Top SAA-C03 Questions | SAA-C03 Exam Revision Plan

P.S. Free & New SAA-C03 dumps are available on Google Drive shared by PracticeMaterial: https://drive.google.com/open?id=1t76fBf81AV4ETkkn6nmy8zSAlgldr2b1

Amazon certifications have strong authority in this field and are recognized by all companies in most of companies in the whole world. SAA-C03 new test camp questions are the best choice for candidates who are determined to clear exam urgently. If you purchase our SAA-C03 New Test Camp questions to pass this exam, you will make a major step forward for relative certification. Also you can use our products pass the other exams.

To prepare for the Amazon SAA-C03 exam, candidates should have a solid understanding of AWS services and solutions. They should also have experience designing and deploying scalable, highly available, and fault-tolerant systems on AWS. Candidates can prepare for the exam by taking AWS training courses, reading AWS documentation, and practicing with AWS services and solutions in a test environment. Passing the Amazon SAA-C03 Exam is a great achievement and can open up many rewarding career opportunities in the field of cloud computing.

>> Top SAA-C03 Questions <<

SAA-C03 Exam Revision Plan & SAA-C03 New Real Test

We are quite confident that all these Amazon SAA-C03 exam dumps feature you will not find anywhere. Just download the Amazon SAA-C03 and start this journey right now. For the well and Amazon SAA-C03 Exam Dumps preparation, you can get help from Amazon SAA-C03 which will provide you with everything that you need to learn, prepare and pass the AWS Certified Solutions Architect - Associate (SAA-C03) certification exam.

Amazon AWS Certified Solutions Architect - Associate Sample Questions (Q247-Q252):

NEW QUESTION # 247
A popular mobile game uses CloudFront, Lambda, and DynamoDB for its backend services. The player data is persisted on a DynamoDB table and the static assets are distributed by CloudFront.
However, there are a lot of complaints that saving and retrieving player information is taking a lot of time.
To improve the game's performance, which AWS service can you use to reduce DynamoDB response times from milliseconds to

microseconds?

- A. AWS Device Farm
- B. Amazon DynamoDB Accelerator (DAX)
- C. Amazon ElastiCache
- D. DynamoDB Auto Scaling

Answer: B

Explanation:
Amazon DynamoDB Accelerator (DAX) is a fully managed, highly available, in-memory cache that can reduce Amazon DynamoDB response times from milliseconds to microseconds, even at millions of requests per second.

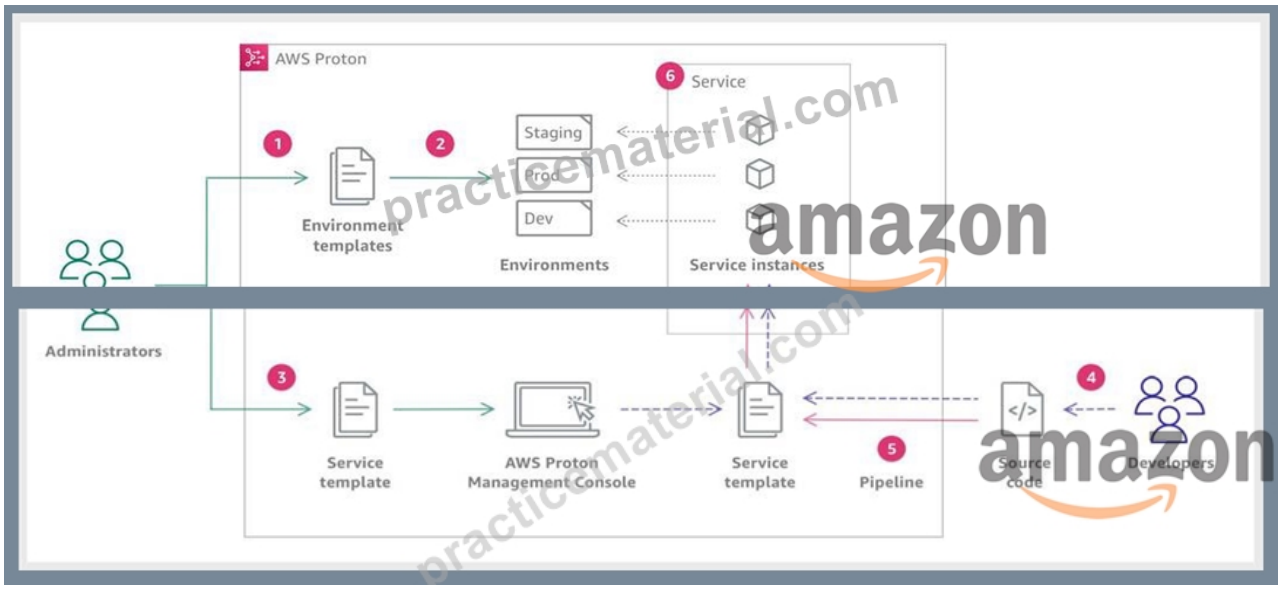
Amazon ElastiCache is incorrect because although you may use ElastiCache as your database cache, it will not reduce the DynamoDB response time from milliseconds to microseconds as compared with DynamoDB DAX.
AWS Device Farm is incorrect because this is an app testing service that lets you test and interact with your Android, iOS, and web apps on many devices at once, or reproduce issues on a device in real time.
DynamoDB Auto Scaling is incorrect because this is primarily used to automate capacity management for your tables and global secondary indexes.
References:
https://aws.amazon.com/dynamodb/dax
https://aws.amazon.com/device-farm
Check out this Amazon DynamoDB Cheat Sheet:
https://tutorialsdojo.com/amazon-dynamodb/

NEW QUESTION # 248
A technology company has a suite of container-based web applications and serverless solutions that are hosted in AWS. The Solutions Architect must define a standard infrastructure that will be used across development teams and applications. There are application-specific resources too that change frequently, especially during the early stages of application development. Developers must be able to add supplemental resources to their applications, which are beyond what the architects predefined in the system environments and service templates.
Which of the following should be implemented to satisfy this requirement?

- A. Set up AWS Control Tower to automate container-based application deployments. Use AWS Config for application-specific resources that change frequently.
- B. Use the Amazon EKS Anywhere service for deploying container applications and serverless solutions. Create a service instance for each application-specific resource.
- C. Set up AWS Proton for deploying container applications and serverless solutions. Create components from the AWS Proton console and attach them to their respective service instance.
- D. Use the Amazon Elastic Container Service (ECS) Anywhere service for deploying container applications and serverless solutions. Configure Prometheus metrics collection on the ECS cluster and use Amazon Managed Service for Prometheus for monitoring frequently-changing resources

Answer: C

Explanation:
AWS Proton allows you to deploy any serverless or container-based application with increased efficiency, consistency, and control. You can define infrastructure standards and effective continuous delivery pipelines for your organization. Proton breaks down the infrastructure into environment and service ("infrastructure as code" templates).
As a developer, you select a standardized service template that AWS Proton uses to create a service that deploys and manages your application in a service instance. An AWS Proton service is an instantiation of a service template, which normally includes several service instances and a pipeline.



The diagram above displays the high-level overview of a simple AWS Proton workflow.
In AWS Proton administrators define standard infrastructure that is used across development teams and applications. However, development teams might need to include additional resources for their specific use cases, like Amazon Simple Queue Service (Amazon SQS) queues or Amazon DynamoDB tables.
These application-specific resources might change frequently, particularly during early application development. Maintaining these frequent changes in administrator-authored templates might be hard to manage and scale-administrators would need to maintain many more templates without real administrator added value. The alternative-letting application developers author templates for their applications-isn't ideal either, because it takes away administrators' ability to standardize the main architecture components, like AWS Fargate tasks. This is where components come in.
With a component, a developer can add supplemental resources to their application, above and beyond what administrators defined in environment and service templates. The developer then attaches the component to a service instance. AWS Proton provisions infrastructure resources defined by the component just like it provisions resources for environments and service instances.
Hence, the correct answer is: Set up AWS Proton for deploying container applications and serverless solutions. Create components from the AWS Proton console and attach them to their respective service instance.
The option that says: Use the Amazon EKS Anywhere service for deploying container applications and serverless solutions. Create a service instance for each application-specific resource is incorrect.
Amazon EKS Anywhere just allows you to manage a Kubernetes cluster on external environments that are supported by AWS. It is better to use AWS Proton with custom Components that can be attached to the different service instances of the company's application suite.
The option that says: Set up AWS Control Tower to automate container-based application deployments.
Use AWS Config for application-specific resources that change frequently is incorrect. AWS Control Tower is used to simplify the creation of new accounts with preconfigured constraints. It isn't used to automate application deployments. Moreover, AWS Config is commonly used for monitoring the changes of AWS resources and not the custom resources for serverless or container-based applications in AWS.
A combination of AWS Proton and Components is the most suitable solution for this scenario.
The option that says: Use the Amazon Elastic Container Service (ECS) Anywhere service for deploying container applications and serverless solutions. Configure Prometheus metrics collection on the ECS cluster and use Amazon Managed Service for Prometheus for monitoring frequently-changing resources is incorrect. The Amazon Managed Service for Prometheus is only a Prometheus-compatible monitoring and alerting service that makes it easy to monitor containerized applications and infrastructure at scale.
It is not capable of tracking or maintaining your application-specific resources that change frequently.
References:
https://docs.aws.amazon.com/proton/latest/userguide/Welcome.html
https://aws.amazon.com/blogs/architecture/simplifying-multi-account-ci-cd-deployments-using-aws-proto n/

NEW QUESTION # 249
A company receives data transfers from a small number of external clients that use SFTP software on an Amazon EC2 instance. The clients use an SFTP client to upload dat a. The clients use SSH keys for authentication. Every hour, an automated script transfers new uploads to an Amazon S3 bucket for processing.
The company wants to move the transfer process to an AWS managed service and to reduce the time required to start data processing. The company wants to retain the existing user management and SSH key generation process. The solution must not require clients to make significant changes to their existing processes.

Which solution will meet these requirements?

- A. Reconfigure the script that runs on the EC2 instance to run every 15 minutes. Create an S3 Event Notifications rule for all new object creation events. Set an Amazon Simple Notification Service (Amazon SNS) topic as the destination.
- B. Create an AWS Transfer Family SFTP connector that has permission to access the target S3 bucket for each client. Store credentials in AWS Systems Manager. Create an IAM role to allow the SFTP connector to securely use the credentials.
- C. Require clients to add the AWS DataSync agent into their local environments. Create an IAM user for each client that has permission to upload data to the target S3 bucket.
- D. Create an AWS Transfer Family SFTP server that uses the existing S3 bucket as a target. Use service-managed users to enable authentication.

Answer: D

Explanation:
AWS Transfer Family (SFTP) allows clients to use standard SFTP clients and SSH keys without changes. By enabling service-managed users, clients can continue uploading files with their existing tools.
The service delivers the files directly into S3, reducing latency between upload and processing. This removes the need for EC2, custom scripts, and periodic transfers. It fully meets the requirement for a managed solution with minimal disruption to client processes.

NEW QUESTION # 250
A company is developing a containerized web application that needs to be highly available and scalable.
The application requires access to GPU resources.

- A. Run the application on Amazon EC2 instances from a GPU instance family by using Amazon Elastic Container Service (Amazon ECS) for orchestration.
- B. Deploy the application container to Amazon Elastic Container Registry (Amazon ECR). Use Amazon ECR to run the containerized application with an attached GPU.
- C. Package the application as an AWS Lambda function in a container image. Use Lambda to run the containerized application on a runtime with GPU access.
- D. Deploy the application container to Amazon Elastic Kubernetes Service (Amazon EKS). Use AWS Fargate to manage compute resources and access to GPU resources.

Answer: A

Explanation:
Why Option D is Correct:
* GPU Access: Only EC2 instances in the GPU family (e.g., P2, P3) can provide GPU resources.
* ECS Orchestration: Simplifies container deployment and management.
Why Other Options Are Not Ideal:
* Option A: Lambda does not support GPU-based runtimes.
* Option B: AWS Fargate does not support GPU-based workloads.
* Option C: ECR is a container registry, not an orchestration or execution service.
AWS References:
* Amazon ECS with GPU Instances:AWS Documentation - ECS GPU Instances

NEW QUESTION # 251
A company tracks customer satisfaction by using surveys that the company hosts on its website. The surveys sometimes reach thousands of customers every hour. Survey results are currently sent in email messages to the company so company employees can manually review results and assess customer sentiment.
The company wants to automate the customer survey process. Survey results must be available for the previous 12 months.
Which solution will meet these requirements in the MOST scalable way?

- A. Send the survey results data to an Amazon API Gateway endpoint that is connected to an Amazon Simple Queue Service (Amazon SQS) queue. Configure the SQS queue to invoke an AWS Lambda function that calls Amazon Lex for sentiment analysis and saves the results to an Amazon DynamoDB table. Set the TTL for all records to 365 days in the future.
- B. Write the survey results data to an Amazon S3 bucket. Use S3 Event Notifications to invoke an AWS Lambda function to read the data and call Amazon Rekognition for sentiment analysis. Store the sentiment analysis results in a second S3 bucket. Use S3 Lifecycle policies on each bucket to expire objects after 365 days.

- C. Send the survey results data to an Amazon API Gateway endpoint that is connected to an Amazon Simple Queue Service (Amazon SQS) queue. Create an AWS Lambda function to poll the SQS queue, call Amazon Comprehend for sentiment analysis, and save the results to an Amazon DynamoDB table.
  Set the TTL for all records to 365 days in the future.
- D. Send the survey results data to an API that is running on an Amazon EC2 instance. Configure the API to store the survey results as a new record in an Amazon DynamoDB table, call Amazon Comprehend for sentiment analysis, and save the results in a second DynamoDB table. Set the TTL for all records to
  365 days in the future.

Answer: C

Explanation:
This solution is the most scalable and efficient way to handle large volumes of survey data while automating sentiment analysis:
* API Gateway and SQS: The survey results are sent to API Gateway, which forwards the data to an SQS queue. SQS can handle large volumes of messages and ensures that messages are not lost.
* AWS Lambda: Lambda is triggered by polling the SQS queue, where it processes the survey data.
* Amazon Comprehend: Comprehend is used for sentiment analysis, providing insights into customer satisfaction.
* DynamoDB with TTL: Results are stored in DynamoDB with a Time to Live (TTL) attribute set to expire after 365 days, automatically removing old data and reducing storage costs.
* Option B (EC2 API): Running an API on EC2 requires more maintenance and scalability management compared to API Gateway.
* Option C (S3 and Rekognition): Amazon Rekognition is for image and video analysis, not sentiment analysis.
* Option D (Amazon Lex): Amazon Lex is used for building conversational interfaces, not sentiment analysis.
AWS References:
* Amazon Comprehend for Sentiment Analysis
* Amazon SQS
* DynamoDB TTL

NEW QUESTION # 252
.....

□ and download it for free immediately on □ www.pdfvce.com □ □SAA-C03 Testking Exam Questions

- SAA-C03 Exam Cram Pdf □ SAA-C03 Test Sample Questions □ SAA-C03 Test Pattern □ Open [www.vceengine.com] and search for □ SAA-C03 □ to download exam materials for free □New SAA-C03 Braindumps Questions
- daotao.wisebusiness.edu.vn, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, www.stes.tyc.edu.tw, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, www.stes.tyc.edu.tw, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, www.stes.tyc.edu.tw, www.stes.tyc.edu.tw, justpaste.me, onartbook.co, Disposable vapes

BTW, DOWNLOAD part of PracticeMaterial SAA-C03 dumps from Cloud Storage: https://drive.google.com/open?id=1t76fBf81AV4ETkkn6nmy8zSAlgldr2b1