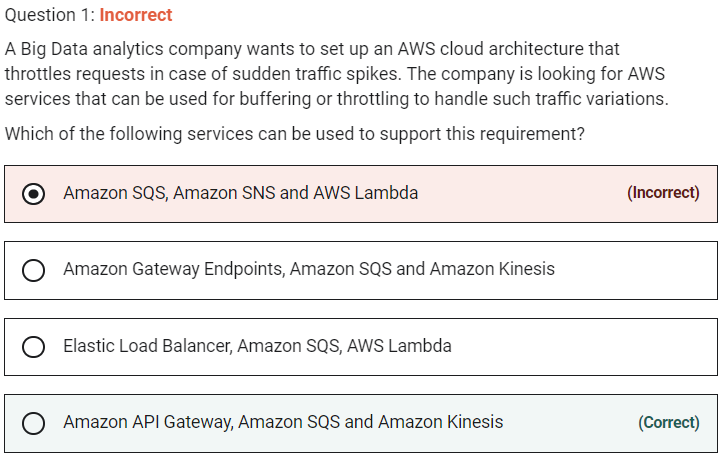
SAA-CO2

Practice Test - 1



Throttling is the process of limiting the number of requests an authorized program can submit to a given operation in a given amount of time.

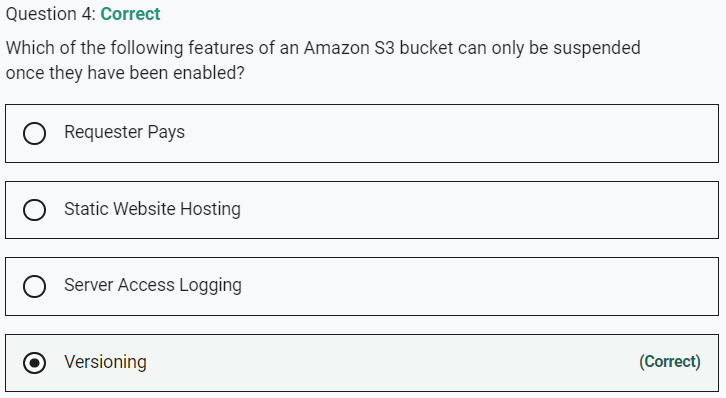
Amazon Simple Notification Service (SNS) cannot buffer messages and is generally used with SQS to provide the buffering facility. AWS Lambda is a compute service and does not provide any buffering capability. So, this combination of services is incorrect.





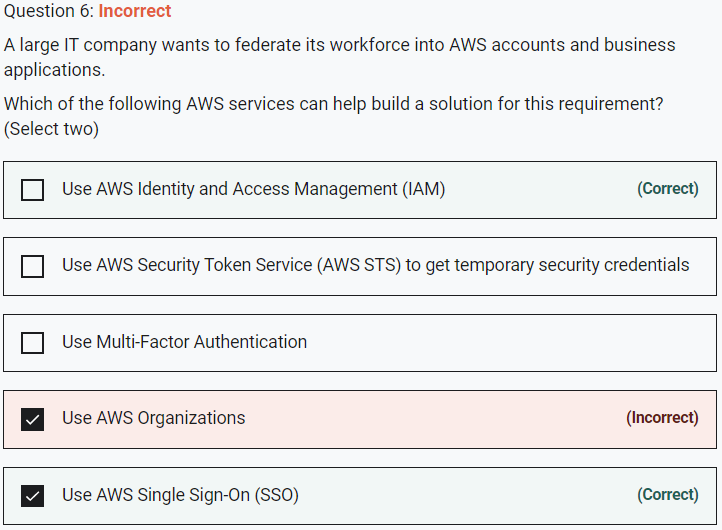
API Gateway supports stateless RESTful APIs as well as stateful WebSocket APIs

**API Gateway creates RESTful APIs that enable stateless client-server communication and API Gateway also creates WebSocket APIs that adhere to the WebSocket protocol, which enables stateful, full-duplex communication between client and server**

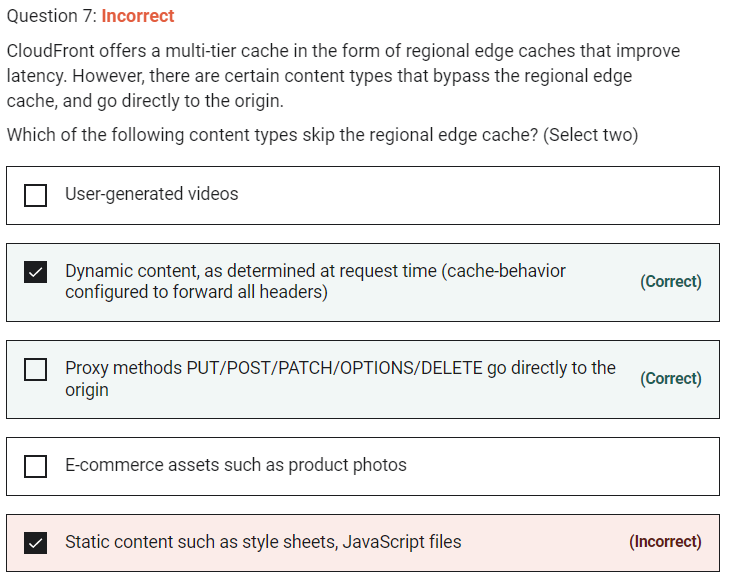


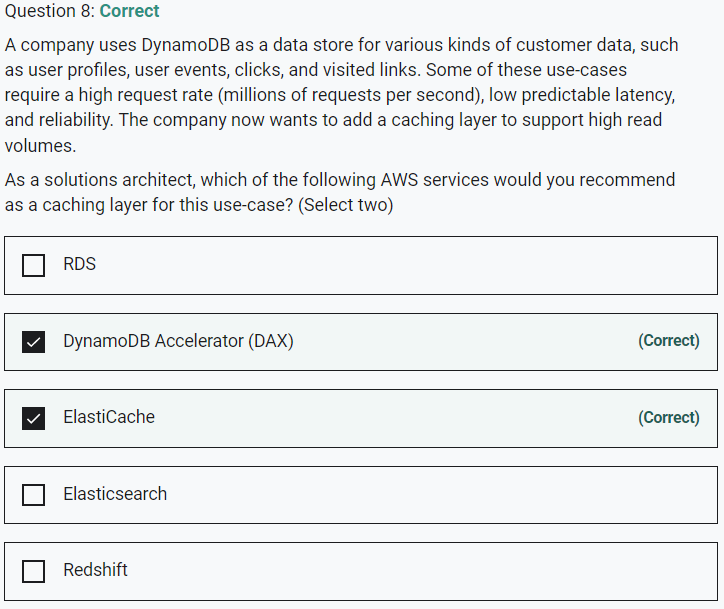


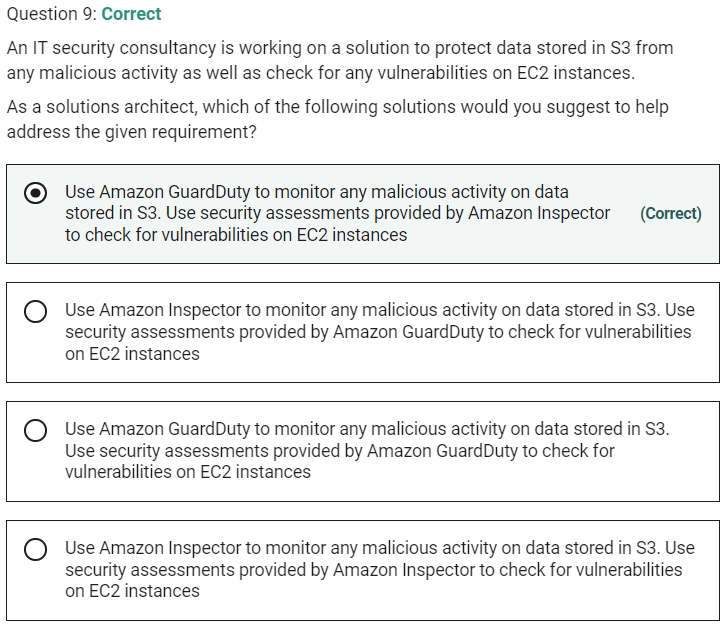
By default, FIFO queues support up to 300 messages per second (300 send, receive, or delete operations per second). When you batch 10 messages per operation (maximum), FIFO queues can support up to 3,000 messages per second. Therefore you need to process 4 messages per operation so that the FIFO queue can support up to 1200 messages per second, which is well within the peak rate.

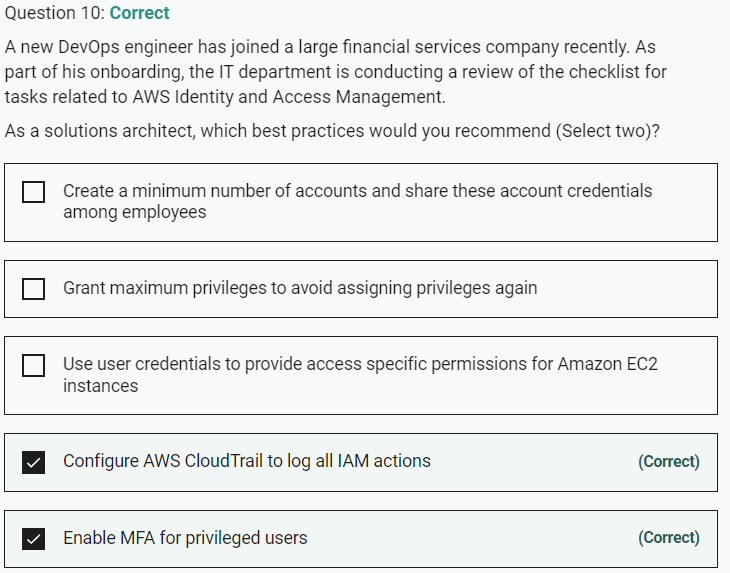


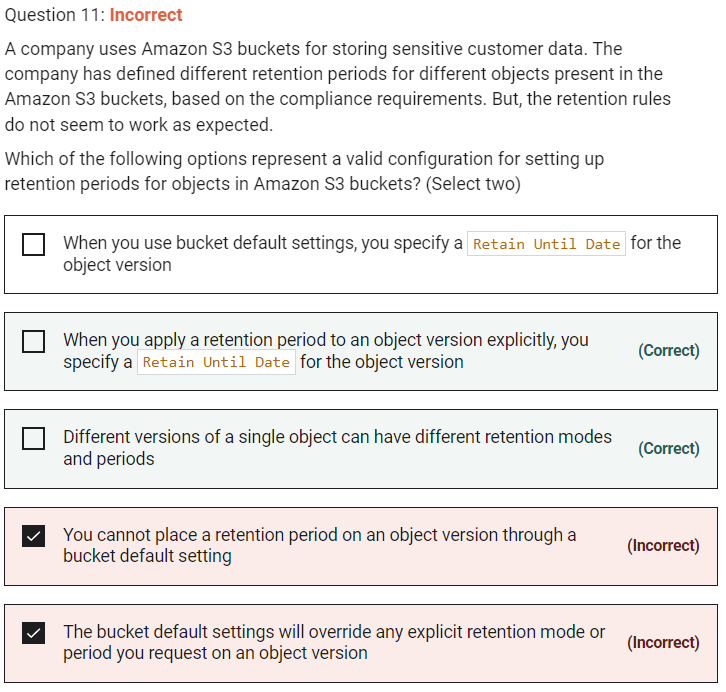
You can use two AWS services to federate your workforce into AWS accounts and business applications: AWS Single Sign-On (SSO) or AWS Identity and Access Management (IAM).











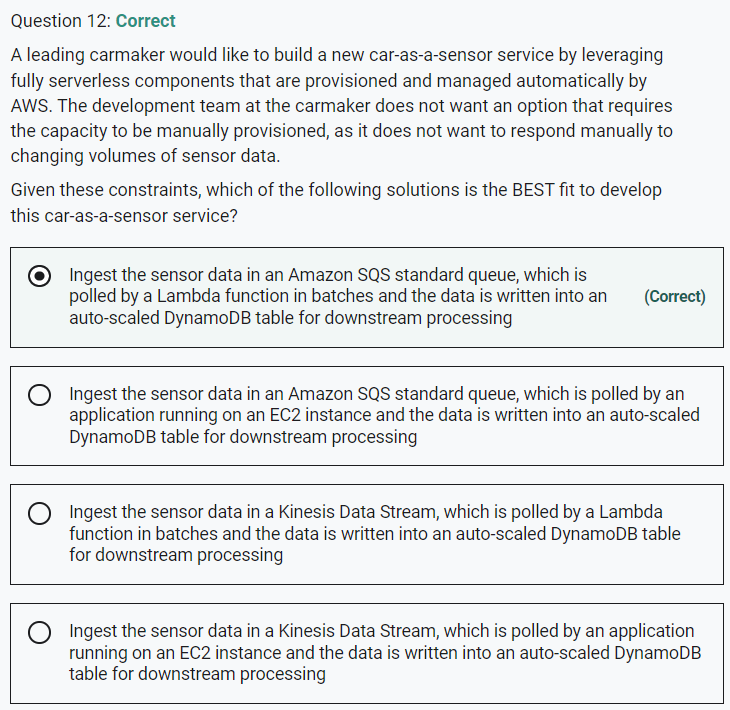
When you apply a retention period to an object version explicitly, you specify a Retain Until Date for the object version - You can place a retention period on an object version either explicitly or through a bucket default setting. When you apply a retention period to an object version explicitly, you specify a Retain Until Date for the object version. Amazon S3 stores the Retain Until Date setting in the object version's metadata and protects the object version until the retention period expires.

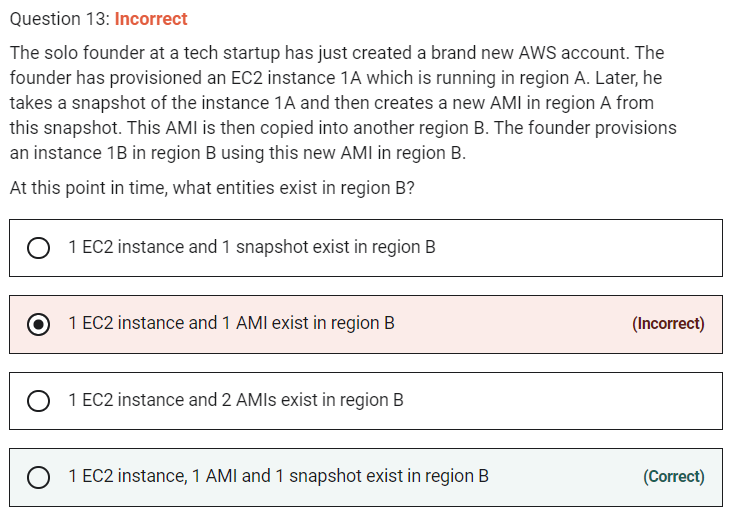
Different versions of a single object can have different retention modes and periods - Like all other Object Lock settings, retention periods apply to individual object versions. Different versions of a single object can have different retention modes and periods.

You cannot place a retention period on an object version through a bucket default setting - You can place a retention period on an object version either explicitly or through a bucket default setting.

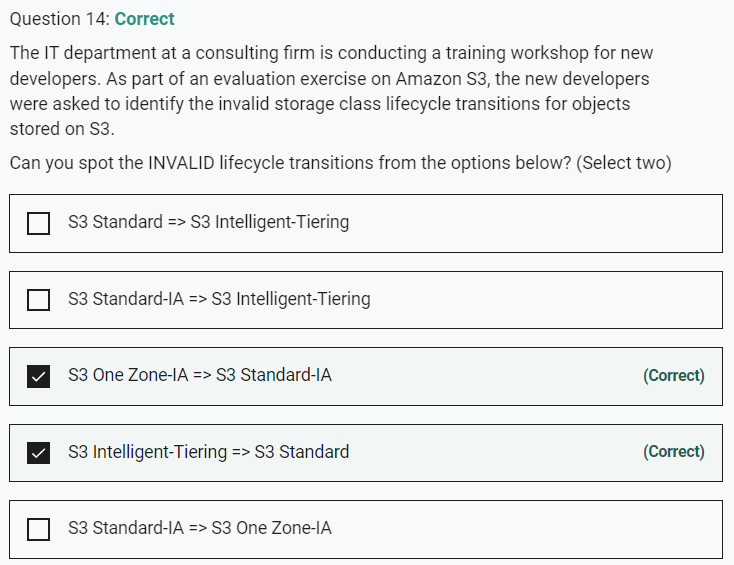
When you use bucket default settings, you specify a Retain Until Date for the object version - When you use bucket default settings, you don't specify a Retain Until Date. Instead, you specify a duration, in either days or years, for which every object version placed in the bucket should be protected.

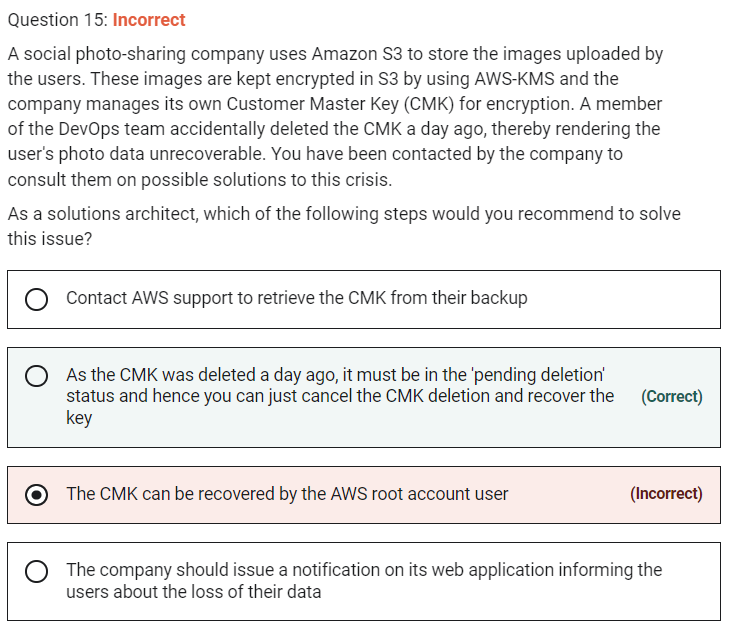
The bucket default settings will override any explicit retention mode or period you request on an object version - If your request to place an object version in a bucket contains an explicit retention mode and period, those settings override any bucket default settings for that object version.



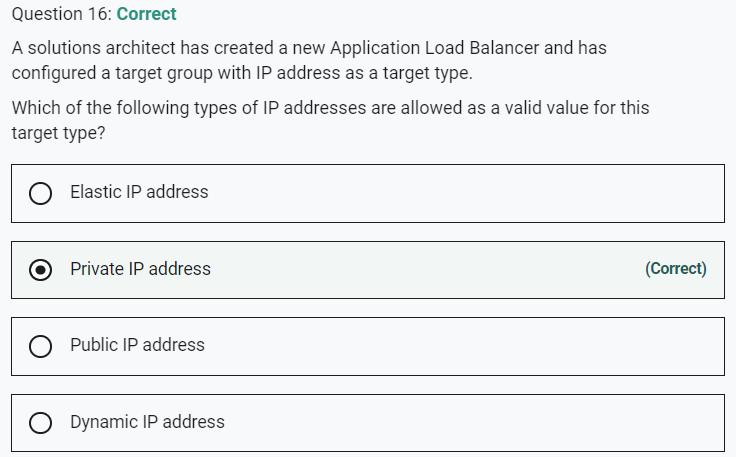


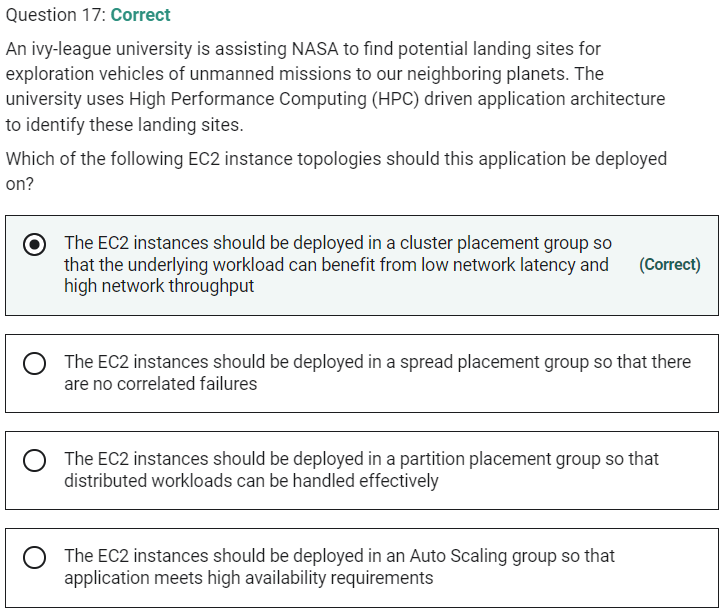
When the new AMI is copied from region A into region B, it automatically creates a snapshot in region B because AMIs are based on the underlying snapshots. Further, an instance is created from this AMI in region B. Hence, we have 1 EC2 instance, 1 AMI and 1 snapshot in region B.

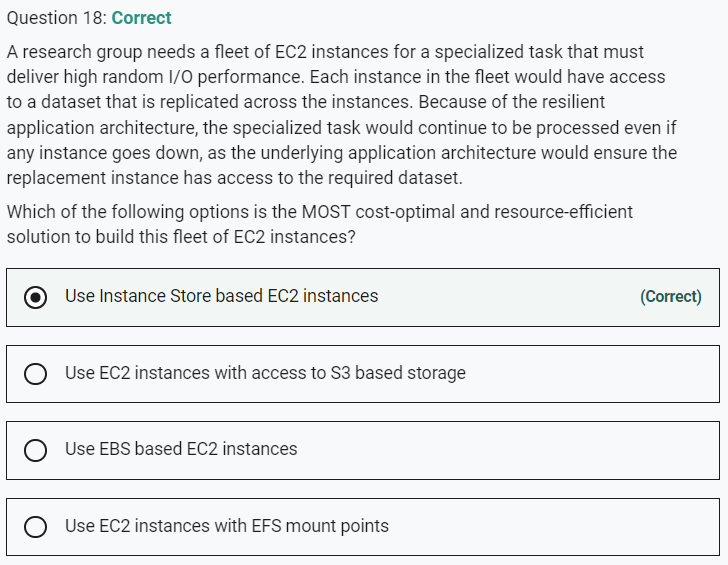


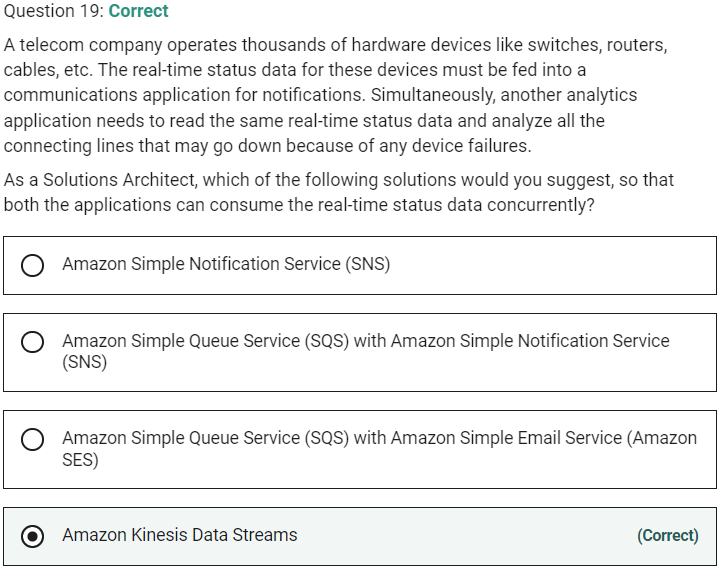


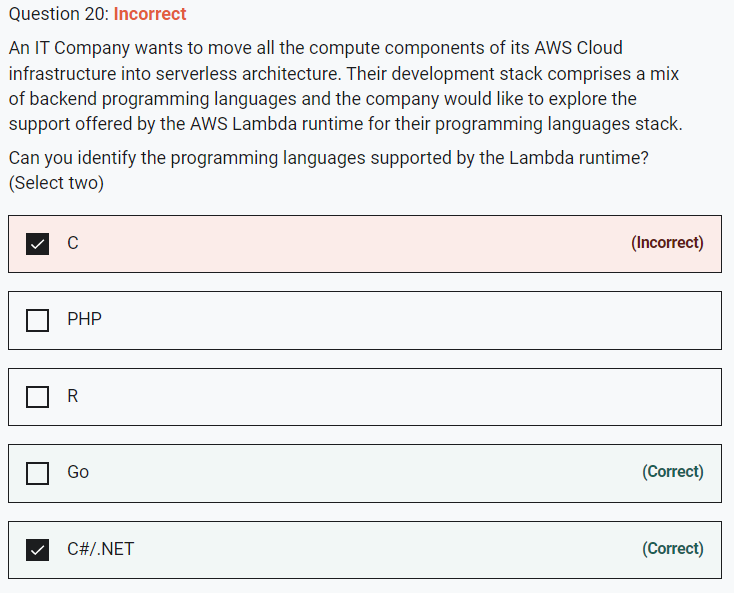
Deleting a customer master key (CMK) in AWS Key Management Service (AWS KMS) is destructive and potentially dangerous. Therefore, AWS KMS enforces a waiting period. To delete a CMK in AWS KMS you schedule key deletion. You can set the waiting period from a minimum of 7 days up to a maximum of 30 days. The default waiting period is 30 days. During the waiting period, the CMK status and key state is Pending deletion. To recover the CMK, you can cancel key deletion before the waiting period ends. After the waiting period ends you cannot cancel key deletion, and AWS KMS deletes the CMK.











AWS Lambda supports runtimes for the following languages:

Python

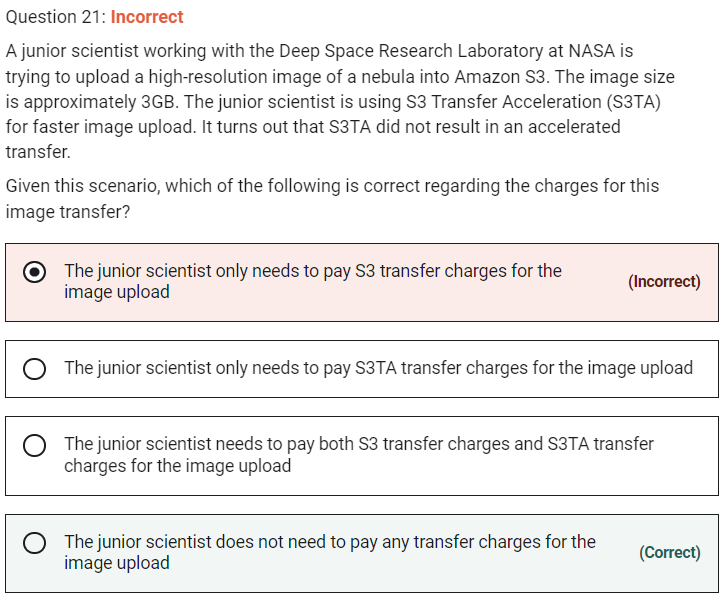
Java

Node.js

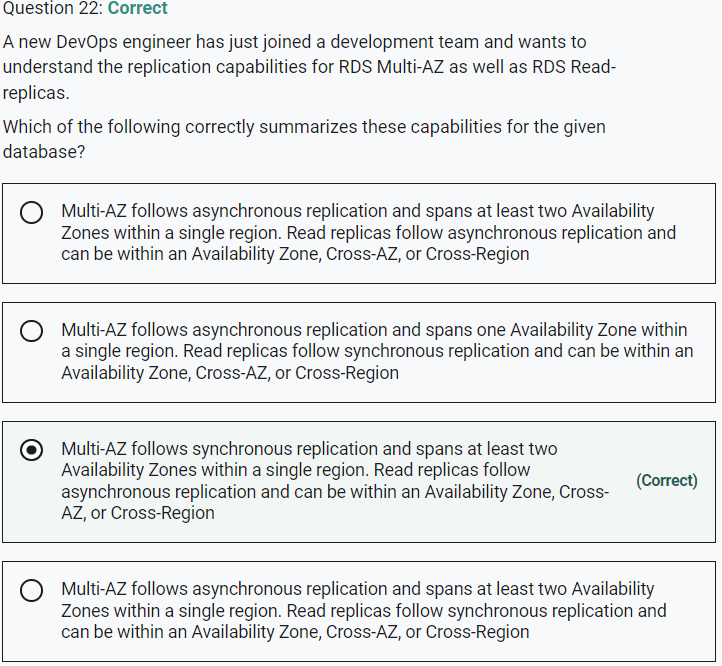
C#/.NET

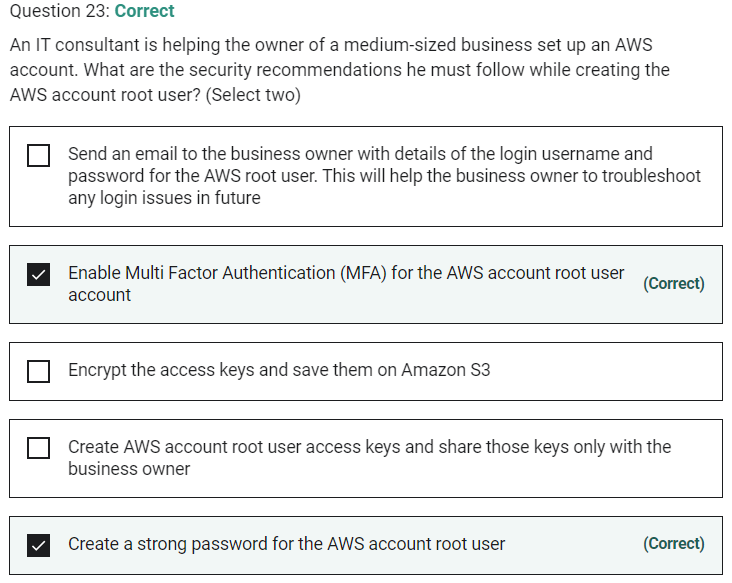
Go

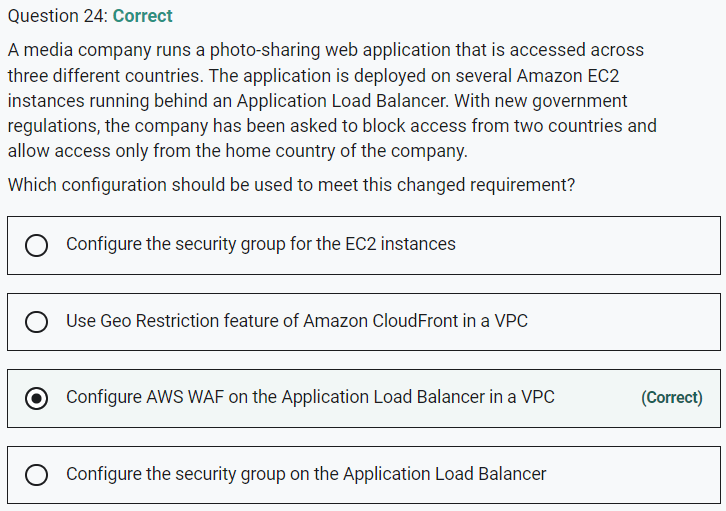
Ruby

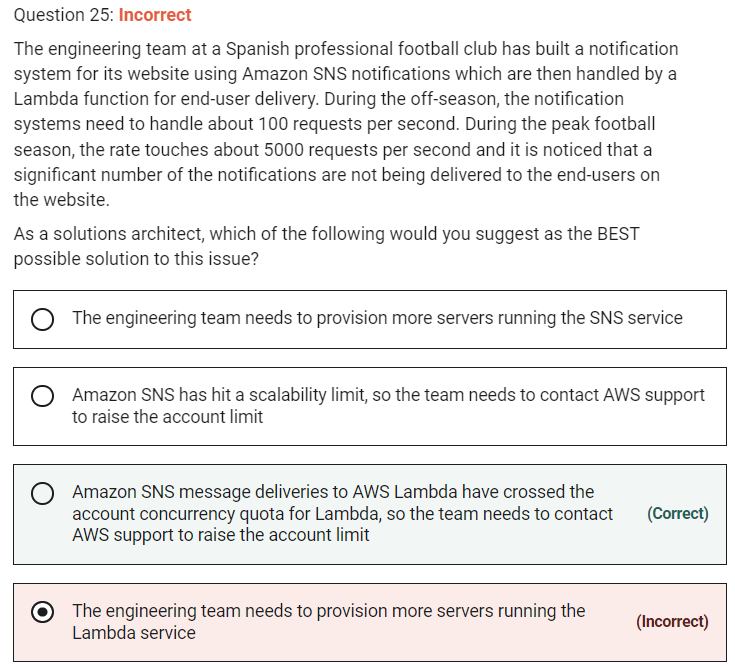


There are no S3 data transfer charges when data is transferred in from the internet. Also with S3TA, you pay only for transfers that are accelerated. Therefore the junior scientist does not need to pay any transfer charges for the image upload because S3TA did not result in an accelerated transfer.

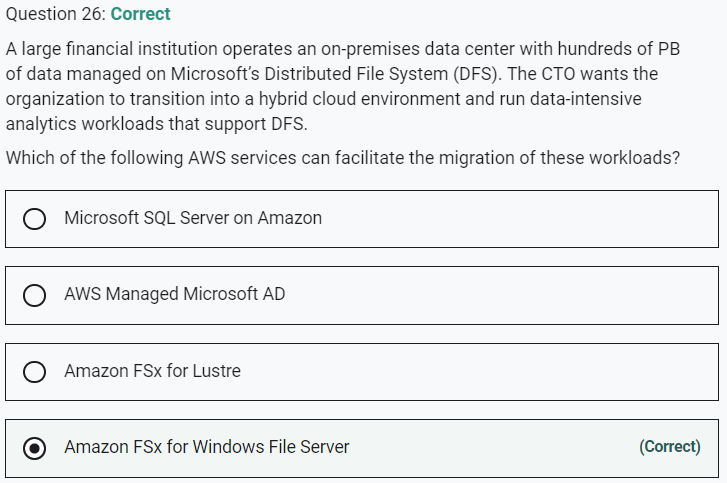








AWS Lambda currently supports 1000 concurrent executions per AWS account per region. If your Amazon SNS message deliveries to AWS Lambda contribute to crossing these concurrency quotas, your Amazon SNS message deliveries will be throttled. You need to contact AWS support to raise the account limit. Therefore this option is correct.



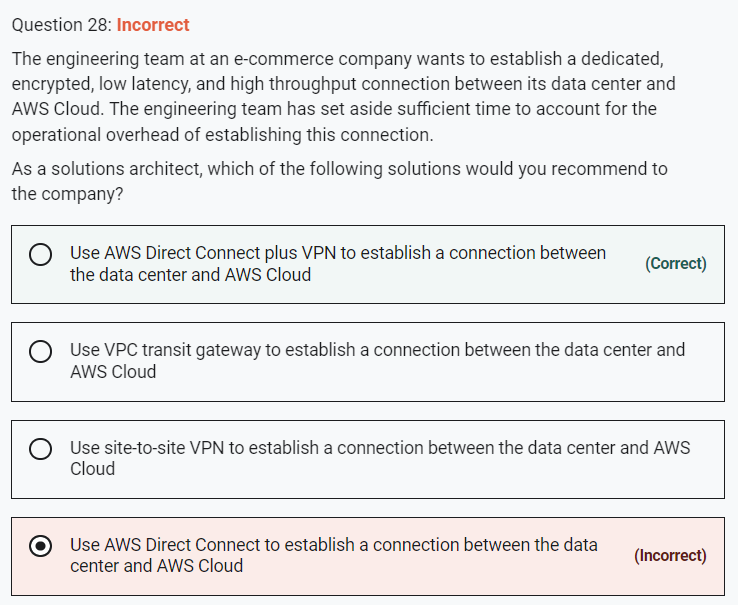
Amazon FSx supports the use of Microsoft’s Distributed File System (DFS) to organize shares into a single folder structure up to hundreds of PB in size

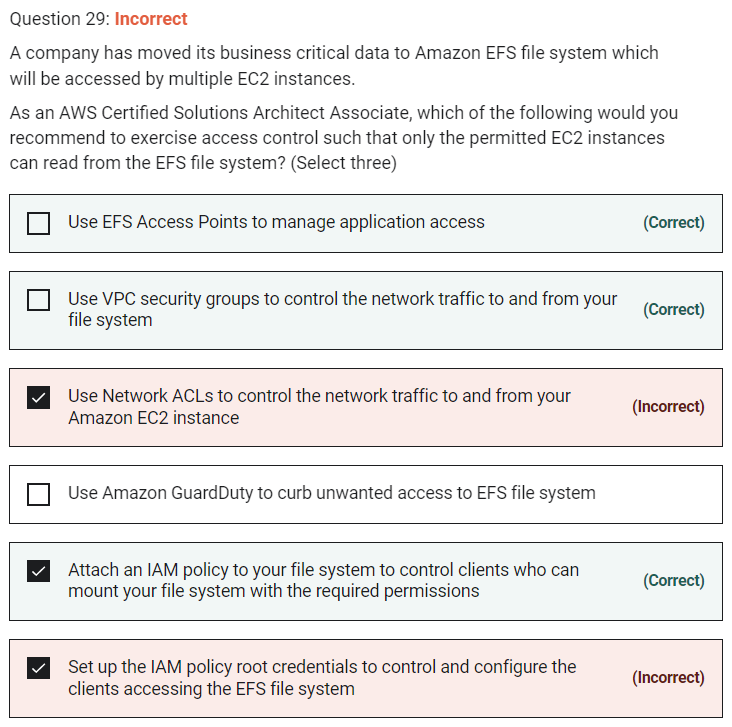


Amazon Elastic File System (Amazon EFS) provides a simple, scalable, fully managed elastic NFS file system for use with AWS Cloud services and on-premises resources.

Amazon EFS is a regional service storing data within and across multiple Availability Zones (AZs) for high availability and durability. Amazon EC2 instances can access your file system across AZs, regions, and VPCs, while on-premises servers can access using AWS Direct Connect or AWS VPN.

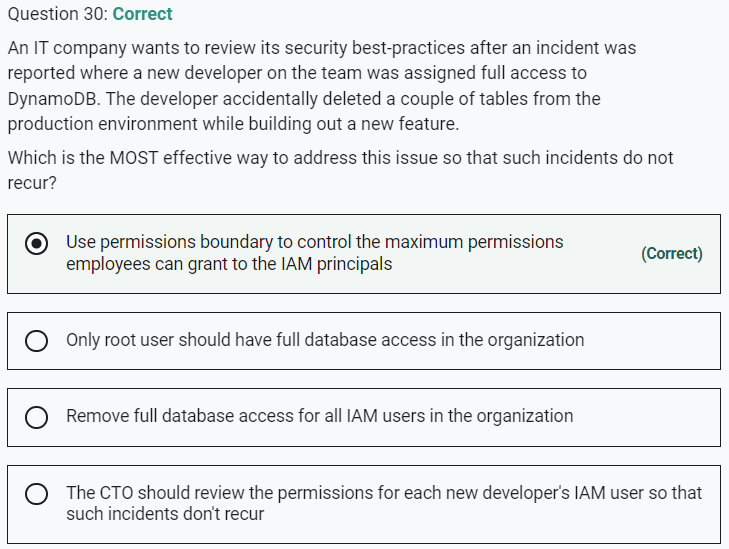
You can connect to Amazon EFS file systems from EC2 instances in other AWS regions using an inter-region VPC peering connection, and from on-premises servers using an AWS VPN connection

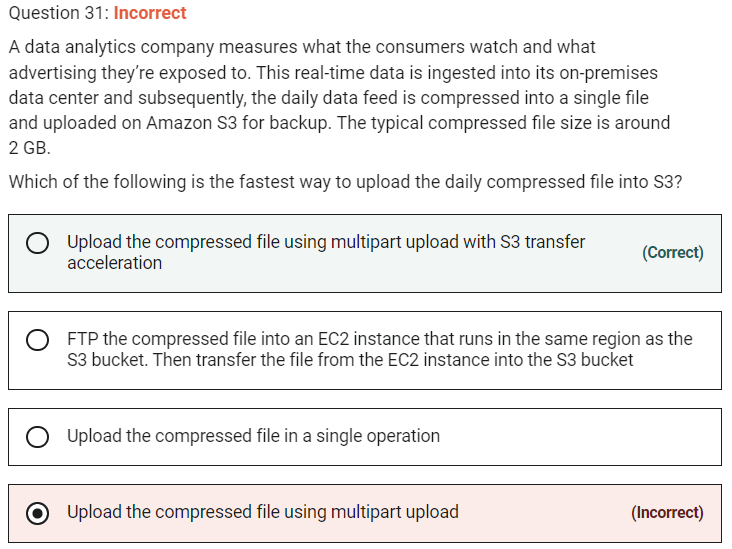




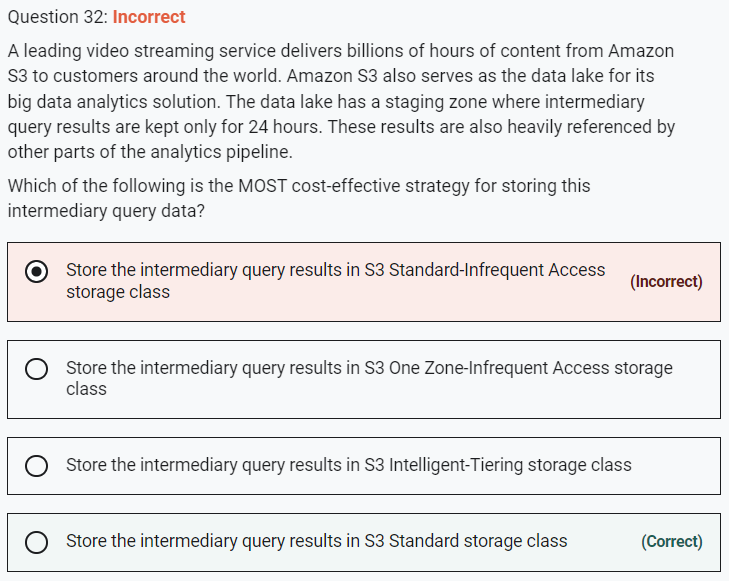
You control which EC2 instances can access your EFS file system by using VPC security group rules and AWS Identity and Access Management (IAM) policies. Use VPC security groups to control the network traffic to and from your file system. Attach an IAM policy to your file system to control which clients can mount your file system and with what permissions, and use EFS Access Points to manage application access. Control access to files and directories with POSIX-compliant user and group-level permissions.

Files and directories in an Amazon EFS file system support standard Unix-style read, write, and execute permissions based on the user ID and group IDs. When an NFS client mounts an EFS file system without using an access point, the user ID and group ID provided by the client is trusted. You can use EFS access points to override user ID and group IDs used by the NFS client. When users attempt to access files and directories, Amazon EFS checks their user IDs and group IDs to verify that each user has permission to access the objects



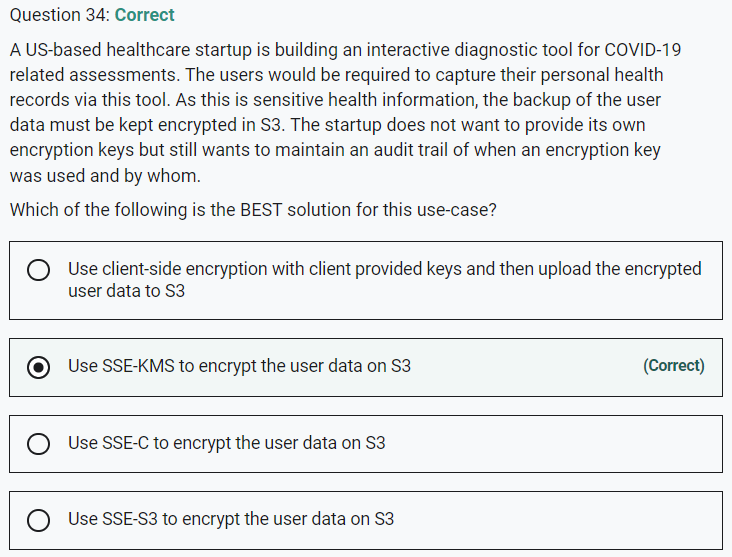


Although using multipart upload would certainly speed up the process, combining with S3 transfer acceleration would further improve the transfer speed. Therefore just using multipart upload is not the correct option.



**Store the intermediary query results in S3 Intelligent-Tiering storage class** - The S3 Intelligent-Tiering storage class is designed to optimize costs by automatically moving data to the most cost-effective access tier, without performance impact or operational overhead. It works by storing objects in two access tiers: one tier that is optimized for frequent access and another lower-cost tier that is optimized for infrequent access. The minimum storage duration charge is 30 days, so this option is NOT cost-effective because intermediary query results need to be kept only for 24 hours. Hence this option is not correct.





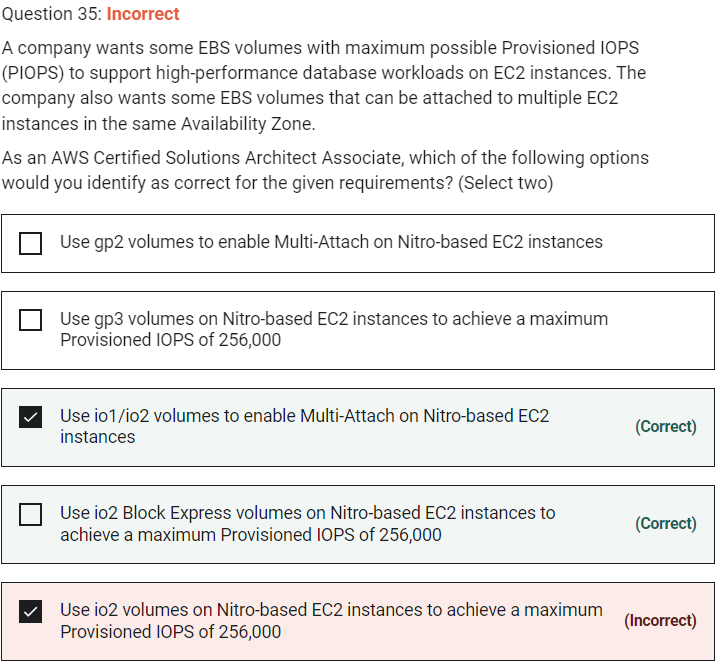
SSE-KMS provides you with an audit trail that shows when your CMK was used and by whom.

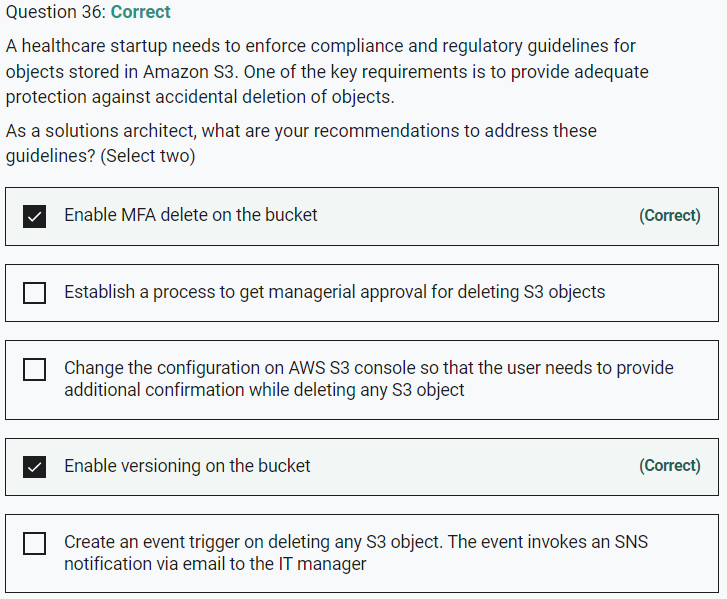
Incorrect options:

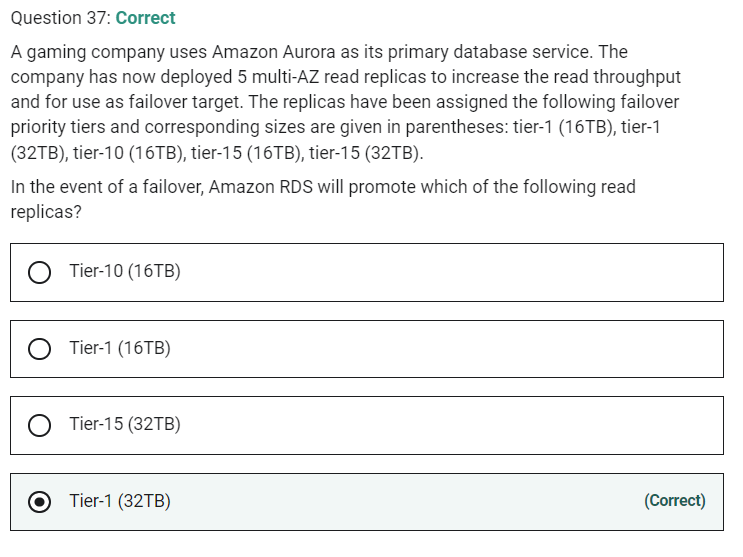
Use SSE-S3 to encrypt the user data on S3 - When you use Server-Side Encryption with Amazon S3-Managed Keys (SSE-S3), each object is encrypted with a unique key. However this option does not provide the ability to audit trail the usage of the encryption keys.

Use SSE-C to encrypt the user data on S3 - With Server-Side Encryption with Customer-Provided Keys (SSE-C), you manage the encryption keys and Amazon S3 manages the encryption, as it writes to disks, and decryption when you access your objects. However this option does not provide the ability to audit trail the usage of the encryption keys.

Use client-side encryption with client provided keys and then upload the encrypted user data to S3 - Using client-side encryption is ruled out as the startup does not want to provide the encryption keys.

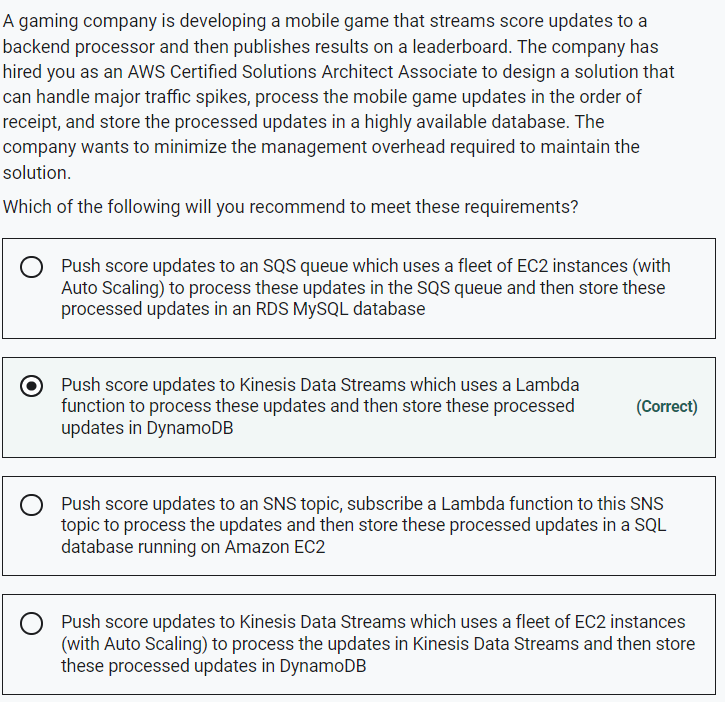


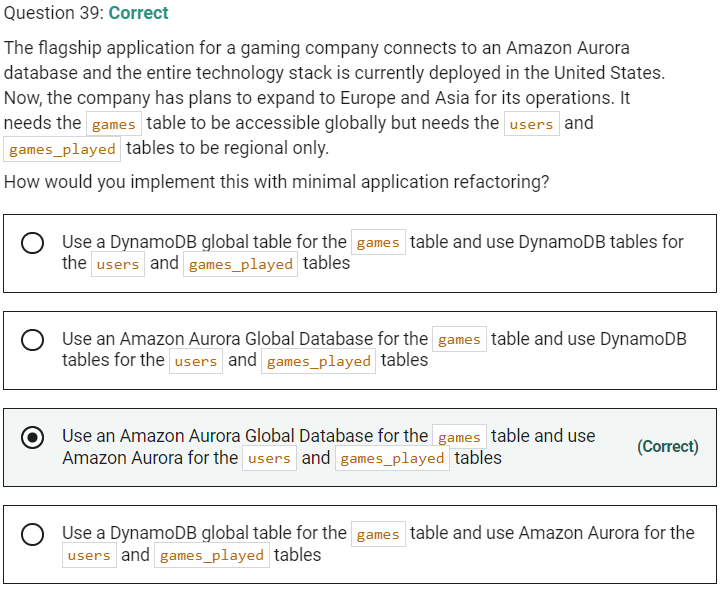


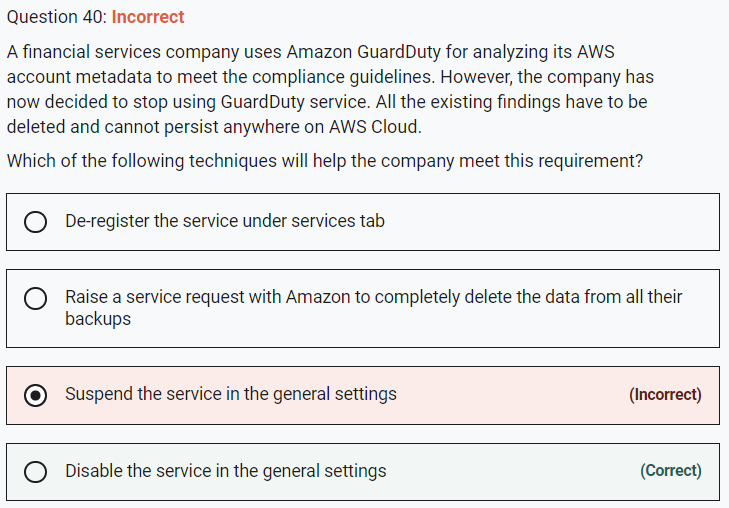


For Amazon Aurora, each Read Replica is associated with a priority tier (0-15). In the event of a failover, Amazon Aurora will promote the Read Replica that has the highest priority (the lowest numbered tier). If two or more Aurora Replicas share the same priority, then Amazon RDS promotes the replica that is largest in size. If two or more Aurora Replicas share the same priority and size, then Amazon Aurora promotes an arbitrary replica in the same promotion tier.

Therefore, for this problem statement, the Tier-1 (32TB) replica will be promoted.







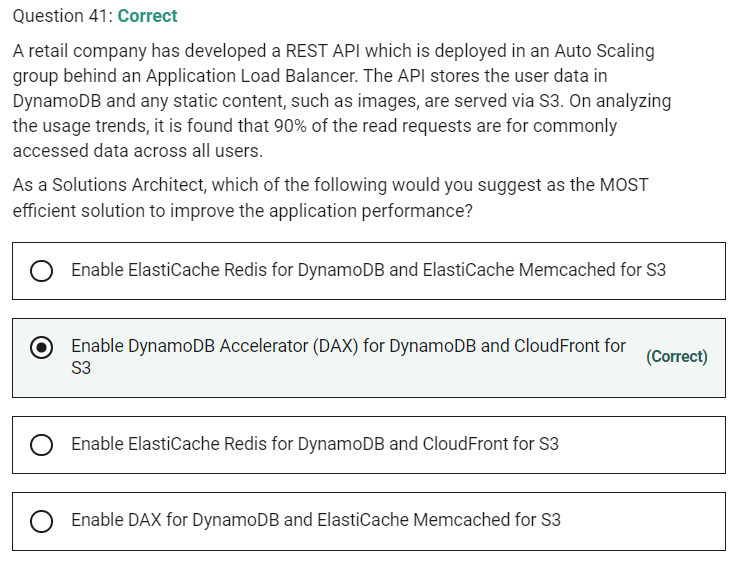
**Disable the service in the general settings** - Disabling the service will delete all remaining data, including your findings and configurations before relinquishing the service permissions and resetting the service. So, this is the correct option for our use case.

Incorrect options:

**Suspend the service in the general settings** - You can stop Amazon GuardDuty from analyzing your data sources at any time by choosing to suspend the service in the general settings. This will immediately stop the service from analyzing data, but does not delete your existing findings or configurations.

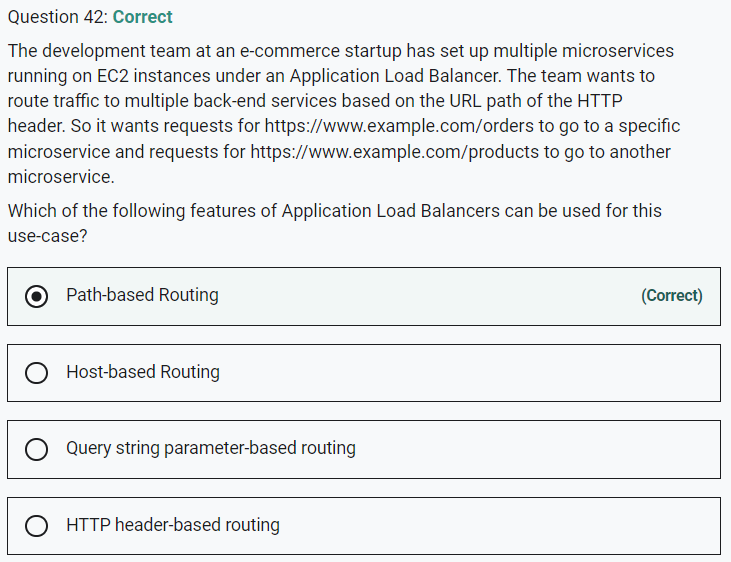
**De-register the service under services tab** - This is a made-up option, used only as a distractor.

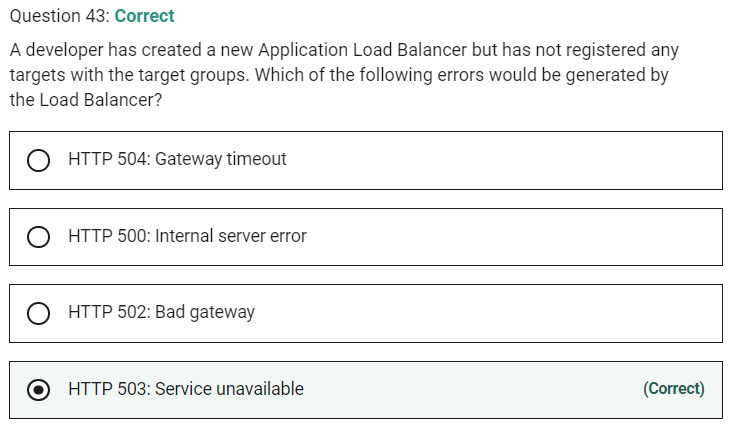
**Raise a service request with Amazon to completely delete the data from all their backups** - There is no need to create a service request as you can delete the existing findings by disabling the service.

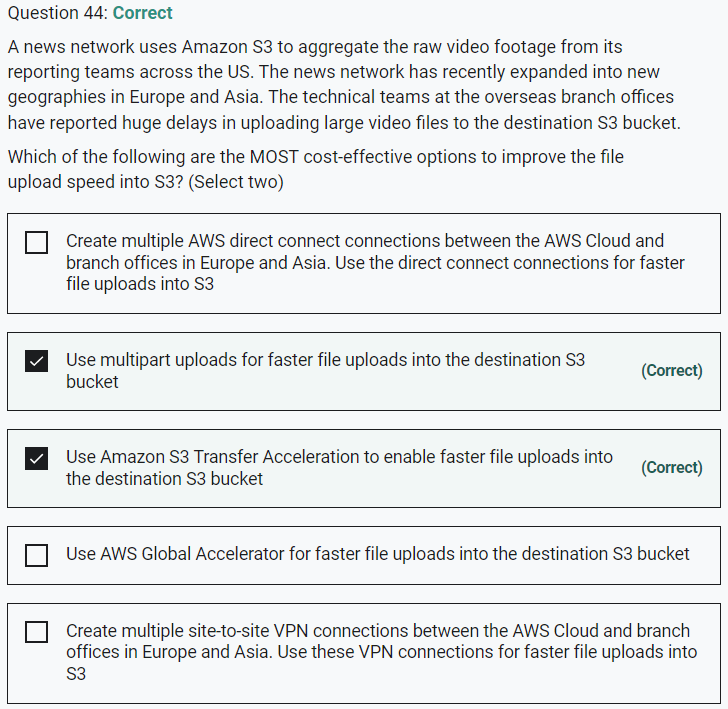


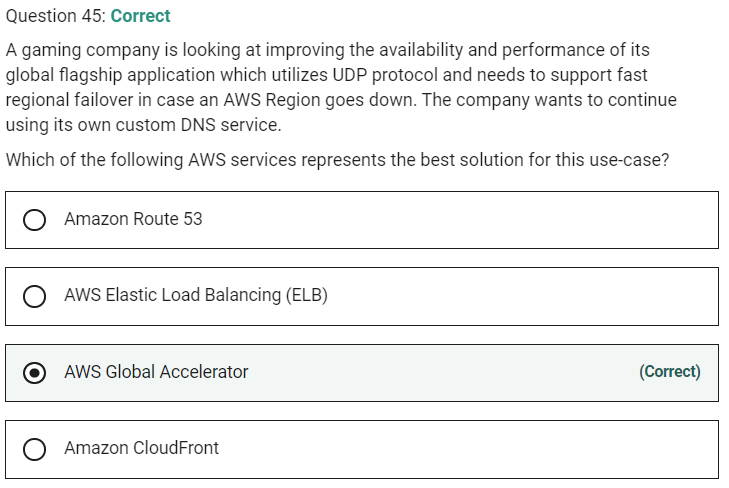
Amazon ElastiCache for Memcached is a great choice for implementing an in-memory cache to decrease access latency, increase throughput, and ease the load off your relational or NoSQL database.

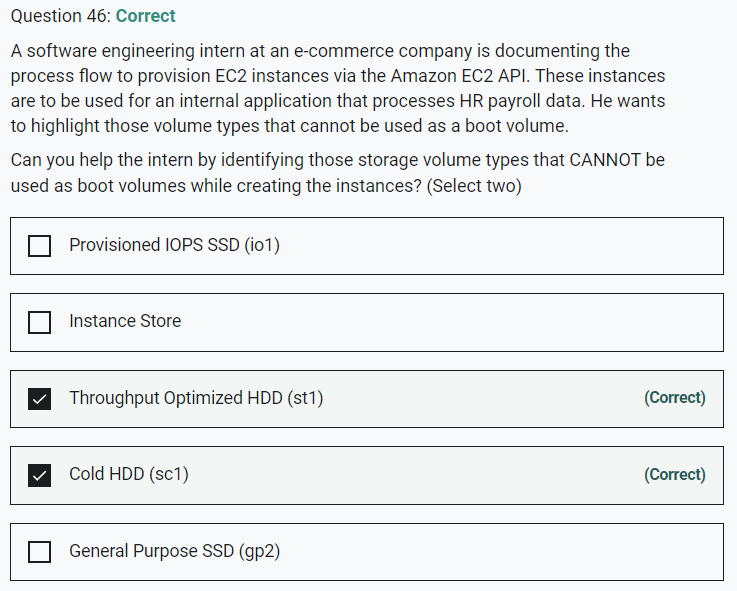
ElastiCache Memcached cannot be used as a cache to serve static content from S3, so both these options are incorrect.

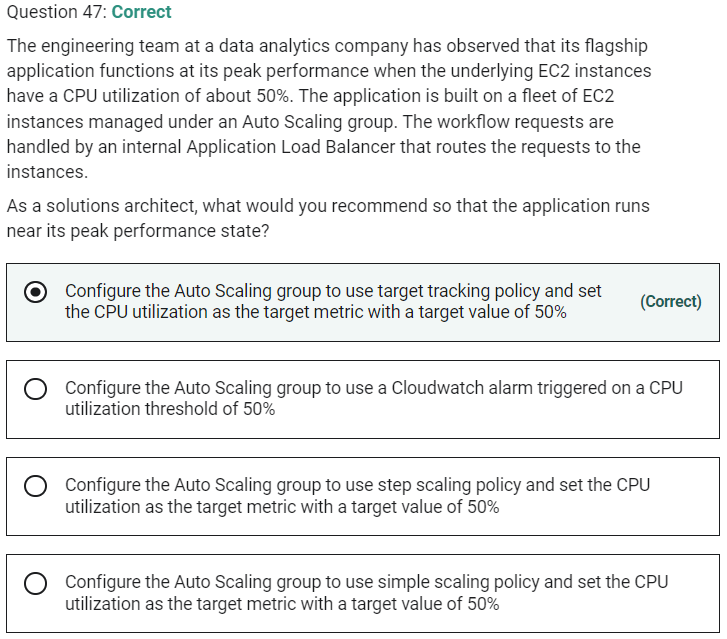


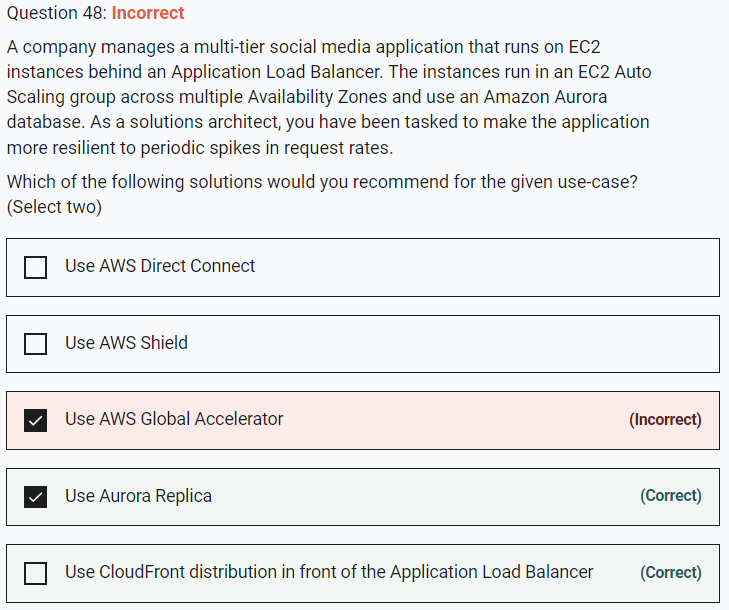












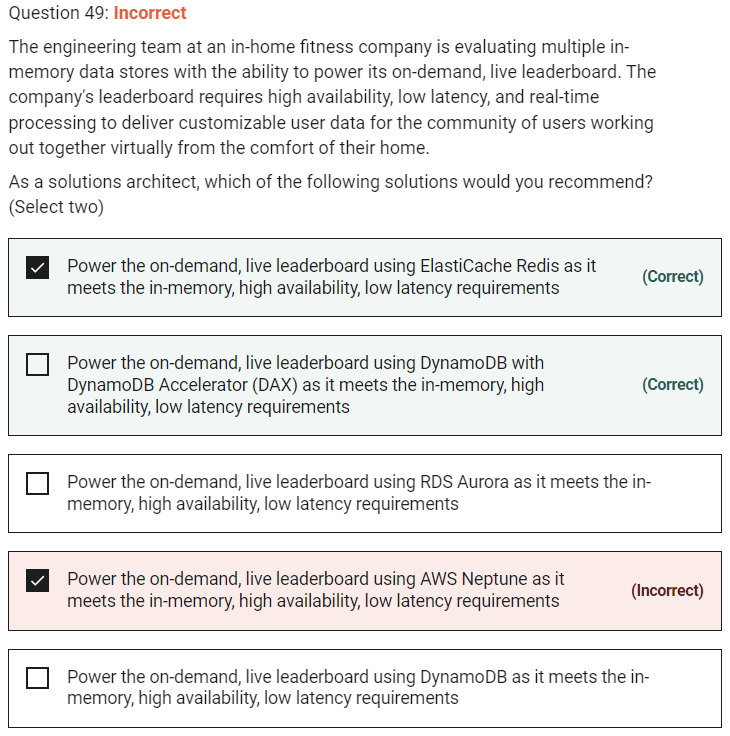
You can use Aurora replicas and CloudFront distribution to make the application more resilient to spikes in request rates.

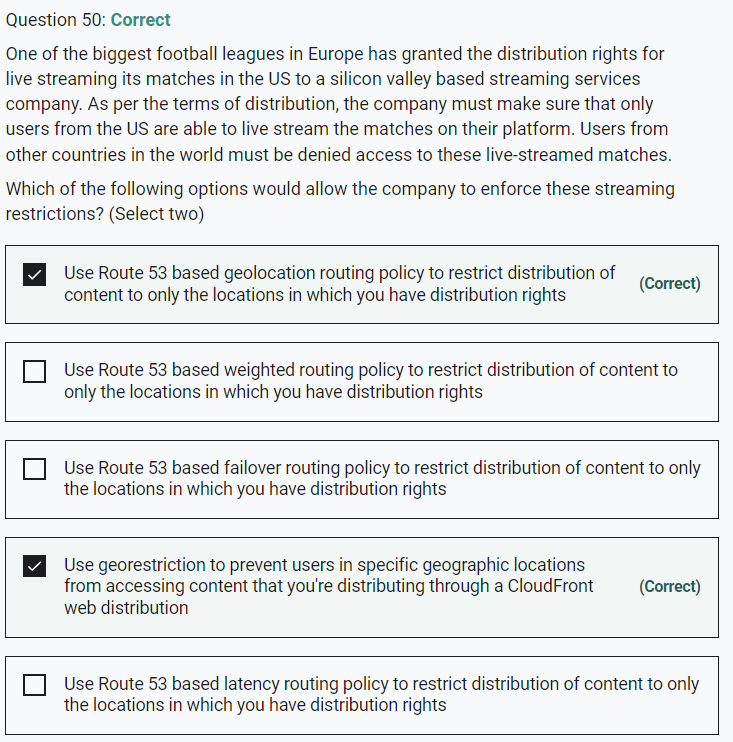
Use Aurora Replica

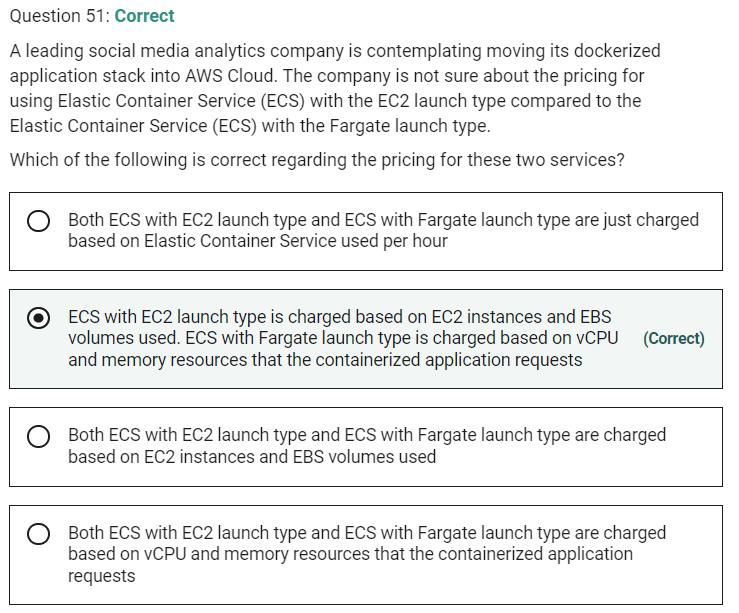
Aurora Replicas have two main purposes. You can issue queries to them to scale the read operations for your application. You typically do so by connecting to the reader endpoint of the cluster. That way, Aurora can spread the load for read-only connections across as many Aurora Replicas as you have in the cluster. Aurora Replicas also help to increase availability. If the writer instance in a cluster becomes unavailable, Aurora automatically promotes one of the reader instances to take its place as the new writer. Up to 15 Aurora Replicas can be distributed across the Availability Zones that a DB cluster spans within an AWS Region.

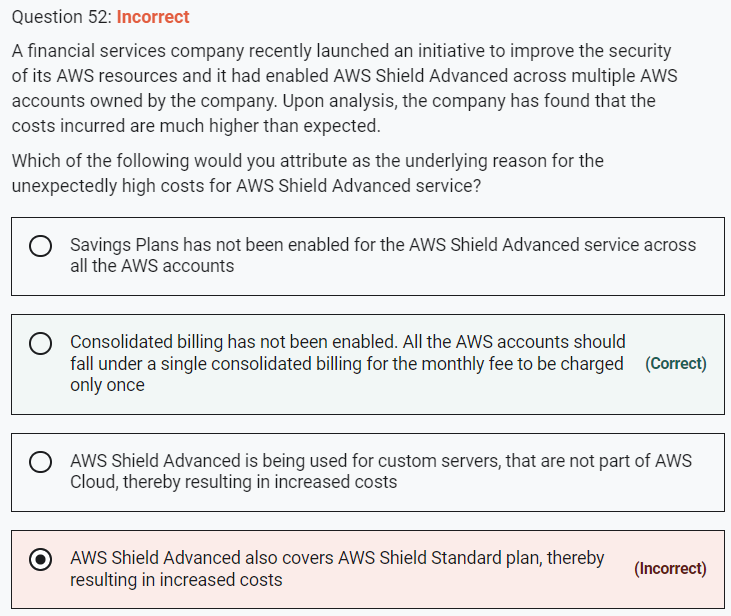
Use CloudFront distribution in front of the Application Load Balancer

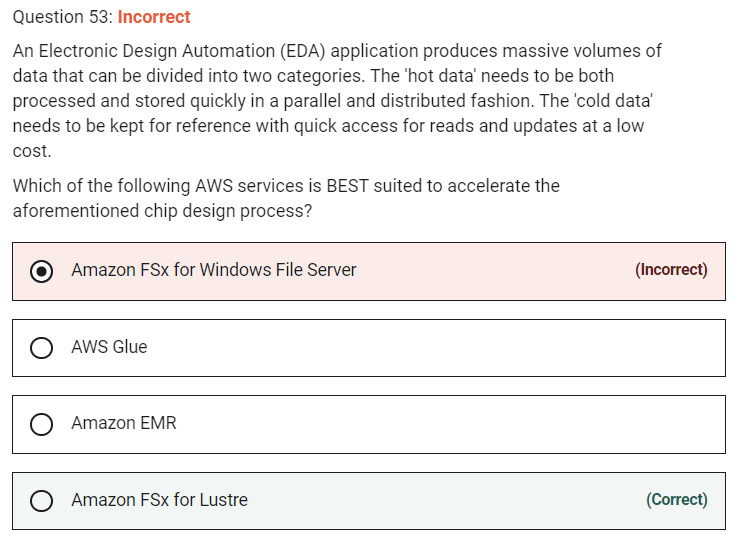
CloudFront offers an origin failover feature to help support your data resiliency needs. CloudFront is a global service that delivers your content through a worldwide network of data centers called edge locations or points of presence (POPs). If your content is not already cached in an edge location, CloudFront retrieves it from an origin that you've identified as the source for the definitive version of the content.

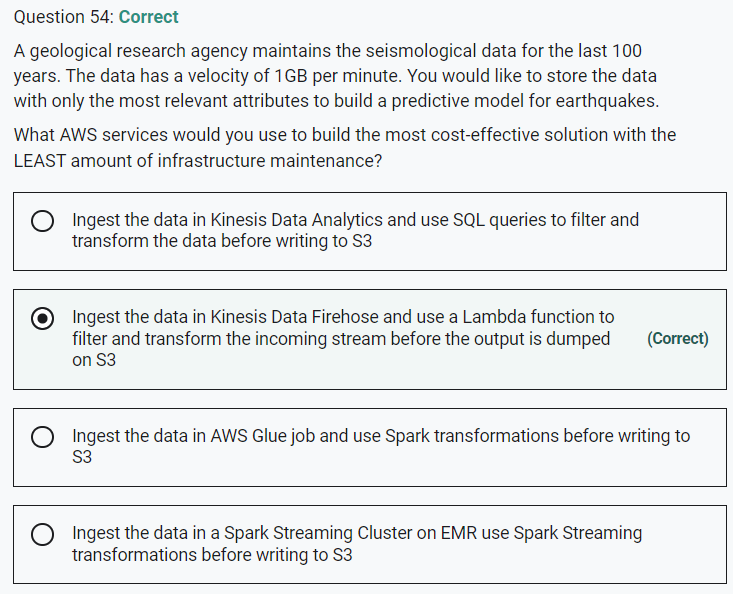








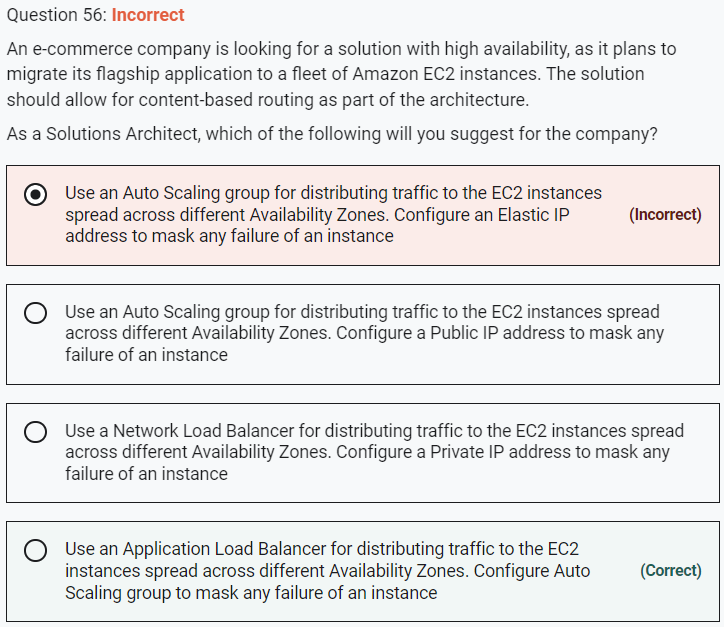




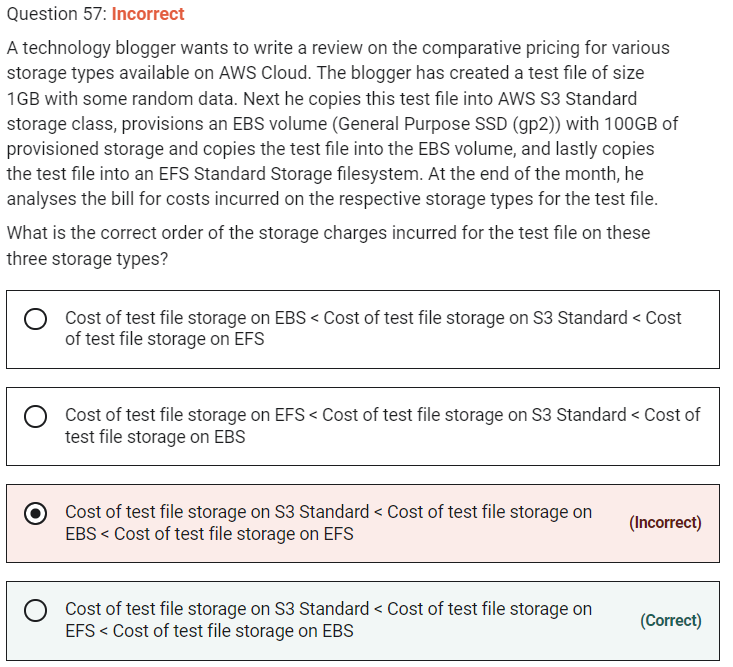
Graphical user interface, application

Description automatically generatedGraphical user interface, application

Description automatically generated



NLB cannot help in content based routing.



Explanation

Correct option:

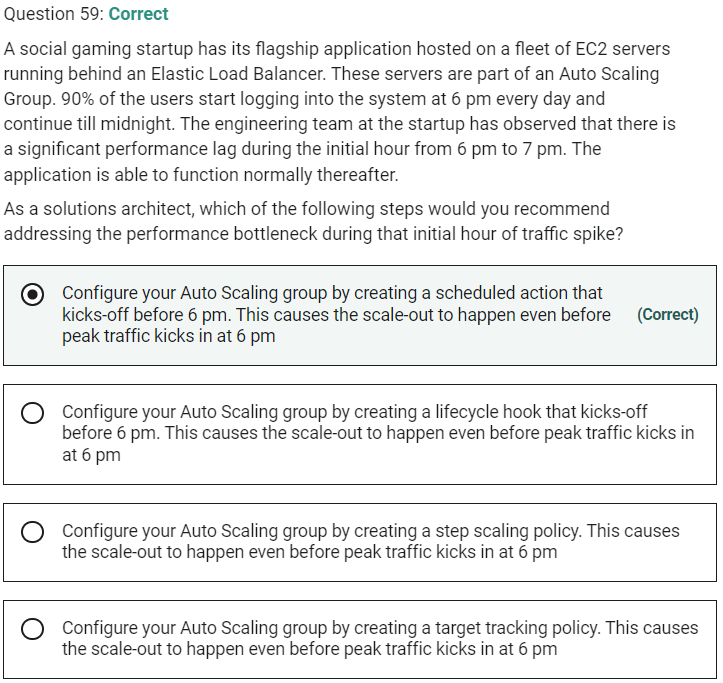
Cost of test file storage on S3 Standard < Cost of test file storage on EFS < Cost of test file storage on EBS

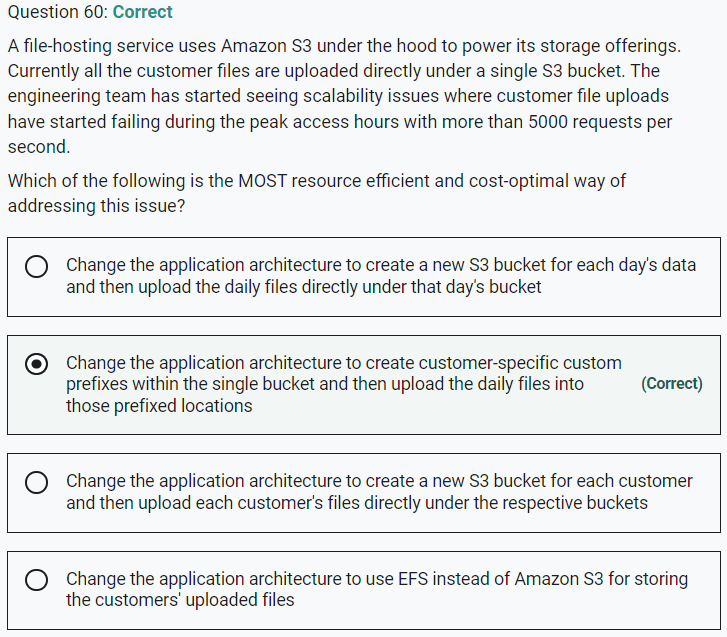
With Amazon EFS, you pay only for the resources that you use. The EFS Standard Storage pricing is $0.30 per GB per month. Therefore the cost for storing the test file on EFS is $0.30 for the month.

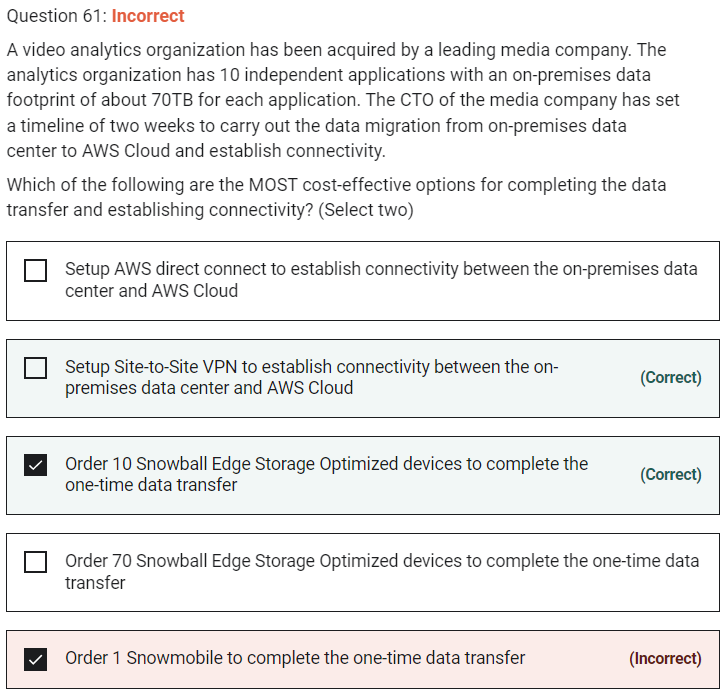
For EBS General Purpose SSD (gp2) volumes, the charges are $0.10 per GB-month of provisioned storage. Therefore, for a provisioned storage of 100GB for this use-case, the monthly cost on EBS is $0.10\*100 = $10. This cost is irrespective of how much storage is actually consumed by the test file.

For S3 Standard storage, the pricing is $0.023 per GB per month. Therefore, the monthly storage cost on S3 for the test file is $0.023.



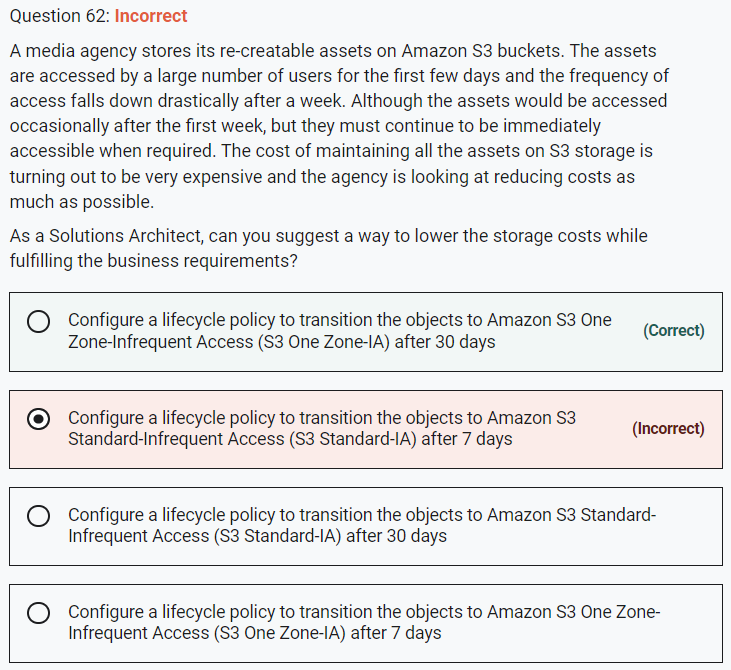




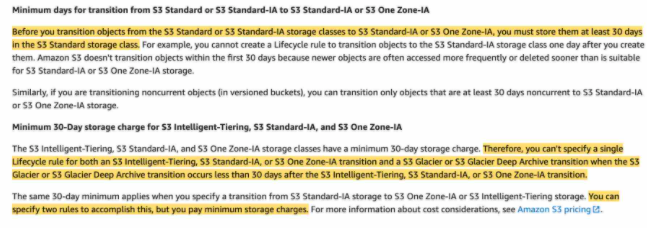


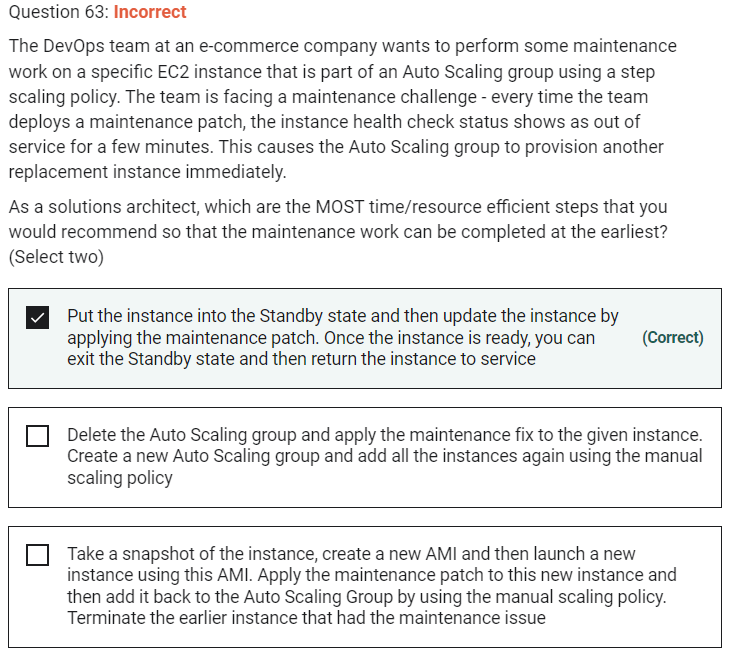
**Order 1 Snowmobile to complete the one-time data transfer** - Each Snowmobile has a total capacity of up to 100 petabytes. To migrate large datasets of 10PB or more in a single location, you should use Snowmobile. For datasets less than 10PB or distributed in multiple locations, you should use Snowball. So Snowmobile is not the right fit for this use-case.

**Order 70 Snowball Edge Storage Optimized devices to complete the one-time data transfer** - As the data-transfer can be completed with just 10 Snowball Edge Storage Optimized devices, there is no need to order 70 devices.



The minimum storage duration is 30 days before you can transition objects from S3 Standard to S3 One Zone-IA.







**Put the instance into the Standby state and then update the instance by applying the maintenance patch. Once the instance is ready, you can exit the Standby state and then return the instance to service** - You can put an instance that is in the InService state into the Standby state, update some software or troubleshoot the instance, and then return the instance to service. Instances that are on standby are still part of the Auto Scaling group, but they do not actively handle application traffic.

