants anonymous users to download the content from the S3 bucket**

**Your answer is correct**

**Generate a pre-signed object URL for the premier content file when a paid subscriber requests a download**

**Enable server-side encryption on the S3 bucket for data protection against the non-paying website visitors**

**Add a bucket policy that requires Multi-Factor Authentication for requests to access the S3 bucket objects**

Overall explanation

When Amazon S3 objects are private, only the object owner has permission to access these objects. However, the object owner can optionally share objects with others by creating a presigned URL, using their own security credentials, to grant time-limited permission to download the objects.

When you create a presigned URL for your object, you must provide your security credentials, specify a bucket name, an object key, specify the HTTP method (GET to download the object) and expiration date and time. The presigned URLs are valid only for the specified duration.

Anyone who receives the presigned URL can then access the object. In this scenario, a pre-signed URL can be generated only for paying customers and they will be the only website visitors who can view the premier content.

**CORRECT:**"Generate a pre-signed object URL for the premier content file when a paid subscriber requests a download" is the correct answer.

**INCORRECT:** "Apply a bucket policy that grants anonymous users to download the content from the S3 bucket" is incorrect as this would provide everyone the ability to download the content.

**INCORRECT:** "Add a bucket policy that requires Multi-Factor Authentication for requests to access the S3 bucket objects" is incorrect as this would be very difficult to manage. Using pre-signed URLs that are dynamically generated by an application for premier users would be much simpler.

**INCORRECT:** "Enable server-side encryption on the S3 bucket for data protection against the non-paying website visitors" is incorrect as this is encryption at rest and S3 will simply unencrypt the objects when users attempt to read them. This provides privacy protection for data at rest but does not restrict access.

**References:**

<https://docs.aws.amazon.com/AmazonS3/latest/dev/ShareObjectPreSignedURL.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-s3-and-glacier/>

**Domain**

AWS Storage

**Question 24Incorrect**

A Developer created an AWS Lambda function and then attempted to add an on failure destination but received the following error:

The function's execution role does not have permissions to call SendMessage on arn:aws:sqs:us-east-1:515148212435:FailureDestination

How can the Developer resolve this issue MOST securely?

**Add the Lambda function to a group with administrative privileges**

**Your answer is incorrect**

**Add the AWSLambdaSQSQueueExecutionRole AWS managed policy to the function’s execution role**

**Add a permissions policy to the SQS queue allowing the SendMessage action and specify the AWS account number**

**Correct answer**

**Create a customer managed policy with all read/write permissions to SQS and attach the policy to the function’s execution role**

Overall explanation

The Lambda function needs the privileges to use the SendMessage API action on the Amazon SQS queue. The permissions should be assigned to the function’s execution role. The AWSLambdaSQSQueueExecutionRole AWS managed policy cannot be used as this policy does not provide the SendMessage action.

The Developer should therefore create a customer managed policy with read/write permissions to SQS and attach the policy to the function’s execution role.

**CORRECT:**"Create a customer managed policy with all read/write permissions to SQS and attach the policy to the function’s execution role" is the correct answer.

**INCORRECT:** "Add the AWSLambdaSQSQueueExecutionRole AWS managed policy to the function’s execution role" is incorrect as this does not provide the necessary permissions.

**INCORRECT:** "Add a permissions policy to the SQS queue allowing the SendMessage action and specify the AWS account number" is incorrect as this would allow any resource in the AWS account to write to the queue which is less secure.

**INCORRECT:** "Add the Lambda function to a group with administrative privileges" is incorrect as you cannot add a Lambda function to an IAM group.

**References:**

<https://docs.aws.amazon.com/lambda/latest/dg/lambda-intro-execution-role.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-lambda/>

**Domain**

AWS Security, Identity, & Compliance

**Question 25Correct**

An Auto Scaling Group (ASG) of Amazon EC2 instances is being created for processing messages from an Amazon SQS queue. To ensure the EC2 instances are cost-effective a Developer would like to configure the ASG to maintain aggregate CPU utilization at 70%.

Which type of scaling policy should the Developer choose?

**Simple Scaling Policy**

**Step Scaling Policy**

**Your answer is correct**

**Target Tracking Scaling Policy**

**Scheduled Scaling Policy**

Overall explanation

With target tracking scaling policies, you select a scaling metric and set a target value. Amazon EC2 Auto Scaling creates and manages the CloudWatch alarms that trigger the scaling policy and calculates the scaling adjustment based on the metric and the target value. The scaling policy adds or removes capacity as required to keep the metric at, or close to, the specified target value. In addition to keeping the metric close to the target value, a target tracking scaling policy also adjusts to the changes in the metric due to a changing load pattern.

For example, you can use target tracking scaling to:

• Configure a target tracking scaling policy to keep the average aggregate CPU utilization of your Auto Scaling group at 40 percent.

• Configure a target tracking scaling policy to keep the request count per target of your Elastic Load Balancing target group at 1000 for your Auto Scaling group.

The target tracking scaling policy is therefore the best choice for this scenario.

**CORRECT:**"Target Tracking Scaling Policy" is the correct answer.

**INCORRECT:** "Step Scaling Policy" is incorrect. (explanation below)

**INCORRECT:** "Simple Scaling Policy" is incorrect. (explanation below)

With step scaling and simple scaling, you choose scaling metrics and threshold values for the CloudWatch alarms that trigger the scaling process. You also define how your Auto Scaling group should be scaled when a threshold is in breach for a specified number of evaluation periods.

**INCORRECT:** "Scheduled Scaling Policy" is incorrect as this is used to schedule a scaling action at a specific time and date rather than dynamically adjusting according to load.

**References:**

<https://docs.aws.amazon.com/autoscaling/ec2/userguide/as-scaling-target-tracking.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-ec2-auto-scaling/>

**Domain**

AWS Compute

**Question 26Correct**

A legacy application is being refactored into a microservices architecture running on AWS. The microservice will include several AWS Lambda functions. A Developer will use AWS Step Functions to coordinate function execution.

How should the Developer proceed?

**Create a layer in AWS Lambda and add the functions to the layer**

**Your answer is correct**

**Create a state machine using the Amazon States Language**

**Create an AWS CloudFormation stack using a YAML-formatted template**

**Create a workflow using the StartExecution API action**

Overall explanation

AWS Step Functions is a web service that enables you to coordinate the components of distributed applications and microservices using visual workflows. You build applications from individual components that each perform a discrete function, or task, allowing you to scale and change applications quickly.

The following are key features of AWS Step Functions:

• Step Functions is based on the concepts of [tasks](https://docs.aws.amazon.com/step-functions/latest/dg/amazon-states-language-task-state.html) and [state machines](https://docs.aws.amazon.com/step-functions/latest/dg/concepts-states.html).

• You define state machines using the JSON-based [Amazon States Language](https://docs.aws.amazon.com/step-functions/latest/dg/concepts-amazon-states-language.html).

• The [Step Functions console](https://console.aws.amazon.com/states/home?region=us-east-1#/) displays a graphical view of your state machine's structure. This provides a way to visually check your state machine's logic and monitor executions.

The Developer needs to create a state machine using the Amazon States Language as this is how you can create an executable state machine that includes the Lambda functions that must be coordinated.

**CORRECT:**"Create a state machine using the Amazon States Language" is the correct answer.

**INCORRECT:** "Create an AWS CloudFormation stack using a YAML-formatted template" is incorrect as AWS Step Functions does not use CloudFormation. The Developer needs to create a state machine.

**INCORRECT:** "Create a workflow using the StartExecution API action" is incorrect as workflows are associated with Amazon SWF whereas the StartExecution API action is a Step Functions action for executing a state machine.

**INCORRECT:** "Create a layer in AWS Lambda and add the functions to the layer" is incorrect as a layer is a ZIP archive that contains libraries, a [custom runtime](https://docs.aws.amazon.com/lambda/latest/dg/runtimes-custom.html), or other dependencies that you can use to pull additional code into a Lambda function.

**References:**

<https://docs.aws.amazon.com/step-functions/latest/dg/welcome.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-application-integration-services/>

**Domain**

AWS Application Integration

**Question 27Incorrect**

The manager of a development team is setting up a shared S3 bucket for team members. The manager would like to use a single policy to allow each user to have access to their objects in the S3 bucket. Which feature can be used to generalize the policy?

**Condition**

**Resource**

**Your answer is incorrect**

**Principal**

**Correct answer**

**Variable**

Overall explanation

In some cases, you might not know the exact name of the resource when you write the policy. You might want to generalize the policy so it works for many users without having to make a unique copy of the policy for each user. For example, consider writing a policy to allow each user to have access to his or her own objects in an Amazon S3 bucket.

Instead of that explicitly specifies the user's name as part of the resource, create a single group policy that works for any user in that group. You can do this by using policy variables, a feature that lets you specify placeholders in a policy. When the policy is evaluated, the policy variables are replaced with values that come from the context of the request itself.

The following example shows a policy for an Amazon S3 bucket that uses a policy variable.



When this policy is evaluated, IAM replaces the variable ${aws:username}with the friendly name of the actual current user. This means that a single policy applied to a group of users can control access to a bucket by using the username as part of the resource's name.

**CORRECT:**"Variable" is the correct answer.

**INCORRECT:** "Condition" is incorrect. The Condition element (or Condition block) lets you specify conditions for when a policy is in effect.

**INCORRECT:** "Principal" is incorrect. You can use the Principal element in a policy to specify the principal that is allowed or denied access to a resource. However, in this scenario a variable is needed to create a generic policy that can provide the necessary permissions to different principals using variables.

**INCORRECT:** "Resource" is incorrect. The Resource element specifies the object or objects that the statement covers.

**References:**

<https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_policies_variables.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-iam/>

**Domain**

AWS Security, Identity, & Compliance

**Question 28Correct**

A company has a global presence and managers must submit large quantities of reporting data to an Amazon S3 bucket located in the us-east-1 region on weekly basis. Uploads have been slow recently, how can you improve data throughput and upload times?

**Use S3 Multi-part upload**

**Create an AWS Direct Connect connection from each remote office**

**Your answer is correct**

**Enable S3 Transfer Acceleration on the S3 bucket**

**Use an AWS Managed VPN**

Overall explanation

Amazon S3 Transfer Acceleration enables fast, easy, and secure transfers of files over long distances between your client and an S3 bucket. Transfer Acceleration takes advantage of Amazon CloudFront’s globally distributed edge locations. As the data arrives at an edge location, data is routed to Amazon S3 over an optimized network path.

You might want to use Transfer Acceleration on a bucket for various reasons, including the following:

You have customers that upload to a centralized bucket from all over the world.

You transfer gigabytes to terabytes of data on a regular basis across continents.

You are unable to utilize all of your available bandwidth over the Internet when uploading to Amazon S3.

Therefore, Amazon S3 Transfer Acceleration is an ideal solution for this use case and will result in improved throughput and upload times.

**CORRECT:**"Enable S3 Transfer Acceleration on the S3 bucket" is the correct answer.

**INCORRECT:** "Use S3 Multi-part upload" is incorrect. Multi-part upload will perform multiple uploads in parallel which does improve performance however Transfer Acceleration will utilize CloudFront and result in much improved performance over multi-part upload.

**INCORRECT:** "Create an AWS Direct Connect connection from each remote office" is incorrect as Direct Connect is used to connect from a data center into an AWS region that is local to the data center, not somewhere else in the world (though Direct Connect Gateway can do this). This would also be an extremely expensive solution.

**INCORRECT:** "Use an AWS Managed VPN" is incorrect as this is used to create an encrypted tunnel into a VPC and will not result in improved upload performance for S3 uploads.

**References:**

<https://docs.aws.amazon.com/AmazonS3/latest/dev/transfer-acceleration.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-s3-and-glacier/>

**Domain**

AWS Storage

**Question 29Correct**

A Developer is migrating Docker containers to Amazon ECS. A large number of containers will be deployed onto an existing ECS cluster that uses container instances of different instance types.

Which task placement strategy can be used to minimize the number of container instances used based on available memory?

**distinctInstance**

**random**

**spread**

**Your answer is correct**

**binpack**

Overall explanation

When a task that uses the EC2 launch type is launched, Amazon ECS must determine where to place the task based on the requirements specified in the task definition, such as CPU and memory. Similarly, when you scale down the task count, Amazon ECS must determine which tasks to terminate. You can apply task placement strategies and constraints to customize how Amazon ECS places and terminates tasks. Task placement strategies and constraints are not supported for tasks using the Fargate launch type. By default, Fargate tasks are spread across Availability Zones.

A task placement strategy is an algorithm for selecting instances for task placement or tasks for termination. For example, Amazon ECS can select instances at random, or it can select instances such that tasks are distributed evenly across a group of instances.

Amazon ECS supports the following task placement strategies:

• binpack

Place tasks based on the least available amount of CPU or memory. This minimizes the number of instances in use.

• random

Place tasks randomly.

• spread

Place tasks evenly based on the specified value. Accepted values are instanceId (or host, which has the same effect), or any platform or custom attribute that is applied to a container instance, such as attribute:ecs.availability-zone. Service tasks are spread based on the tasks from that service. Standalone tasks are spread based on the tasks from the same task group.

The Developer should use the binpack task placement strategy using available memory to determine the placement of tasks. This will minimize the number of container instances required.

**CORRECT:**"binpack" is the correct answer.

**INCORRECT:** "random" is incorrect as this would just randomly assign the tasks across the available container instances in the cluster.

**INCORRECT:** "spread" is incorrect as this would attempt to spread the tasks across the cluster instances for better high availability.

**INCORRECT:** "distinctInstance" is incorrect as this is a task placement constraint, not a strategy. This constraint would result in the tasks being each placed on a separate instance which would not assist with meeting the requirements.

**References:**

<https://docs.aws.amazon.com/AmazonECS/latest/developerguide/task-placement-strategies.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-ecs-and-eks/>

**Domain**

AWS Compute

**Question 30Correct**

A developer is designing a web application that will be used by thousands of users. The users will sign up using their email addresses and the application will store attributes for each user.

Which service should the developer use to enable users to sign-up for the web application?

**Amazon Cognito Sync**

**AWS AppSync**

**Your answer is correct**

**Amazon Cognito user pool**

**AWS Inspector**

Overall explanation

A user pool is a user directory in Amazon Cognito. With a user pool, your users can sign in to your web or mobile app through Amazon Cognito. Your users can also sign in through social identity providers like Google, Facebook, Amazon, or Apple, and through SAML identity providers.

Whether your users sign in directly or through a third party, all members of the user pool have a directory profile that you can access through a Software Development Kit (SDK).

User pools provide:

• Sign-up and sign-in services.

• A built-in, customizable web UI to sign in users.

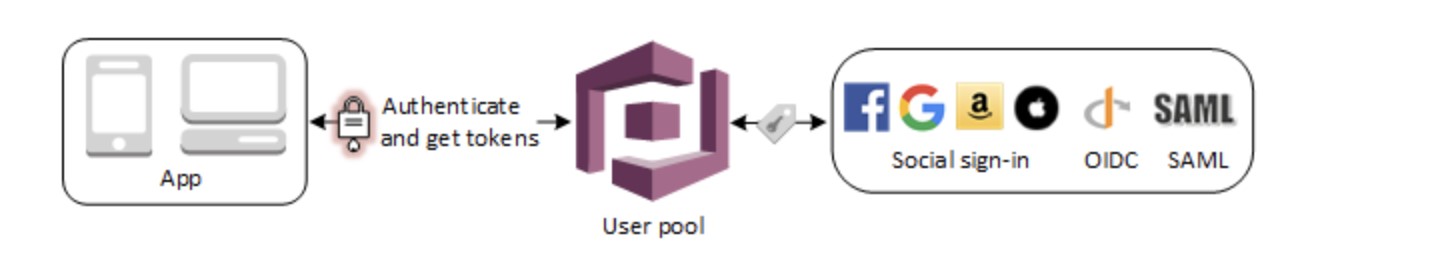
• Social sign-in with Facebook, Google, Login with Amazon, and Sign in with Apple, as well as sign-in with SAML identity providers from your user pool.

• User directory management and user profiles.

• Security features such as multi-factor authentication (MFA), checks for compromised credentials, account takeover protection, and phone and email verification.

• Customized workflows and user migration through AWS Lambda triggers.

After successfully authenticating a user, Amazon Cognito issues JSON web tokens (JWT) that you can use to secure and authorize access to your own APIs, or exchange for AWS credentials.



Therefore, an Amazon Cognito user pool is the best solution for enabling sign-up to the new web application.

**CORRECT:**"Amazon Cognito user pool" is the correct answer.

**INCORRECT:** "Amazon Cognito Sync" is incorrect as it is used to synchronize user profile data across mobile devices and the web without requiring your own backend.

**INCORRECT:** "AWS Inspector" is incorrect. Amazon Inspector is an automated security assessment service that helps improve the security and compliance of applications deployed on AWS.

**INCORRECT:** "AWS AppSync" is incorrect. AWS AppSync simplifies application development by letting you create a flexible API to securely access, manipulate, and combine data from one or more data sources.

**References:**

<https://docs.aws.amazon.com/cognito/latest/developerguide/cognito-user-identity-pools.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-cognito/>

**Domain**

AWS Security, Identity, & Compliance

**Question 31Correct**

A Developer has code running on Amazon EC2 instances that needs read-only access to an Amazon DynamoDB table.

What is the MOST secure approach the Developer should take to accomplish this task?

**Create a user access key for each EC2 instance with read-only access to DynamoDB. Place the keys in the code. Redeploy the code as keys rotate**

**Use an IAM role with Administrator access applied to the EC2 instance**

**Your answer is correct**

**Use an IAM role with an AmazonDynamoDBReadOnlyAccess policy applied to the EC2 instances**

**Run all code with only AWS account root user access keys to ensure maximum access to services**

Overall explanation

According to the principle of least privilege the Developer needs to provide the minimum permissions that application requires. The application needs read-only access and therefore an IAM role with an AmazonDynamoDBReadOnlyAccess policy applied that only provides read-only access to DynamoDB is secure.

This role can be applied to the EC2 instance through the management console or programmatically by creating an instance profile and attaching the role to the instance profile. The EC2 instance can then assume the role and get read-only access to DynamoDB.

**CORRECT:**"Use an IAM role with an AmazonDynamoDBReadOnlyAccess policy applied to the EC2 instances" is the correct answer.

**INCORRECT:** "Create a user access key for each EC2 instance with read-only access to DynamoDB. Place the keys in the code. Redeploy the code as keys rotate" is incorrect as access keys are less secure than using roles as the keys are stored in the code.

**INCORRECT:** "Run all code with only AWS account root user access keys to ensure maximum access to services" is incorrect as this is highly insecure as the access keys are stored in code and these access keys provide full permissions to the AWS account.

**INCORRECT:** "Use an IAM role with Administrator access applied to the EC2 instance" is incorrect as this does not follow the principle of least privilege and is therefore less secure. The role used should have read-only access to DynamoDB.

**References:**

<https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/using-identity-based-policies.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-dynamodb/>

**Domain**

AWS Security, Identity, & Compliance

**Question 32Correct**

A Developer is creating an application that will utilize an Amazon DynamoDB table for storing session data. The data being stored is expected to be around 4.5KB in size and the application will make 20 eventually consistent reads/sec, and 12 standard writes/sec.

How many RCUs/WCUs are required?

**10 RCU and 24 WCU**

**40 RCU and 12 WCU**

**Your answer is correct**

**20 RCU and 60 WCU**

**40 RCU and 60 WCU**

Overall explanation

With provisioned capacity mode, you specify the number of data reads and writes per second that you require for your application.

**Read capacity unit (RCU):**

• Each API call to read data from your table is a read request.

• Read requests can be strongly consistent, eventually consistent, or transactional.

• For items up to 4 KB in size, one RCU can perform one strongly consistent read request per second.

• Items larger than 4 KB require additional RCUs.

• For items up to 4 KB in size, one RCU can perform two eventually consistent read requests per second.

• Transactional read requests require two RCUs to perform one read per second for items up to 4 KB.

• For example, a strongly consistent read of an 8 KB item would require two RCUs, an eventually consistent read of an 8 KB item would require one RCU, and a transactional read of an 8 KB item would require four RCUs.

**Write capacity unit (WCU):**

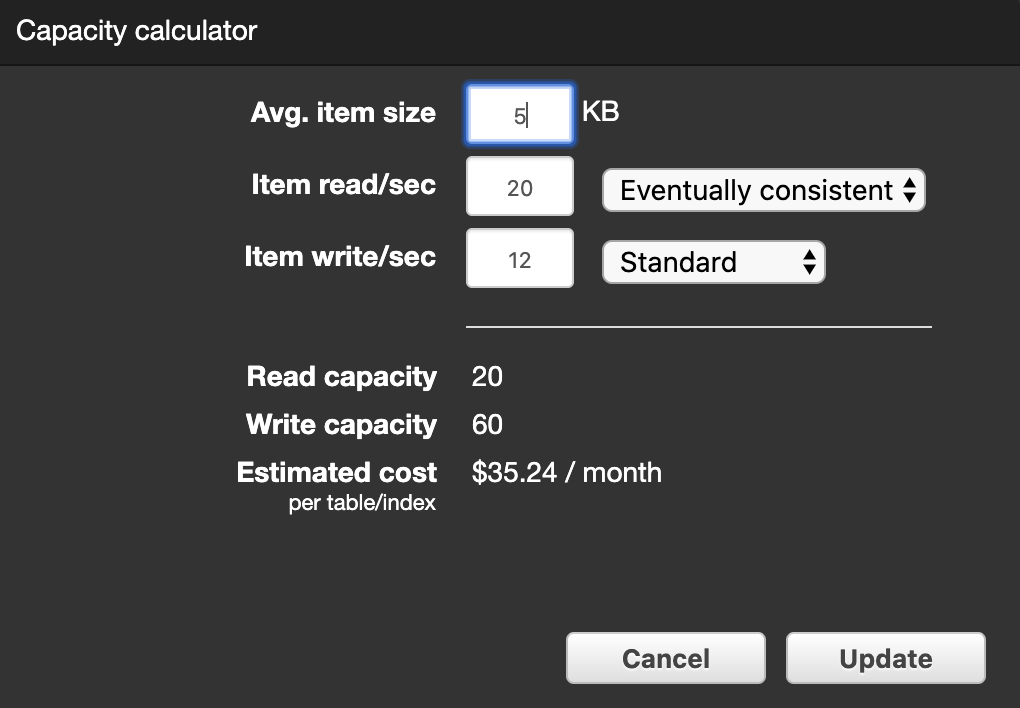
• Each API call to write data to your table is a write request.

• For items up to 1 KB in size, one WCU can perform one standard write request per second.

• Items larger than 1 KB require additional WCUs.

• Transactional write requests require two WCUs to perform one write per second for items up to 1 KB.

• For example, a standard write request of a 1 KB item would require one WCU, a standard write request of a 3 KB item would require three WCUs, and a transactional write request of a 3 KB item would require six WCUs.



To determine the number of RCUs required to handle 20 eventually consistent reads per/second with an average item size of 4.5KB, perform the following steps:

1. Determine the average item size by rounding up the next multiple of 4KB (4.5KB rounds up to 8KB).

2. Determine the RCU per item by dividing the item size by 8KB (8KB/8KB = 1).

3. Multiply the value from step 2 with the number of reads required per second (1x20 = 20).

To determine the number of WCUs required to handle 12 standard writes per/second with an average item size of 8KB, simply multiply the average item size by the number of writes required (5x12=60).

**CORRECT:**"20 RCU and 60 WCU" is the correct answer.

**INCORRECT:** "40 RCU and 60 WCU" is incorrect. This would be the correct answer for strongly consistent reads and standard writes.

**INCORRECT:** "40 RCU and 120 WCU" is incorrect. This would be the correct answer for strongly consistent reads and transactional writes.

**INCORRECT:** "6 RCU and 18 WCU" is incorrect.

**References:**

<https://aws.amazon.com/dynamodb/pricing/provisioned/>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-dynamodb/>

**Domain**

AWS Database

**Question 33Correct**

A developer has created a Docker image and uploaded it to an Amazon Elastic Container Registry (ECR) repository. How can the developer pull the image to his workstation using the docker client?

**Run aws ecr describe-images --repository-name repositoryname**

**Run docker login with an IAM key pair then issue a docker pull command specifying the image name using registry/repository[@digest]**

**Your answer is correct**

**Run aws ecr get-login-password use the output to login in then issue a docker pull command specifying the image name using registry/repository[:tag]**

**Run the docker pull command specifying the image name using registry/repository[:tag]**

Overall explanation

If you would like to run a Docker image that is available in Amazon ECR, you can pull it to your local environment with the docker pull command. You can do this from either your default registry or from a registry associated with another AWS account.

Docker CLI does not support standard AWS authentication methods, so client authentication must be handled so that ECR knows who is requesting to push or pull an image. To do this you can issue the aws ecr get-login or aws ecr get-login-password (AWS CLI v2) and then use the output to login using docker login and then issue a docker pull command specifying the image name using registry/repository[:tag]

**CORRECT:**"Run aws ecr get-login-password use the output to login in then issue a docker pull command specifying the image name using registry/repository[:tag]" is the correct answer.

**INCORRECT:** "Run the docker pull command specifying the image name using registry/repository[:tag]" is incorrect as you first need to authenticate to get an access token so you can pull the image down.

**INCORRECT:** "Run aws ecr describe-images --repository-name repositoryname" is incorrect as this would just list the images available in the repository.

**INCORRECT:** "Run docker login with an IAM key pair then issue a docker pull command specifying the image name using registry/repository[@digest]" is incorrect as you cannot run docker login with an IAM key pair.

**References:**

<https://docs.aws.amazon.com/AmazonECR/latest/userguide/docker-pull-ecr-image.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-ecs-and-eks/>

**Domain**

AWS Compute

**Question 34Correct**

A company is migrating an application with a website and MySQL database to the AWS Cloud. The company require the application to be refactored so it offers high availability and fault tolerance.

How should a Developer refactor the application? (Select TWO.)

**Migrate the website to an Auto Scaling group of EC2 instances across a single AZ and use an Elastic Load Balancer**

**Migrate the MySQL database to an Amazon DynamoDB with Global Tables**

**Your selection is correct**

**Migrate the MySQL database to an Amazon RDS Multi-AZ deployment**

**Your selection is correct**

**Migrate the website to an Auto Scaling group of EC2 instances across multiple AZs and use an Elastic Load Balancer**

**Migrate the MySQL database to an Amazon RDS instance with a Read Replica in another AZ**

Overall explanation

The key requirements are to add high availability and fault tolerance to the application. To do this the Developer should put the website into an Auto Scaling group of EC2 instances across multiple AZs. An Elastic Load Balancer can be deployed in front of the EC2 instances to distribute incoming connections. This solution is highly available and fault tolerant.

For the MySQL database the Developer should use Amazon RDS with the MySQL engine. To provide fault tolerance the Developer should configure Amazon RDS as a Multi-AZ deployment which will create a standby instance in another AZ that can be failed over to.

**CORRECT:**"Migrate the website to an Auto Scaling group of EC2 instances across multiple AZs and use an Elastic Load Balancer" is a correct answer.

**CORRECT:**"Migrate the MySQL database to an Amazon RDS Multi-AZ deployment" is also a correct answer.

**INCORRECT:** "Migrate the website to an Auto Scaling group of EC2 instances across a single AZ and use an Elastic Load Balancer" is incorrect as to be fully fault tolerant the solution should be spread across multiple AZs.

**INCORRECT:** "Migrate the MySQL database to an Amazon RDS instance with a Read Replica in another AZ" is incorrect as read replicas are used for performance, not fault tolerance

**INCORRECT:** "Migrate the MySQL database to an Amazon DynamoDB with Global Tables" is incorrect as the MySQL database is a relational database so it is a better fit to be migrated to Amazon RDS rather than DynamoDB.

**References:**

<https://d1.awsstatic.com/whitepapers/aws-building-fault-tolerant-applications.pdf>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-elastic-load-balancing-aws-elb/>

<https://digitalcloud.training/amazon-ec2-auto-scaling/>

<https://digitalcloud.training/amazon-rds/>

**Domain**

AWS Database

**Question 35Incorrect**

A nightly batch job loads 1 million new records in to a DynamoDB table. The records are only needed for one hour, and the table needs to be empty by the next night’s batch job.

Which is the MOST efficient and cost-effective method to provide an empty table?

**Write a recursive function that scans and calls out DeleteItem**

**Your answer is incorrect**

**Use DeleteItem using a ConditionExpression**

**Use BatchWriteItem to empty all of the rows**

**Correct answer**

**Create and then delete the table after the task has completed**

Overall explanation

The key requirements here are to be efficient and cost-effective. Therefore, it’s important to choose the option that requires the fewest API calls. As the table is only required for a short period of time, the most efficient and cost-effective option is to simply delete and recreate the table.

The following API actions can be used to perform this operation programmatically:

· CreateTable - The CreateTable operation adds a new table to your account.

· DeleteTable - The DeleteTable operation deletes a table and all of its items.

This solution means fewer API calls and also the table is not consuming RCUs/WCUs whilst not being used. Therefore, the best option is to create and then delete the table after the task has completed.

**CORRECT:**"Create and then delete the table after the task has completed" is the correct answer.

**INCORRECT:** "Use DeleteItem using a ConditionExpression" is incorrect as this will use more RCUs and WCUs and is not cost-effective.

**INCORRECT:** "Use BatchWriteItem to empty all of the rows" is incorrect. The BatchWriteItem operation puts or deletes multiple items (not rows) in one or more tables. This would use more RCUs and WCUs and is not cost-effective.

**INCORRECT:** "Write a recursive function that scans and calls out DeleteItem" is incorrect as scans are the least efficient and cost-effective option as all items must be retrieved from the table.

**References:**

<https://docs.aws.amazon.com/amazondynamodb/latest/APIReference/API_Operations.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-dynamodb/>

**Domain**

AWS Database

**Question 36Correct**

A Developer is publishing custom metrics for Amazon EC2 using the Amazon CloudWatch CLI. The Developer needs to add further context to the metrics being published by organizing them by EC2 instance and Auto Scaling Group.

What should the Developer add to the CLI command when publishing the metrics using put-metric-data

**The --statistic-values parameter**

**The --metric-name parameter**

**Your answer is correct**

**1. The --dimensions parameter**

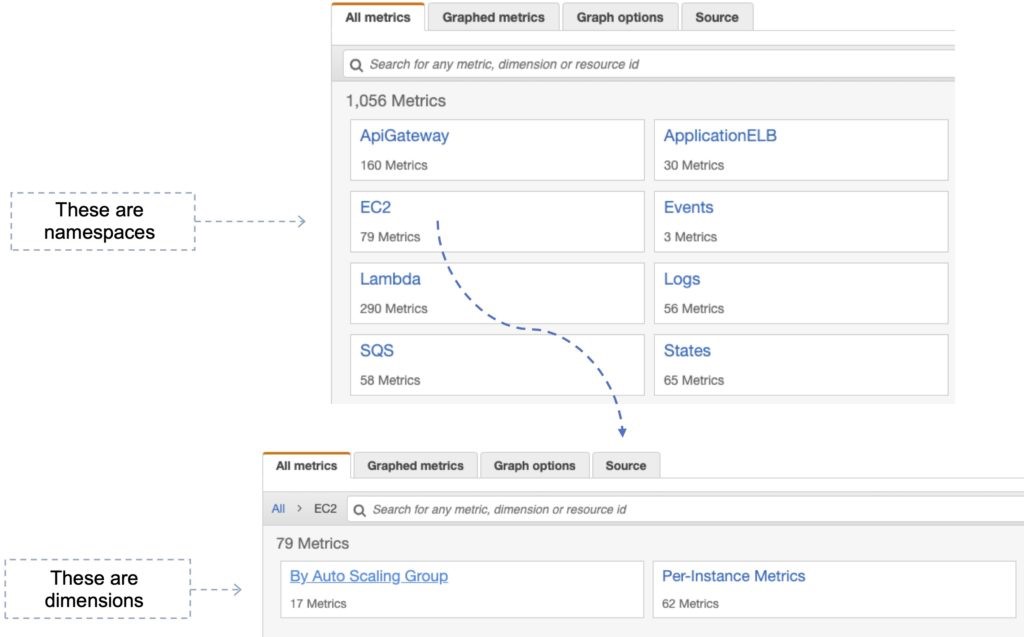
**The --namespace parameter**

Overall explanation

You can publish your own metrics to CloudWatch using the AWS CLI or an API. You can view statistical graphs of your published metrics with the AWS Management Console.

CloudWatch stores data about a metric as a series of data points. Each data point has an associated time stamp. You can even publish an aggregated set of data points called a statistic set.

In custom metrics, the --dimensions parameter is common. A dimension further clarifies what the metric is and what data it stores. You can have up to 10 dimensions in one metric, and each dimension is defined by a name and value pair.



As you can see in the above example there are two dimensions associated with the EC2 namespace. These organize the metrics by Auto Scaling Group and Per-Instance metrics. Therefore the Developer should the --dimensions parameter.

**CORRECT:**"The --dimensions parameter" is the correct answer.

**INCORRECT:** "The --namespace parameter" is incorrect as a namespace is a container for CloudWatch metrics. To add further context the Developer should use a dimension.

**INCORRECT:** "The --statistic-values parameter" is incorrect as this is a parameter associated with the publishing of statistic sets.

**INCORRECT:** "The --metric-name parameter" is incorrect as this simply provides the name for the metric that is being published.

**References:**

<https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/publishingMetrics.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-cloudwatch/>

**Domain**

AWS Management & Governance

**Question 37Correct**

A Java based application generates email notifications to customers using Amazon SNS. The emails must contain links to access data in a secured Amazon S3 bucket. What is the SIMPLEST way to maintain security of the bucket whilst allowing the customers to access specific objects?

**Use the AWS SDK for Java to assume a role with AssumeRole to gain temporary security credentials**

**Use the AWS SDK for Java to update the bucket Access Control List to allow the customers to access the bucket**

**Use the AWS SDK for Java with the AWS STS service to gain temporary security credentials**

**Your answer is correct**

**Use the AWS SDK for Java with GeneratePresignedUrlRequest to create a presigned URL**

Overall explanation

A presigned URL gives you access to the object identified in the URL, provided that the creator of the presigned URL has permissions to access that object. That is, if you receive a presigned URL to upload an object, you can upload the object only if the creator of the presigned URL has the necessary permissions to upload that object.

You can use the AWS SDK for Java to generate a presigned URL that you, or anyone you give the URL, can use to upload an object to Amazon S3. When you use the URL to upload an object, Amazon S3 creates the object in the specified bucket.

If an object with the same key that is specified in the presigned URL already exists in the bucket, Amazon S3 replaces the existing object with the uploaded object. To successfully complete an upload, you must do the following:

• Specify the HTTP PUT verb when creating the GeneratePresignedUrlRequest and HttpURLConnection objects.

• Interact with the HttpURLConnection object in some way after finishing the upload. The following example accomplishes this by using the HttpURLConnection object to check the HTTP response code.

**CORRECT:**"Use the AWS SDK for Java with GeneratePresignedUrlRequest to create a presigned URL" is the correct answer.

**INCORRECT:** "Use the AWS SDK for Java to update the bucket Access Control List to allow the customers to access the bucket" is incorrect. Bucket ACLs are used to grant access to predefined groups and accounts and are not suitable for this purpose.

**INCORRECT:** "Use the AWS SDK for Java with the AWS STS service to gain temporary security credentials" is incorrect as this requires the creation of policies and security credentials and is not as simple as creating a presigned URL.

**INCORRECT:** "Use the AWS SDK for Java to assume a role with AssumeRole to gain temporary security credentials" is incorrect as this requires the creation of policies and security credentials and is not as simple as creating a presigned URL.

**References:**

<https://docs.aws.amazon.com/AmazonS3/latest/dev/ShareObjectPreSignedURLJavaSDK.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-s3-and-glacier/>

**Domain**

AWS Storage

**Question 38Incorrect**

A Developer has created a task definition that includes the following JSON code:

1. "placementConstraints": [
2. {
3. "expression": "attribute:ecs.instance-type =~ t2.\*",
4. "type": "memberOf"
5. }
6. ]

What will be the effect for tasks using this task definition?

**Your answer is incorrect**

**They will be spread across all instances except for T2 instances**

**Correct answer**

**They will be placed only on container instances using the T2 instance type**

**They will be added to distinct instances using the T2 instance type**

**They will be placed only on container instances of T2 or T3 instance types**

Overall explanation

A task placement constraint is a rule that is considered during task placement. Task placement constraints can be specified when either running a task or creating a new service.

The memberOf task placement constraint places tasks on container instances that satisfy an expression.

The memberOf task placement constraint can be specified with the following actions:

• Running a task

• Creating a new service

• Creating a new task definition

• Creating a new revision of an existing task definition

The example JSON code uses the memberOf constraint to place tasks on T2 instances. It can be specified with the following actions: [CreateService](https://docs.aws.amazon.com/AmazonECS/latest/APIReference/API_CreateService.html), [UpdateService](https://docs.aws.amazon.com/AmazonECS/latest/APIReference/API_UpdateService.html), [RegisterTaskDefinition](https://docs.aws.amazon.com/AmazonECS/latest/APIReference/API_RegisterTaskDefinition.html), and [RunTask](https://docs.aws.amazon.com/AmazonECS/latest/APIReference/API_RunTask.html).

**CORRECT:**"They will be placed only on container instances using the T2 instance type" is the correct answer.

**INCORRECT:** "They will be added to distinct instances using the T2 instance type" is incorrect. The memberOf constraint does not choose distinct instances.

**INCORRECT:** "They will be placed only on container instances of T2 or T3 instance types" is incorrect as only T2 instance types will be used. The wildcard means any T2 instance type such as t2.micro or t2.large.

**INCORRECT:** "They will be spread across all instances except for T2 instances" is incorrect as this code ensures the instances WILL be placed on T2 instance types.

**References:**

<https://docs.aws.amazon.com/AmazonECS/latest/developerguide/task-placement-constraints.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-ecs-and-eks/>

**Domain**

AWS Compute

**Question 39Correct**

A serverless application uses Amazon API Gateway, AWS Lambda and DynamoDB. The application writes statistical data that is constantly received from sensors. The data is analyzed soon after it is written to the database and is then not required.

What is the EASIEST method to remove stale data and optimize database size?

**Use atomic counters to decrement the data when it becomes stale**

**Your answer is correct**

**Enable the TTL attribute and add expiry timestamps to items**

**Delete the table and recreate it every hour**

**Scan the table for stale data and delete it once every hour**

Overall explanation

Time to Live (TTL) for Amazon DynamoDB lets you define when items in a table expire so that they can be automatically deleted from the database. With TTL enabled on a table, you can set a timestamp for deletion on a per-item basis, allowing you to limit storage usage to only those records that are relevant.

TTL is useful if you have continuously accumulating data that loses relevance after a specific time period (for example, session data, event logs, usage patterns, and other temporary data). If you have sensitive data that must be retained only for a certain amount of time according to contractual or regulatory obligations, TTL helps you ensure that it is removed promptly and as scheduled.

Therefore, the best answer is to enable the TTL attribute and add expiry timestamps to items.

**CORRECT:**"Enable the TTL attribute and add expiry timestamps to items" is the correct answer.

**INCORRECT:** "Use atomic counters to decrement the data when it becomes stale" is incorrect. Atomic counters are useful for incrementing or decrementing the value of an attribute. A good use case is counting website visitors.

**INCORRECT:** "Scan the table for stale data and delete it once every hour" is incorrect as this is costly in terms of RCUs and WCUs. It also may result in data that has just been written but not analyzed yet.

**INCORRECT:** "Delete the table and recreate it every hour" is incorrect. The table is constantly being written to and the analysis of data happens soon after the data is written. Therefore, there isn’t a good time to delete and recreate the table as data loss is likely to occur at any time.

**References:**

<https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/TTL.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-dynamodb/>

**Question 40Correct**

A Developer is creating an AWS Lambda function that will process medical images. The function is dependent on several libraries that are not available in the Lambda runtime environment. Which strategy should be used to create the Lambda deployment package?

**Create a ZIP file with the source code and a script that installs the dependent libraries at runtime**

**Create a ZIP file with the source code and a buildspec.yaml file that installs the dependent libraries on AWS Lambda**

**Create a ZIP file with the source code. Stage the dependent libraries on an Amazon S3 bucket indicated by the Lambda environment variable LIBRARY\_PATH**

**Your answer is correct**

**Create a ZIP file with the source code and all dependent libraries**

Overall explanation

A deployment package is a ZIP archive that contains your function code and dependencies. You need to create a deployment package if you use the Lambda API to manage functions, or if you need to include libraries and dependencies other than the AWS SDK.

You can upload the package directly to Lambda, or you can use an Amazon S3 bucket, and then upload it to Lambda. If the deployment package is larger than 50 MB, you must use Amazon S3.

If your function depends on libraries not included in the Lambda runtime, you can install them to a local directory and include them in your deployment package.

**CORRECT:**"Create a ZIP file with the source code and all dependent libraries" is the correct answer.

**INCORRECT:** "Create a ZIP file with the source code and a script that installs the dependent libraries at runtime" is incorrect as though it is possible to call a script within the function code, this would need to run every time and pull in the files which would cause latency.

**INCORRECT:** "Create a ZIP file with the source code. Stage the dependent libraries on an Amazon S3 bucket indicated by the Lambda environment variable LIBRARY\_PATH" is incorrect as you cannot map an external path to a Lambda function using an environment variable.

**INCORRECT:** "Create a ZIP file with the source code and a buildspec.yaml file that installs the dependent libraries on AWS Lambda" is incorrect as a buildspec.yaml file that is used by AWS CodeBuild to run a build. The libraries need to be included in the package zip file for the Lambda function.

**References:**

<https://docs.aws.amazon.com/lambda/latest/dg/python-package.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-lambda/>

**Domain**

AWS Compute

**Question 41Correct**

An independent software vendor (ISV) uses Amazon S3 and Amazon CloudFront to distribute software updates. They would like to provide their premium customers with access to updates faster. What is the MOST efficient way to distribute these updates only to the premium customers? (Select TWO.)

**Create a signed cookie and associate it with the Amazon S3 distribution**

**Use an access control list (ACL) on the Amazon S3 bucket to restrict access based on IP address**

**Your selection is correct**

**Create an origin access identity (OAI) and associate it with the distribution and configure permissions**

**Use an IAM policy to restrict access to the content using a condition attribute and specify the IP addresses of the premium customers**

**Your selection is correct**

**Create a signed URL with access to the content and distribute it to the premium customers**

Overall explanation

To restrict access to content that you serve from Amazon S3 buckets, you create CloudFront signed URLs or signed cookies to limit access to files in your Amazon S3 bucket, and then you create a special CloudFront user called an origin access identity (OAI) and associate it with your distribution. Then you configure permissions so that CloudFront can use the OAI to access and serve files to your users, but users can't use a direct URL to the S3 bucket to access a file there. Taking these steps help you maintain secure access to the files that you serve through CloudFront.

**CORRECT:**"Create a signed URL with access to the content and distribute it to the premium customers" is the correct answer.

**CORRECT:**"Create an origin access identity (OAI) and associate it with the distribution and configure permissions" is the correct answer.

**INCORRECT:** "Create a signed cookie and associate it with the Amazon S3 distribution" is incorrect as you cannot associated signed cookies with Amazon S3 and a distribution is a CloudFront concept, not an S3 concept.

**INCORRECT:** "Use an access control list (ACL) on the Amazon S3 bucket to restrict access based on IP address" is incorrect as you cannot restrict access to buckets by IP address when using an ACL.

**INCORRECT:** "Use an IAM policy to restrict access to the content using a condition attribute and specify the IP addresses of the premium customers " is incorrect. You can restrict access to buckets using policy statements with conditions based on source IP address. However, this is cumbersome to manage as IP addresses change (and you need to know all your customer’s IPs in the first place). Also, because the content is being cached on CloudFront, this would not stop others accessing it anyway.

**References:**

<https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/private-content-signed-urls.html#private-content-choosing-canned-custom-policy>

<https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/private-content-restricting-access-to-s3.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-cloudfront/>

**Domain**

AWS Networking & Content Delivery

**Question 42Incorrect**

A Developer is creating an AWS Lambda function to process a stream of data from an Amazon Kinesis Data Stream. When the Lambda function parses the data and encounters a missing field, it exits the function with an error. The function is generating duplicate records from the Kinesis stream. When the Developer looks at the stream output without the Lambda function, there are no duplicate records.

What is the reason for the duplicates?

**Your answer is incorrect**

**The Lambda function did not advance the Kinesis stream point to the next record after the error**

**Correct answer**

**The Lambda function did not handle the error, and the Lambda service attempted to reprocess the data**

**The Lambda function is not keeping up with the amount of data coming from the stream**

**The Lambda event source used asynchronous invocation, resulting in duplicate records**

Overall explanation

When you invoke a function, two types of error can occur. Invocation errors occur when the invocation request is rejected before your function receives it. Function errors occur when your function's code or [runtime](https://docs.aws.amazon.com/lambda/latest/dg/lambda-runtimes.html) returns an error.

Depending on the type of error, the type of invocation, and the client or service that invokes the function, the retry behavior and the strategy for managing errors varies. Function errors occur when your function code or the runtime that it uses return an error.

In this case, with an event source mapping from a stream (Kinesis Data Stream), Lambda retries the entire batch of items. Therefore, the best explanation is that the Lambda function did not handle the error, and the Lambda service attempted to reprocess the data.

**CORRECT:**"The Lambda function did not handle the error, and the Lambda service attempted to reprocess the data" is the correct answer.

**INCORRECT:** "The Lambda function did not advance the Kinesis stream point to the next record after the error" is incorrect. Lambda does not advance a stream “point” to the next record. It processed records in batches.

**INCORRECT:** "The Lambda event source used asynchronous invocation, resulting in duplicate records" is incorrect as Lambda processes records from Kinesis Data Streams synchronously.

**INCORRECT:** "The Lambda function is not keeping up with the amount of data coming from the stream" is incorrect as Lambda can scale seamlessly to handle the load coming from the stream.

**References:**

<https://docs.aws.amazon.com/lambda/latest/dg/invocation-retries.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-lambda/>

**Domain**

AWS Compute

**Question 43Correct**

A Development team are developing a micro-services application that will use Docker containers on Amazon ECS. There will be 6 distinct services included in the architecture. Each service requires specific permissions to various AWS services.

What is the MOST secure way to grant the services the necessary permissions?

**Create a new Identity and Access Management (IAM) instance profile containing the required permissions for the various ECS services, then associate that instance role with the underlying EC2 instances**

**Create a single IAM policy and use principal statements referencing the ECS tasks and assigning the required permissions, then apply the policy to the ECS service**

**Your answer is correct**

**Create six separate IAM roles, each containing the required permissions for the associated ECS service, then configure each ECS task definition to reference the associated IAM role**

**Create six separate IAM roles, each containing the required permissions for the associated ECS service, then create an IAM group and configure the ECS cluster to reference that group**

Overall explanation

With IAM roles for Amazon ECS tasks, you can specify an IAM role that can be used by the containers in a task. Applications must sign their AWS API requests with AWS credentials, and this feature provides a strategy for managing credentials for your applications to use, similar to the way that Amazon EC2 instance profiles provide credentials to EC2 instances.

Instead of creating and distributing your AWS credentials to the containers or using the EC2 instance’s role, you can associate an IAM role with an ECS task definition or RunTask API operation. The applications in the task’s containers can then use the AWS SDK or CLI to make API requests to authorized AWS services.

Therefore, the most secure solution is to use a separate IAM role with the specific permissions required for an individual service and associate that role to the relevant ECS task definition. This should then be repeated for the remaining 5 services.

**CORRECT:**"Create six separate IAM roles, each containing the required permissions for the associated ECS service, then configure each ECS task definition to reference the associated IAM role" is the correct answer.

**INCORRECT:** "Create six separate IAM roles, each containing the required permissions for the associated ECS service, then create an IAM group and configure the ECS cluster to reference that group" is incorrect. The IAM role should be applied to the ECS task definition, not the ECS cluster.

**INCORRECT:** "Create a new Identity and Access Management (IAM) instance profile containing the required permissions for the various ECS services, then associate that instance role with the underlying EC2 instances" is incorrect. With IAM Roles for Tasks you apply the permissions directly to the task definition. This means multiple services can share the underlying EC2 instance and only have the minimum privileges required.

**INCORRECT:** "Create a single IAM policy and use principal statements referencing the ECS tasks and assigning the required permissions, then apply the policy to the ECS service" is incorrect. Identity-based policies attached to the ECS service can be used to control permissions for viewing, launching, and managing resources within ECS. However, for this solution we need to control the permissions for an ECS task to access other AWS services. For this we need to use IAM Roles for Tasks.

**References:**

<https://docs.aws.amazon.com/AmazonECS/latest/developerguide/task-iam-roles.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-ec2/>

**Domain**

AWS Compute

**Question 44Incorrect**

A Developer is creating a serverless website with content that includes HTML files, images, videos, and JavaScript (client-side scripts).

Which combination of services should the Developer use to create the website?

**Amazon EC2 and Amazon ElastiCache**

**Your answer is incorrect**

**AWS Lambda and Amazon API Gateway**

**Amazon ECS and Redis**

**Correct answer**

**Amazon S3 and Amazon CloudFront**

Overall explanation

You can use Amazon S3 to host a static website. On a static website, individual webpages include static content. They might also contain client-side scripts.

To host a static website on Amazon S3, you configure an Amazon S3 bucket for website hosting and then upload your website content to the bucket. When you configure a bucket as a static website, you enable static website hosting, set permissions, and add an index document.

To get content closer to users for better performance you can also use Amazon CloudFront in front of the S3 static website. To serve a static website hosted on Amazon S3, you can deploy a CloudFront distribution using one of these configurations:

• Using a REST API endpoint as the origin with access restricted by an origin access identity (OAI)

• Using a website endpoint as the origin with anonymous (public) access allowed

• Using a website endpoint as the origin with access restricted by a Referer header

Therefore, the combination of services should be Amazon S3 and Amazon CloudFront

**CORRECT:**"Amazon S3 and Amazon CloudFront" is the correct answer.

**INCORRECT:** "Amazon EC2 and Amazon ElastiCache" is incorrect. The website is supposed to be serverless and neither of these services are serverless as they both use Amazon EC2 instances.

**INCORRECT:** "Amazon ECS and Redis" is incorrect. These services are also not serverless. Also Redis is an in-memory cache and is typically placed in front of a database, not a Docker container.

**INCORRECT:** "AWS Lambda and Amazon API Gateway" is incorrect. These are both serverless services however for serving content such as HTML files, images, videos, and client-side JavaScript, Amazon S3 and CloudFront are more appropriate.

**References:**

<https://docs.aws.amazon.com/AmazonS3/latest/dev/WebsiteHosting.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-cloudfront/>

**Domain**

AWS Networking & Content Delivery

**Question 45Correct**

An Amazon RDS database is experiencing a high volume of read requests that are slowing down the database. Which fully managed, in-memory AWS database service can assist with offloading reads from the RDS database?

**Memcached on Amazon EC2**

**Amazon Aurora Serverless**

**Amazon RDS Read Replica**

**Your answer is correct**

**Amazon ElastiCache Redis**

Overall explanation

ElastiCache is a web service that makes it easy to deploy and run Memcached or Redis protocol-compliant server nodes in the cloud. The in-memory caching provided by ElastiCache can be used to significantly improve latency and throughput for many read-heavy application workloads or compute-intensive workloads.

This is a fully managed AWS service and is ideal for offloading reads from the main database to reduce the performance impact.

**CORRECT:**"Amazon ElastiCache Redis" is the correct answer.

**INCORRECT:** "Amazon RDS Read Replica" is incorrect as it is not an in-memory database. RDS Read Replicas can be used for offloading reads from the main database, however.

**INCORRECT:** "Amazon Aurora Serverless" is incorrect. Aurora Serverless is not an in-memory solution, nor is it suitable for functioning as a method of offloading reads from RDS databases.

**INCORRECT:** "Memcached on Amazon EC2" is incorrect as this is an implementation of Memcached running on EC2 and therefore is not a fully managed AWS service.

**References:**

<https://aws.amazon.com/elasticache/redis/>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-elasticache/>

**Domain**

AWS Database

**Question 46Incorrect**

A monitoring application that keeps track of a large eCommerce website uses Amazon Kinesis for data ingestion. During periods of peak data rates, the producers are not making best use of the available shards.  
What step will allow the producers to better utilize the available shards and increase write throughput to the Kinesis data stream?

**Create an SQS queue and decouple the producers from the Kinesis data stream**

**Increase the shard count of the stream using UpdateShardCount**

**Your answer is incorrect**

**Ingest multiple records into the stream in a single call using BatchWriteItem**

**Correct answer**

**Install the Kinesis Producer Library (KPL) for ingesting data into the stream**

Overall explanation

An Amazon Kinesis Data Streams producer is an application that puts user data records into a Kinesis data stream (also called data ingestion). The Kinesis Producer Library (KPL) simplifies producer application development, allowing developers to achieve high write throughput to a Kinesis data stream.

The KPL is an easy-to-use, highly configurable library that helps you write to a Kinesis data stream. It acts as an intermediary between your producer application code and the Kinesis Data Streams API actions. The KPL performs the following primary tasks:

• Writes to one or more Kinesis data streams with an automatic and configurable retry mechanism

• Collects records and uses PutRecords to write multiple records to multiple shards per request

• Aggregates user records to increase payload size and improve throughput

• Integrates seamlessly with the [Kinesis Client Library](https://docs.aws.amazon.com/kinesis/latest/dev/developing-consumers-with-kcl.html) (KCL) to de-aggregate batched records on the consumer

• Submits Amazon CloudWatch metrics on your behalf to provide visibility into producer performance

The question states that the producers are not making best use of the available shards. Therefore, we understand that there are adequate shards available but the producers are either not discovering them or are not writing records at sufficient speed to best utilize the shards.

We therefore need to install the Kinesis Producer Library (KPL) for ingesting data into the stream.

**CORRECT:**"Install the Kinesis Producer Library (KPL) for ingesting data into the stream" is the correct answer.

**INCORRECT:** "Create an SQS queue and decouple the producers from the Kinesis data stream " is incorrect. In this case we need to ensure our producers are discovering shards and writing records to best utilize those shards.

**INCORRECT:** "Increase the shard count of the stream using UpdateShardCount" is incorrect. The problem statement is that the producers are not making best use of the available shards. We don’t need to add more shards, we need to make sure the producers are discovering and then fully utilizing the shards that are available.

**INCORRECT:** "Ingest multiple records into the stream in a single call using BatchWriteItem" is incorrect. This API is used with DynamoDB, not Kinesis.

**References:**

<https://docs.aws.amazon.com/streams/latest/dev/developing-producers-with-kpl.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-kinesis/>

**Domain**

AWS Application Integration

**Question 47Correct**

A company will be hiring a large number of Developers for a series of projects. The Develops will bring their own devices to work and the company want to ensure consistency in tooling. The Developers must be able to write, run, and debug applications with just a browser, without needing to install or maintain a local Integrated Development Environment (IDE).

Which AWS service should the Developers use?

**AWS X-Ray**

**AWS CodeCommit**

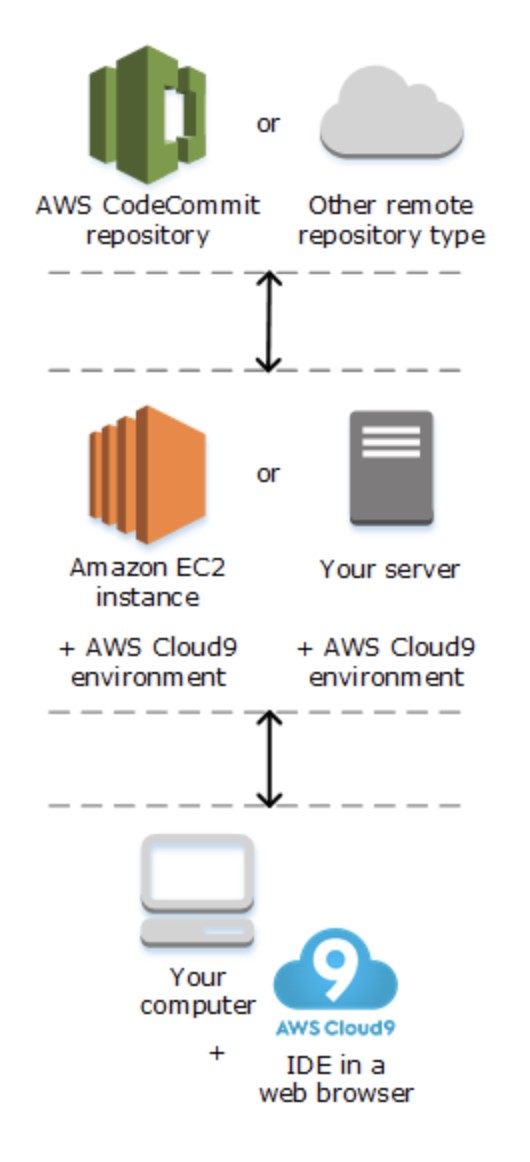
**Your answer is correct**

**AWS Cloud9**

**AWS CodeDeploy**

Overall explanation

AWS Cloud9 is an integrated development environment, or IDE. The AWS Cloud9 IDE offers a rich code-editing experience with support for several programming languages and runtime debuggers, and a built-in terminal. It contains a collection of tools that you use to code, build, run, test, and debug software, and helps you release software to the cloud.



You access the AWS Cloud9 IDE through a web browser. You can configure the IDE to your preferences. You can switch color themes, bind shortcut keys, enable programming language-specific syntax coloring and code formatting, and more.

**CORRECT:**"AWS Cloud9" is the correct answer.

**INCORRECT:** "AWS CodeCommit" is incorrect. AWS CodeCommit is a fully-managed [source control](https://aws.amazon.com/devops/source-control/) service that hosts secure Git-based repositories. It is not an IDE.

**INCORRECT:** "AWS CodeDeploy" is incorrect. CodeDeploy is a deployment service that automates application deployments to Amazon EC2 instances, on-premises instances, serverless Lambda functions, or Amazon ECS services.

**INCORRECT:** "AWS X-Ray" is incorrect. AWS X-Ray helps developers analyze and debug production, distributed applications, such as those built using a microservices architecture.

**References:**

<https://docs.aws.amazon.com/cloud9/latest/user-guide/welcome.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-developer-tools/>

**Domain**

AWS Developer Tools

**Question 48Correct**

A website delivers images stored in an Amazon S3 bucket. The site uses Amazon Cognito-enabled and guest users without logins need to be able to view the images from the S3 bucket..

How can a Developer enable access for guest users to the AWS resources?

**Create a blank user ID in a user pool, add to the user group, and grant access to AWS resources**

**Create a new user pool, disable authentication access, and grant access to AWS resources**

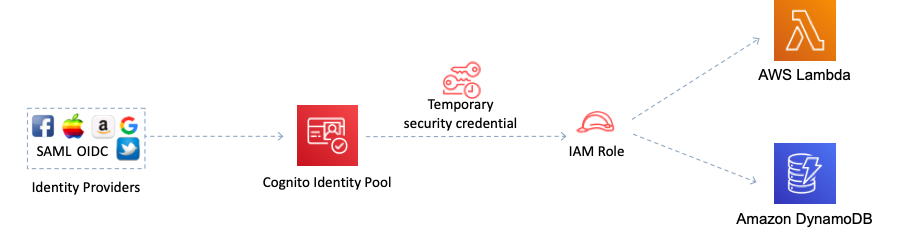
**Your answer is correct**

**Create a new identity pool, enable access to unauthenticated identities, and grant access to AWS resources**

**Create a new user pool, enable access to unauthenticated identities, and grant access to AWS resources**

Overall explanation

Amazon Cognito identity pools provide temporary AWS credentials for users who are guests (unauthenticated) and for users who have been authenticated and received a token. An identity pool is a store of user identity data specific to your account.



Amazon Cognito identity pools support both authenticated and unauthenticated identities. Authenticated identities belong to users who are authenticated by any supported identity provider. Unauthenticated identities typically belong to guest users.

• To configure authenticated identities with a public login provider, see [Identity Pools (Federated Identities) External Identity Providers](https://docs.aws.amazon.com/cognito/latest/developerguide/external-identity-providers.html).

• To configure your own backend authentication process, see [Developer Authenticated Identities (Identity Pools)](https://docs.aws.amazon.com/cognito/latest/developerguide/developer-authenticated-identities.html).

Therefore, the Developer should create a new identity pool, enable access to unauthenticated identities, and grant access to AWS resources.

**CORRECT:**"Create a new identity pool, enable access to unauthenticated identities, and grant access to AWS resources" is the correct answer.

**INCORRECT:** "Create a blank user ID in a user pool, add to the user group, and grant access to AWS resources" is incorrect as you must use identity pools for unauthenticated users.

**INCORRECT:** "Create a new user pool, enable access to unauthenticated identities, and grant access to AWS resources" is incorrect as you must use identity pools for unauthenticated users.

**INCORRECT:** "Create a new user pool, disable authentication access, and grant access to AWS resources" is incorrect as you must use identity pools for unauthenticated users.

**References:**

<https://docs.aws.amazon.com/cognito/latest/developerguide/identity-pools.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-cognito/>

**Domain**

AWS Security, Identity, & Compliance

**Question 49Correct**

A company is migrating an on-premises web application to AWS. The web application runs on a single server and stores session data in memory. On AWS the company plan to implement multiple Amazon EC2 instances behind an Elastic Load Balancer (ELB). The company want to refactor the application so that data is resilient if an instance fails and user downtime is minimized.

Where should the company move session data to MOST effectively reduce downtime and make users’ session data more fault tolerant?

**The web server’s primary disk**

**An Amazon EC2 instance dedicated to session data**

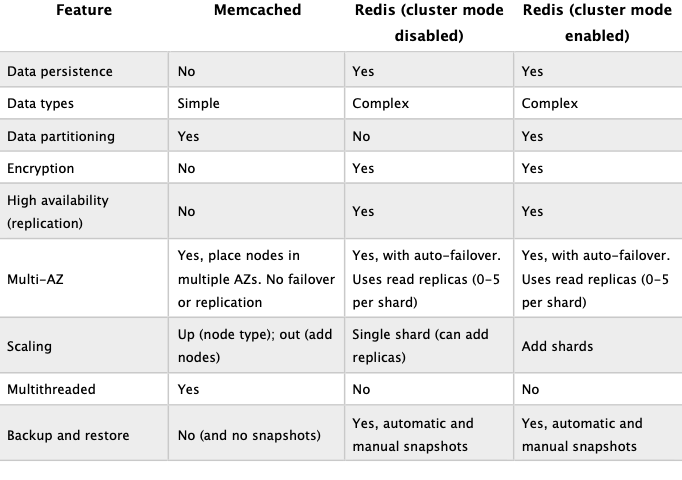
**A second Amazon EBS volume**

**Your answer is correct**

**An Amazon ElastiCache for Redis cluster**

Overall explanation

ElastiCache is a fully managed, low latency, in-memory data store that supports either Memcached or Redis. The Redis engine supports multi-AZ and high availability.



With ElastiCache the company can move the session data to a high-performance, in-memory data store that is well suited to this use case. This will provide high availability for the session data in the case of EC2 instance failure and will reduce downtime for users.

**CORRECT:**"An Amazon ElastiCache for Redis cluster" is the correct answer.

**INCORRECT:** "A second Amazon EBS volume" is incorrect as the session data needs to be highly available so should not be stored on an EC2 instance.

**INCORRECT:** "The web server’s primary disk" is incorrect as the session data needs to be highly available so should not be stored on an EC2 instance.

**INCORRECT:** "An Amazon EC2 instance dedicated to session data" is incorrect as the session data needs to be highly available so should not be stored on an EC2 instance.

**References:**

<https://aws.amazon.com/elasticache/features/>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-elasticache/>

**Domain**

AWS Database

**Question 50Correct**

A Developer wants the ability to roll back to a previous version of an AWS Lambda function in the event of errors caused by a new deployment.

How can the Developer achieve this with MINIMAL impact on users?

**Change the application to use an alias that points to the current version. Deploy the new version of the code. Update the alias to use the newly deployed version. If too many errors are encountered, point the alias back to the previous version**

**Your answer is correct**

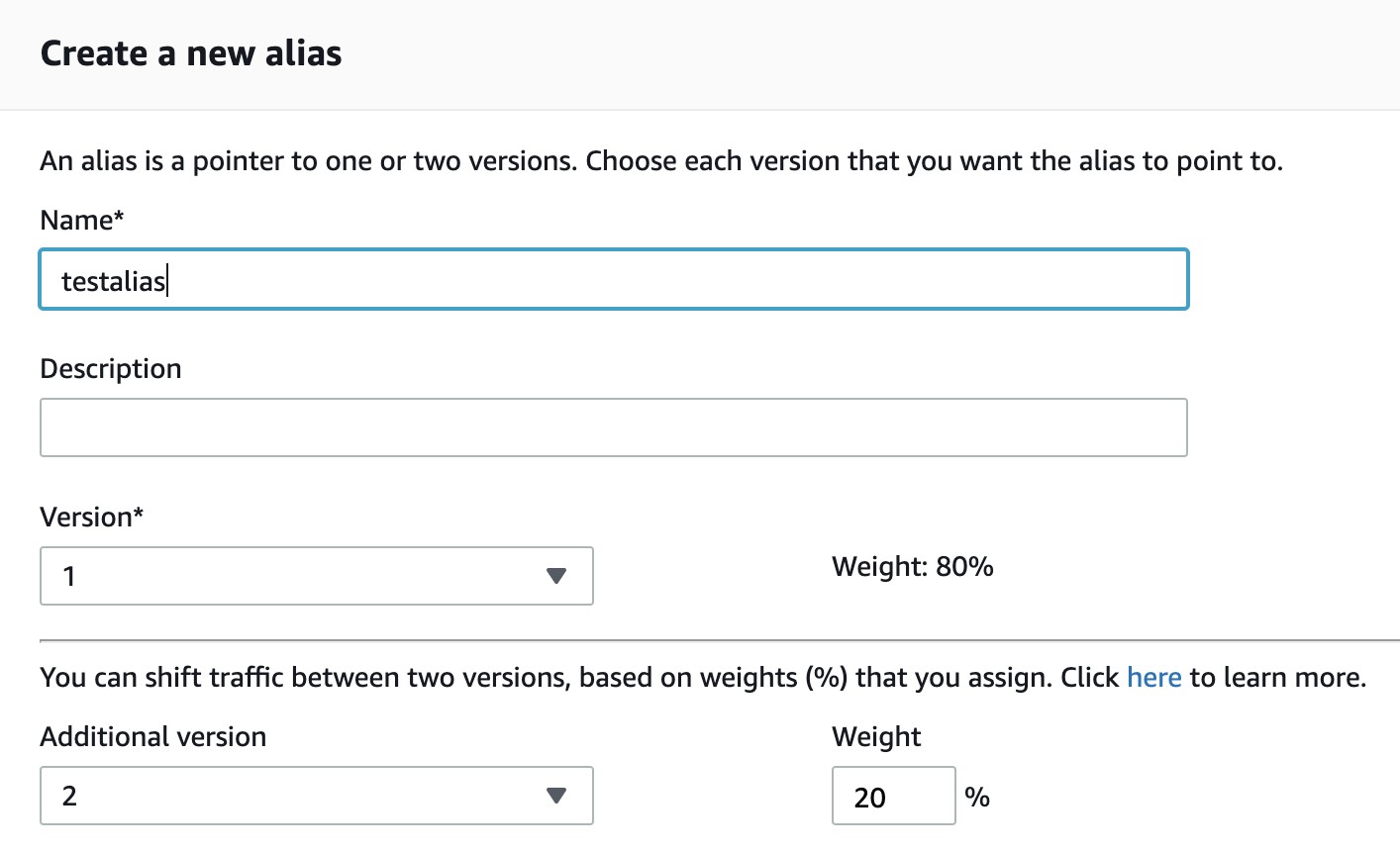
**Change the application to use an alias that points to the current version. Deploy the new version of the code. Update the alias to direct 10% of users to the newly deployed version. If too many errors are encountered, send 100% of traffic to the previous version**

**Change the application to use the $LATEST version. Update and save code. If too many errors are encountered, modify and save the code**

**Change the application to use a version ARN that points to the latest published version. Deploy the new version of the code. Update the application to point to the ARN of the new version of the code. If too many errors are encountered, point the application back to the ARN of the previous version**

Overall explanation

You can create one or more aliases for your AWS Lambda function. A Lambda alias is like a pointer to a specific Lambda function version. Users can access the function version using the alias ARN.



You can update the versions that an alias points to and you can also add multiple versions and use weightings to direct a percentage of traffic to a new version of the code.

For this example the best choice is to use an alias and direct 10% of traffic to the new version. If errors are encountered the rollback is easy (change the pointer in the alias) and a minimum of impact has been made to users.

**CORRECT:**"Change the application to use an alias that points to the current version. Deploy the new version of the code. Update the alias to direct 10% of users to the newly deployed version. If too many errors are encountered, send 100% of traffic to the previous version" is the correct answer.

**INCORRECT:** "Change the application to use an alias that points to the current version. Deploy the new version of the code. Update the alias to use the newly deployed version. If too many errors are encountered, point the alias back to the previous version" is incorrect. This is not the best answer as 100% of the users will be directed to the new version so if any errors do occur more users will be affected.

**INCORRECT:** "Change the application to use a version ARN that points to the latest published version. Deploy the new version of the code. Update the application to point to the ARN of the new version of the code. If too many errors are encountered, point the application back to the ARN of the previous version" is incorrect. This answer involves a lot of updates to the application that could be completely avoided by using an alias.

**INCORRECT:** "Change the application to use the $LATEST version. Update and save code. If too many errors are encountered, modify and save the code" is incorrect as this is against best practice. The $LATEST is the unpublished version of the code where you make changes. You should publish to a version when the code is ready for deployment.

**References:**

<https://docs.aws.amazon.com/lambda/latest/dg/configuration-aliases.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-lambda/>

**Domain**

AWS Compute

**Question 51Correct**

A Development team are creating a financial trading application. The application requires sub-millisecond latency for processing trading requests. Amazon DynamoDB is used to store the trading data. During load testing the Development team found that in periods of high utilization the latency is too high and read capacity must be significantly over-provisioned to avoid throttling.

How can the Developers meet the latency requirements of the application?

**Store the trading data in Amazon S3 and use Transfer Acceleration**

**Your answer is correct**

**Use Amazon DynamoDB Accelerator (DAX) to cache the data**

**Use exponential backoff in the application code for DynamoDB queries**

**Create a Global Secondary Index (GSI) for the trading data**

Overall explanation

Amazon DynamoDB is designed for scale and performance. In most cases, the DynamoDB response times can be measured in single-digit milliseconds. However, there are certain use cases that require response times in microseconds. For these use cases, DynamoDB Accelerator (DAX) delivers fast response times for accessing eventually consistent data.

DAX is a DynamoDB-compatible caching service that enables you to benefit from fast in-memory performance for demanding applications. DAX addresses three core scenarios:

1. As an in-memory cache, DAX reduces the response times of eventually consistent read workloads by an order of magnitude from single-digit milliseconds to microseconds.

2. DAX reduces operational and application complexity by providing a managed service that is API-compatible with DynamoDB. Therefore, it requires only minimal functional changes to use with an existing application.

3. For read-heavy or bursty workloads, DAX provides increased throughput and potential operational cost savings by reducing the need to overprovision read capacity units. This is especially beneficial for applications that require repeated reads for individual keys.

In this scenario the question is calling for sub-millisecond (e.g. microsecond) latency and this is required for read traffic as evidenced by the need to over-provision reads. Therefore, DynamoDB DAX would be the best solution for reducing the latency and meeting the requirements.

**CORRECT:**"Use Amazon DynamoDB Accelerator (DAX) to cache the data" is the correct answer.

**INCORRECT:** "Create a Global Secondary Index (GSI) for the trading data" is incorrect as a GSI is used to speed up queries on non-key attributes. There is no requirement here for a Global Secondary Index.

**INCORRECT:** "Use exponential backoff in the application code for DynamoDB queries" is incorrect as this may reduce the requirement for over-provisioning reads but it will not solve the problem of reducing latency. With this solution the application performance will be worse, it’s a case of reducing cost along with performance.

**INCORRECT:** "Store the trading data in Amazon S3 and use Transfer Acceleration" is incorrect as this will not reduce the latency of the application. Transfer Acceleration is used for improving performance of uploads of data to Amazon S3.

**References:**

<https://aws.amazon.com/dynamodb/dax/>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-dynamodb/>

**Domain**

AWS Database

**Question 52Incorrect**

An application uses Amazon EC2 instances, AWS Lambda functions and an Amazon SQS queue. The Developer must ensure all communications are within an Amazon VPC using private IP addresses. How can this be achieved? (Select TWO.)

**Your selection is correct**

**Add the AWS Lambda function to the VPC**

**Your selection is incorrect**

**Create a VPC endpoint for AWS Lambda**

**Correct selection**

**Create a VPC endpoint for Amazon SQS**

**Create a VPN and connect the services to the VPG**

**Create the Amazon SQS queue within a VPC**

Overall explanation

This solution can be achieved by adding the AWS Lambda function to a VPC through the function configuration, and by creating a VPC endpoint for Amazon SQS. This will result in the services using purely private IP addresses to communicate without traversing the public Internet.

**CORRECT:**"Add the AWS Lambda function to the VPC" is the correct answer.

**CORRECT:**"Create a VPC endpoint for Amazon SQS" is also correct.

**INCORRECT:** "Create the Amazon SQS queue within a VPC" is incorrect as you cannot create a queue within a VPC as Amazon SQS is a public service.

**INCORRECT:** "Create a VPC endpoint for AWS Lambda" is incorrect as you can't create a VPC endpoint for AWS Lambda. You can, however, connect a Lambda function to a VPC.

**INCORRECT:** "Create a VPN and connect the services to the VPG" is incorrect as you cannot create a VPN between each of these services.

**References:**

<https://aws.amazon.com/about-aws/whats-new/2018/12/amazon-sqs-vpc-endpoints-aws-privatelink/>

<https://docs.aws.amazon.com/lambda/latest/dg/configuration-vpc.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-vpc/>

<https://digitalcloud.training/aws-lambda/>

<https://digitalcloud.training/aws-application-integration-services/>

**Domain**

AWS Networking & Content Delivery

**Question 53Correct**

A retail organization stores stock information in an Amazon RDS database. An application reads and writes data to the database. A Developer has been asked to provide read access to the database from a reporting application in another region.

Which configuration would provide BEST performance for the reporting application without impacting the performance of the main database?

**Implement a cross-region multi-AZ deployment in the region where the reporting application will run**

**Your answer is correct**

**Implement a cross-region read replica in the region where the reporting application will run**

**Implement a read replica in another AZ and configure the reporting application to connect to the read replica using a VPN connection**

**Create a snapshot of the database and create a new database from the snapshot in the region where the reporting application will run**

Overall explanation

With Amazon RDS, you can create a MariaDB, MySQL, Oracle, or PostgreSQL read replica in a different AWS Region than the source DB instance. Creating a cross-Region read replica isn't supported for SQL Server on Amazon RDS.

You create a read replica in a different AWS Region to do the following:

• Improve your disaster recovery capabilities.

• Scale read operations into an AWS Region closer to your users.

• Make it easier to migrate from a data center in one AWS Region to a data center in another AWS Region.

Creating a read replica in a different AWS Region from the source instance is similar to creating a replica in the same AWS Region. You can use the AWS Management Console, run the [create-db-instance-read-replica](https://docs.aws.amazon.com/cli/latest/reference/rds/create-db-instance-read-replica.html) command, or call the [CreateDBInstanceReadReplica](https://docs.aws.amazon.com/AmazonRDS/latest/APIReference/API_CreateDBInstanceReadReplica.html) API operation.

Creating read replica in the region where the reporting application is going to run will provide the best performance as latency will be much lower than connecting across regions. As the database is a replica it will also be continuously updated using asynchronous replication so the reporting application will have the latest data available.

**CORRECT:**"Implement a cross-region read replica in the region where the reporting application will run" is the correct answer.

**INCORRECT:** "Implement a cross-region multi-AZ deployment in the region where the reporting application will run" is incorrect as multi-AZ is used across availability zones, not regions.

**INCORRECT:** "Create a snapshot of the database and create a new database from the snapshot in the region where the reporting application will run" is incorrect as this would be OK from a performance perspective but the database would not receive ongoing updates from the main database so the data would quickly become out of date.

**INCORRECT:** "Implement a read replica in another AZ and configure the reporting application to connect to the read replica using a VPN connection" is incorrect as this would result in much higher latency than having the database in the local region close to the reporting application and would impact performance.

**References:**

<https://docs.aws.amazon.com/AmazonRDS/latest/UserGuide/USER_ReadRepl.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-rds/>

**Domain**

AWS Database

**Question 54Correct**

A serverless application composed of multiple Lambda functions has been deployed. A developer is setting up AWS CodeDeploy to manage the deployment of code updates. The developer would like a 10% of the traffic to be shifted to the new version in equal increments, 10 minutes apart.

Which setting should be chosen for configuring how traffic is shifted?

**Canary**

**Blue/green**

**Your answer is correct**

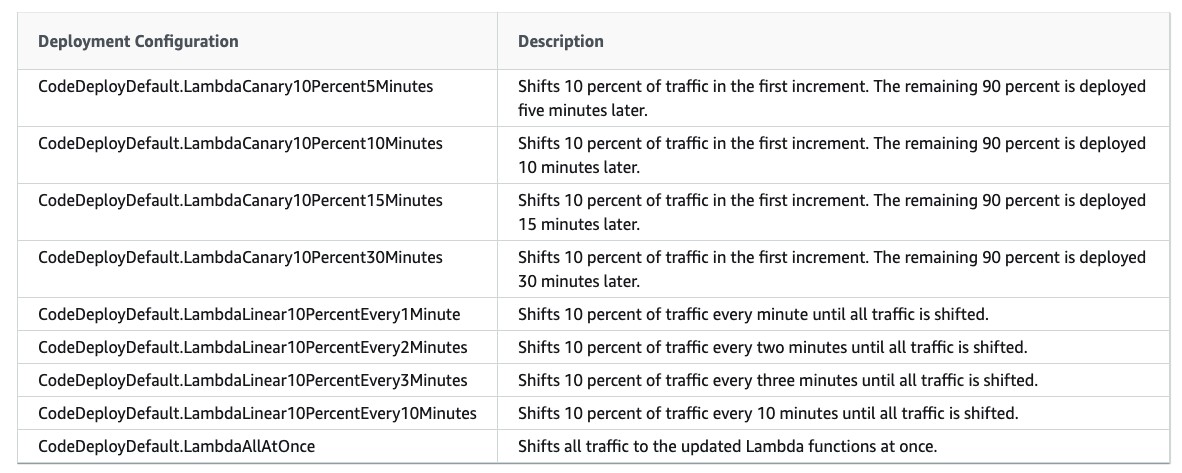
**Linear**

**All-at-once**

Overall explanation

A deployment configuration is a set of rules and success and failure conditions used by CodeDeploy during a deployment. These rules and conditions are different, depending on whether you deploy to an EC2/On-Premises compute platform or an AWS Lambda compute platform.

The following table lists the predefined configurations available for AWS Lambda deployments.



As you can see from the table above, the linear option shifts a specific amount of traffic in equal increments of time. Therefore, the following option should be chosen:

CodeDeployDefault.LambdaLinear10PercentEvery10Minutes

**CORRECT:**"Linear" is the correct answer.

**INCORRECT:** "Canary" is incorrect as it does not shift traffic in equal increments.

**INCORRECT:** "All-at-once" is incorrect as it shifts all traffic at once.

**INCORRECT:** "Blue/green" is incorrect as it is a type of deployment, not a setting for traffic shifting.

**References:**

<https://docs.aws.amazon.com/codedeploy/latest/userguide/deployment-configurations.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-developer-tools/>

**Domain**

AWS Developer Tools

**Question 55Correct**

A company is designing a new application that will store thousands of terabytes of data. They need a fully managed NoSQL data store that provides low-latency and can store key-value pairs. Which type of database should they use?

**Amazon ElastiCache**

**Amazon RDS**

**Amazon S3**

**Your answer is correct**

**Amazon DynamoDB**

Overall explanation

Amazon DynamoDB is a fully managed NoSQL database. With DynamoDB, you can create database tables that can store and retrieve any amount of data and serve any level of request traffic. You can scale up or scale down your tables' throughput capacity without downtime or performance degradation.

DynamoDB is a key-value database. A key-value database is a type of nonrelational database that uses a simple key-value method to store data. A key-value database stores data as a collection of key-value pairs in which a key serves as a unique identifier. Both keys and values can be anything, ranging from simple objects to complex compound objects.

**CORRECT:**"Amazon DynamoDB" is the correct answer.

**INCORRECT:** "Amazon RDS" is incorrect as RDS is a SQL (not a NoSQL) type of database.

**INCORRECT:** "Amazon ElastiCache" is incorrect as ElastiCache is a SQL (not a NoSQL) type of database. ElastiCache is an in-memory database typically used for caching data.

**INCORRECT:** "Amazon S3" is incorrect as S3 is not a NoSQL database. S3 is an object storage system.

**References:**

<https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/Introduction.html>

<https://aws.amazon.com/nosql/key-value/>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-dynamodb/>

**Domain**

AWS Database

**Question 56Incorrect**

An application that is being migrated to AWS and refactored requires a storage service. The storage service should provide a standards-based REST web service interface and store objects based on keys.

Which AWS service would be MOST suitable?

**Amazon EBS**

**Your answer is incorrect**

**Amazon DynamoDB**

**Amazon EFS**

**Correct answer**

**Amazon S3**

Overall explanation

Amazon S3 is object storage built to store and retrieve any amount of data from anywhere on the Internet. Amazon S3 uses standards-based REST and SOAP interfaces designed to work with any internet-development toolkit.

Amazon S3 is a simple key-based object store. The key is the name of the object and the value is the actual data itself. Keys can be any string, and they can be constructed to mimic hierarchical attributes.

**CORRECT:**"Amazon S3" is the correct answer.

**INCORRECT:** "Amazon DynamoDB" is incorrect. DynamoDB is a key/value database service that provides tables to store your data. This is not the most suitable solution for this requirement as the cost will be higher and there are more design considerations that need to be addressed.

**INCORRECT:** "Amazon EBS" is incorrect as this is a block-based storage system with which you attach volumes to Amazon EC2 instances. It is not a key-based object storage system.

**INCORRECT:** "Amazon EFS" is incorrect as this is a filesystem that you mount to Amazon EC2 instances, it is also not a key-based object storage system.

**References:**

<https://docs.aws.amazon.com/AmazonS3/latest/dev/Welcome.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-s3-and-glacier/>

**Domain**

AWS Storage

**Question 57Incorrect**

A company has hired a team of remote Developers. The Developers need to work programmatically with AWS resources from their laptop computers.

Which security components MUST the Developers use to authenticate? (Select TWO.)

**IAM user ID**

**Your selection is correct**

**Secret access key**

**Console password**

**Your selection is incorrect**

**MFA device**

**Correct selection**

**Access key ID**

Overall explanation

Access keys consist of two parts: an access key ID (for example, AKIAIOSFODNN7EXAMPLE) and a secret access key (for example, wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY). You use access keys to sign programmatic requests that you make to AWS if you use AWS CLI commands (using the SDKs) or using AWS API operations.

For this scenario, the Developers will be connecting programmatically to AWS resources and will therefore be required to use an access key ID and secret access key.

**CORRECT:**"Access key ID" is a correct answer.

**CORRECT:**"Secret access key" is a correct answer.

**INCORRECT:** "Console password " is incorrect as this is used for accessing AWS via the console with an IAM user ID and is not used for programmatic access.

**INCORRECT:** "IAM user ID" is incorrect as the IAM user ID is used with the password (see above) to access the AWS management console.

**INCORRECT:** "MFA device" is incorrect as this is not required for making programmatic requests but can be added for additional security

**References:**

<https://docs.aws.amazon.com/general/latest/gr/aws-sec-cred-types.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-iam/>

**Domain**

AWS Security, Identity, & Compliance

**Question 58Incorrect**

A Developer is creating a social networking app for games that uses a single Amazon DynamoDB table. All users’ saved game data is stored in the single table, but users should not be able to view each other’s data.

How can the Developer restrict user access so they can only view their own data?

**Use separate access keys for each user to call the API and restrict access to specific items based on access key ID**

**Read records from DynamoDB and discard irrelevant data client-side**

**Your answer is incorrect**

**Use an identity-based policy that restricts read access to the table to specific principals**

**Correct answer**

**Restrict access to specific items based on certain primary key values**

Overall explanation

In DynamoDB, you have the option to specify conditions when granting permissions using an IAM policy. For example, you can:

• Grant permissions to allow users read-only access to certain items and attributes in a table or a secondary index.

• Grant permissions to allow users write-only access to certain attributes in a table, based upon the identity of that user.

To implement this kind of fine-grained access control, you write an IAM permissions policy that specifies conditions for accessing security credentials and the associated permissions. You then apply the policy to IAM users, groups, or roles that you create using the IAM console. Your IAM policy can restrict access to individual items in a table, access to the attributes in those items, or both at the same time.

You use the IAM Condition element to implement a fine-grained access control policy. By adding a Condition element to a permissions policy, you can allow or deny access to items and attributes in DynamoDB tables and indexes, based upon your particular business requirements. You can also grant permissions on a table, but restrict access to specific items in that table based on certain primary key values.

**CORRECT:**"Restrict access to specific items based on certain primary key values" is the correct answer.

**INCORRECT:** "Use separate access keys for each user to call the API and restrict access to specific items based on access key ID" is incorrect. You cannot restrict access based on access key ID.

**INCORRECT:** "Use an identity-based policy that restricts read access to the table to specific principals" is incorrect as this would only restrict read access to the entire table, not to specific items in the table.

**INCORRECT:** "Read records from DynamoDB and discard irrelevant data client-side" is incorrect as this is inefficient and insecure as it will use more RCUs and has more potential to leak the information.

**References:**

<https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/specifying-conditions.html>

<https://docs.aws.amazon.com/amazondynamodb/latest/developerguide/using-identity-based-policies.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-dynamodb/>

**Domain**

AWS Database

**Question 59Correct**

A mobile application has thousands of users. Each user may use multiple devices to access the application. The Developer wants to assign unique identifiers to these users regardless of the device they use.

Which of the below is the BEST method to obtain unique identifiers?

**Use IAM-generated access key IDs for the users as the unique identifier, but do not store secret keys**

**Your answer is correct**

**Implement developer-authenticated identities by using Amazon Cognito and get credentials for these identities**

**Assign IAM users and roles to the users. Use the unique IAM resource ID as the unique identifier**

**Create a user table in Amazon DynamoDB with key-value pairs of users and their devices. Use these keys as unique identifiers**

Overall explanation

Amazon Cognito lets you add user sign-up, sign-in, and access control to your web and mobile apps quickly and easily. Amazon Cognito provides authentication, authorization, and user management for your web and mobile apps.

Amazon Cognito identity pools enable you to create unique identities for your users and authenticate them with identity providers. With an identity, you can obtain temporary, limited-privilege AWS credentials to access other AWS services.

Amazon Cognito supports developer authenticated identities, in addition to web identity federation through Facebook (Identity Pools), Google (Identity Pools), and Login with Amazon (Identity Pools).

With developer authenticated identities, you can register and authenticate users via your own existing authentication process, while still using Amazon Cognito to synchronize user data and access AWS resources. Using developer authenticated identities involves interaction between the end user device, your backend for authentication, and Amazon Cognito.

In this scenario, this would be the best method of obtaining unique identifiers for each user. This is natively supported through Amazon Cognito.

**CORRECT:**"Implement developer-authenticated identities by using Amazon Cognito and get credentials for these identities" is the correct answer.

**INCORRECT:** "Create a user table in Amazon DynamoDB with key-value pairs of users and their devices. Use these keys as unique identifiers" is incorrect. This is not the best method of implementing this requirement as it requires more custom implementation and management.

**INCORRECT:** "Use IAM-generated access key IDs for the users as the unique identifier, but do not store secret keys" is incorrect. As this is a mobile application it is a good use case for Amazon Cognito so authentication can be handled without needing to create IAM users.

**INCORRECT:** "Assign IAM users and roles to the users. Use the unique IAM resource ID as the unique identifier" is incorrect as this mobile application is a good use case for Amazon Cognito. With Cognito the authentication can be handled using identities in Cognito itself or a federated identity provider. Therefore, the users will not have identities in IAM.

**References:**

<https://docs.aws.amazon.com/cognito/latest/developerguide/developer-authenticated-identities.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-cognito/>

**Domain**

AWS Security, Identity, & Compliance

**Question 60Correct**

A Developer has lost their access key ID and secret access key for programmatic access. What should the Developer do?

**Reset the AWS account access keys**

**Generate a new key pair from the EC2 management console**

**Your answer is correct**

**Disable and delete the user's access key and generate a new set**

**Contact AWS support and request a password reset**

Overall explanation

Access keys consist of two parts:

**The access key identifier**. This is not a secret, and can be seen in the IAM console wherever access keys are listed, such as on the user summary page.

**The secret access key**. This is provided when you initially create the access key pair. Just like a password, it **cannot be retrieved later**. If you lost your secret access key, then you must create a new access key pair. If you already have the [maximum number of access keys](https://docs.aws.amazon.com/IAM/latest/UserGuide/reference_iam-limits.html#reference_iam-limits-entities), you must delete an existing pair before you can create another.

Therefore, the Developer should disable and delete their access keys and generate a new set.

**CORRECT:**"Disable and delete the users’ access key and generate a new set" is the correct answer.

**INCORRECT:** "Contact AWS support and request a password reset" is incorrect as a user name and password are used for console access, not programmatic access.

**INCORRECT:** "Generate a new key pair from the EC2 management console" is incorrect as a key pair is used for accessing EC2 instances, not for programmatic access to work with AWS services.

**INCORRECT:** "Reset the AWS account access keys" is incorrect as these are the access keys associated with the root account rather than the users’ individual IAM account.

**References:**

<https://docs.aws.amazon.com/IAM/latest/UserGuide/troubleshoot_general.html#troubleshoot_general_access-keys>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-iam/>

**Domain**

AWS Security, Identity, & Compliance

**Question 61Correct**

An application will generate thumbnails from objects uploaded to an Amazon S3 bucket. The Developer has created the bucket configuration and the AWS Lambda function and has formulated the following AWS CLI command:

aws lambda add-permission --function-name CreateThumbnail --principal s3.amazonaws.com --statement-id s3invoke --action "lambda:InvokeFunction" --source-arn arn:aws:s3:::digitalcloudbucket-source --source-account 523107438921

What will be achieved by running the AWS CLI command?

**A Lambda function will be created called CreateThumbnail with an Amazon SNS event source mapping that executes the function when objects are uploaded**

**Your answer is correct**

**The Amazon S3 service principal (s3.amazonaws.com) will be granted permissions to perform the lambda:InvokeFunction action**

**The Amazon S3 service principal (s3.amazonaws.com) will be granted permissions to perform the create an event-source mapping with the digitalcloudbucket-source bucket**

**The Lambda function CreateThumbnail will be granted permissions to access the objects in the digitalcloudbucket-source bucket**

Overall explanation

In this scenario the Developer is using an AWS Lambda function to process images that are uploaded to an Amazon S3 bucket. The AWS Lambda function has been created and the notification settings on the bucket have been configured. The last thing to do is to grant permissions for the Amazon S3 service principal to invoke the function.

The Lambda CLI add-permission command grants the Amazon S3 service principal (s3.amazonaws.com) permissions to perform the lambda:InvokeFunction action.

**CORRECT:**"The Amazon S3 service principal (s3.amazonaws.com) will be granted permissions to perform the lambda:InvokeFunction action" is the correct answer.

**INCORRECT:** "The Lambda function CreateThumbnail will be granted permissions to access the objects in the digitalcloudbucket-source bucket" is incorrect as the CLI command grants S3 the ability to execute the Lambda function.

**INCORRECT:** "The Amazon S3 service principal (s3.amazonaws.com) will be granted permissions to perform the create an event-source mapping with the digitalcloudbucket-source bucket" is incorrect as event source mappings are created with services such as Kinesis, DynamoDB, and SQS.

**INCORRECT:** "A Lambda function will be created called CreateThumbnail with an Amazon SNS event source mapping that executes the function when objects are uploaded" is incorrect. This solution does not use Amazon SNS, the S3 notification invokes the Lambda function directly.

**References:**

<https://docs.aws.amazon.com/lambda/latest/dg/with-s3-example.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-lambda/>

**Domain**

AWS Compute

**Question 62Correct**

A Developer is creating a script to automate the deployment process for a serverless application. The Developer wants to use an existing AWS Serverless Application Model (SAM) template for the application.

What should the Developer use for the project? (Select TWO.)

**Your selection is correct**

**Call aws cloudformation package to create the deployment package. Call aws cloudformation deploy to deploy the package afterward**

**Create a ZIP package locally and call aws serverlessrepo create-application to create the application**

**Call aws s3 cp to upload the AWS SAM template to Amazon S3. Call aws lambda update-function-code to create the application**

**Create a ZIP package and upload it to Amazon S3. Call aws cloudformation create-stack to create the application**

**Your selection is correct**

**Call sam package to create the deployment package. Call sam deploy to deploy the package afterward**

Overall explanation

The AWS Serverless Application Model (SAM) is an open-source framework for building serverless applications. It provides shorthand syntax to express functions, APIs, databases, and event source mappings. With just a few lines per resource, you can define the application you want and model it using YAML. During deployment, SAM transforms and expands the SAM syntax into AWS CloudFormation syntax, enabling you to build serverless applications faster.

To get started with building SAM-based applications, use the AWS SAM CLI. SAM CLI provides a Lambda-like execution environment that lets you locally build, test, and debug applications defined by SAM templates. You can also use the SAM CLI to deploy your applications to AWS.

With the SAM CLI you can package and deploy your source code using two simple commands:

• sam package

• sam deploy

Alternatively, you can use:

• aws cloudformation package

• aws cloudformation deploy

Therefore, the Developer can use either the sam or aws cloudformation CLI commands to package and deploy the serverless application.

**CORRECT:**"Call aws cloudformation package to create the deployment package. Call aws cloudformation deploy to deploy the package afterward" is a correct answer.

**CORRECT:**"Call sam package to create the deployment package. Call sam deploy to deploy the package afterward" is a correct answer.

**INCORRECT:** "Call aws s3 cp to upload the AWS SAM template to Amazon S3. Call aws lambda update-function-code to create the application" is incorrect as this is not how to use a SAM template. With SAM the commands mentioned above must be run.

**INCORRECT:** "Create a ZIP package locally and call aws serverlessrepo create-application to create the application" is incorrect as this is not the correct way to use a SAM template.

**INCORRECT:** "Create a ZIP package and upload it to Amazon S3. Call aws cloudformation create-stack to create the application" is incorrect as this is not required when deploying a SAM template.

**References:**

<https://aws.amazon.com/serverless/sam/>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/aws-sam/>

**Domain**

AWS Management & Governance

**Question 63Correct**

A company use Amazon CloudFront to deliver application content to users around the world. A Developer has made an update to some files in the origin however users have reported that they are still getting the old files.

How can the Developer ensure that the old files are replaced in the cache with the LEAST disruption?

**Disable the CloudFront distribution and enable it again to update all the edge locations**

**Your answer is correct**

**Invalidate the files from the edge caches**

**Create a new origin with the new files and remove the old origin**

**Add code to Lambda@Edge that updates the files in the cache**

Overall explanation

If you need to remove files from CloudFront edge caches before they expire you can invalidate the files from the edge caches. To invalidate files, you can specify either the path for individual files or a path that ends with the \* wildcard, which might apply to one file or to many, as shown in the following examples:

• /images/image1.jpg

• /images/image\*

• /images/\*

You can submit a specified number of invalidation paths each month for free. If you submit more than the allotted number of invalidation paths in a month, you pay a fee for each invalidation path that you submit.

**CORRECT:**"Invalidate the files from the edge caches" is the correct answer.

**INCORRECT:** "Create a new origin with the new files and remove the old origin" is incorrect as this would be more disruptive and costly as the entire cache would need to be updated.

**INCORRECT:** "Disable the CloudFront distribution and enable it again to update all the edge locations" is incorrect as this will cause an outage (disruption) and will not replace files that have not yet expired.

**INCORRECT:** "Add code to Lambda@Edge that updates the files in the cache" is incorrect as there’s no value in running code in Lambda@Edge to update the files. Instead the files in the cache can be invalidated.

**References:**

<https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/Invalidation.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-cloudfront/>

**Domain**

AWS Networking & Content Delivery

**Question 64Correct**

A Development team manage a hybrid cloud environment. They would like to collect system-level metrics from on-premises servers and Amazon EC2 instances. How can the Development team collect this information MOST efficiently?

**Use CloudWatch detailed monitoring for both EC2 instances and on-premises servers**

**Use CloudWatch for monitoring EC2 instances and custom AWS CLI scripts using the put-metric-data API**

**Install the CloudWatch agent on the EC2 instances and use a cron job on the on-premises servers**

**Your answer is correct**

**Install the CloudWatch agent on the on-premises servers and EC2 instances**

Overall explanation

The unified CloudWatch agent can be installed on both on-premises servers and Amazon EC2 instances using multiple operating system versions. It enables you to do the following:

• Collect more system-level metrics from Amazon EC2 instances across operating systems. The metrics can include in-guest metrics, in addition to the metrics for EC2 instances.

• Collect system-level metrics from on-premises servers. These can include servers in a hybrid environment as well as servers not managed by AWS.

• Retrieve custom metrics from your applications or services using the StatsD and collectd protocols.

• Collect logs from Amazon EC2 instances and on-premises servers, running either Linux or Windows Server.

Therefore, the Development team should install the CloudWatch agent on the on-premises servers and EC2 instances. This will allow them to collect system-level metrics from servers and instances across the hybrid cloud environment.

**CORRECT:**"Install the CloudWatch agent on the on-premises servers and EC2 instances" is the correct answer.

**INCORRECT:** "Use CloudWatch for monitoring EC2 instances and custom AWS CLI scripts using the put-metric-data API" is incorrect as this is not the most efficient option as you must write and maintain custom scripts. It is better to use the CloudWatch agent as it provides all the functionality required.

**INCORRECT:** "Install the CloudWatch agent on the EC2 instances and use a cron job on the on-premises servers" is incorrect as the answer does not even specify what the cron job is going to do / use for gathering and sending the data.

**INCORRECT:** "Use CloudWatch detailed monitoring for both EC2 instances and on-premises servers" is incorrect as this would not do anything for the on-premises instances.

**References:**

<https://docs.aws.amazon.com/AmazonCloudWatch/latest/monitoring/Install-CloudWatch-Agent.html>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-cloudwatch/>

**Domain**

AWS Management & Governance

**Question 65Correct**

A company has released a new application on AWS. The company are concerned about security and require a tool that can automatically assess applications for exposure, vulnerabilities, and deviations from best practices.

Which AWS service should they use?

**AWS Secrets Manager**

**AWS WAF**

**AWS Shield**

**Your answer is correct**

**Amazon Inspector**

Overall explanation

Amazon Inspector is an automated security assessment service that helps improve the security and compliance of applications deployed on AWS. Amazon Inspector automatically assesses applications for exposure, vulnerabilities, and deviations from best practices.

After performing an assessment, Amazon Inspector produces a detailed list of security findings prioritized by level of severity. These findings can be reviewed directly or as part of detailed assessment reports which are available via the Amazon Inspector console or API.

**CORRECT:**"Amazon Inspector" is the correct answer.

**INCORRECT:** "AWS Shield" is incorrect as this service is used to protect from distributed denial of service (DDoS) attacks.

**INCORRECT:** "AWS WAF" is incorrect as this is a web application firewall.

**INCORRECT:** "AWS Secrets Manager" is incorrect as this service is used to store secure secrets.

**References:**

<https://aws.amazon.com/inspector/>

**Save time with our AWS cheat sheets:**

<https://digitalcloud.training/amazon-inspector/>

**Domain**

AWS Security, Identity, & Compliance