

AWS Certified Developer – Associate (DVA-C01) Guía del examen

Introducción

El examen AWS Certified Developer – Associate (DVA-C01) está dirigido a personas que desempeñan un rol de desarrollador. El examen valida la capacidad de un candidato para hacer lo siguiente:

- demostrar que conoce los servicios, los usos y las prácticas recomendadas de la arquitectura básica de AWS
- demostrar competencia en el desarrollo, la implementación y la depuración de aplicaciones basadas en la nube mediante AWS

Descripción del candidato objetivo

El candidato objetivo debe tener un año o más de experiencia práctica en el desarrollo y el mantenimiento de una aplicación basada en AWS.

Conocimientos generales de TI recomendados

El candidato objetivo debe tener los siguientes conocimientos:

- conocimiento profundo de al menos un lenguaje de programación de alto nivel
- conocimiento de la administración del ciclo de vida de las aplicaciones
- capacidad de escribir código para aplicaciones sin servidor
- conocimiento del uso de contenedores en el proceso de desarrollo

Conocimientos de AWS recomendados

El candidato objetivo debería poder hacer lo siguiente:

- utilizar las API de servicio de AWS, la CLI y los kits de desarrollo de software (SDK) para escribir aplicaciones
- identificar características clave de los servicios de AWS
- conocer el modelo de responsabilidad compartida de AWS
- utilizar una canalización de integración continua y entrega continua (CI/CD) para implementar aplicaciones en AWS
- utilizar los servicios de AWS e interactuar con ellos
- aplicar conocimientos básicos de las aplicaciones nativas en la nube para escribir código
- escribir código mediante las prácticas recomendadas de seguridad de AWS (por ejemplo, utilizar roles de IAM en lugar de claves secretas y de acceso en el código)
- crear, mantener y depurar módulos de código en AWS

¿Qué tareas no le competen al candidato objetivo?

A continuación, se muestra una lista no exhaustiva de tareas relacionadas que no se espera que el candidato objetivo pueda realizar. Estas tareas están fuera del alcance del examen:

- diseñar arquitecturas (por ejemplo, sistema distribuido, microservicios)
- diseñar e implementar canalizaciones de CI/CD
- administrar usuarios y grupos de IAM
- administrar Amazon Elastic Container Service (Amazon ECS)

- diseñar la infraestructura de redes de AWS (por ejemplo, Amazon VPC, AWS Direct Connect)
- tener conocimientos de la conformidad y de las licencias

Para ver una lista detallada de herramientas y tecnologías específicas que podrían abordarse en el examen, así como listas de los servicios de AWS que se encuentran dentro y fuera del alcance, consulte el Apéndice.

Contenido del examen

Tipos de respuestas

En el examen, hay dos tipos de preguntas:

- **Opción múltiple:** hay una respuesta correcta y tres incorrectas (distracciones).
- **Respuesta múltiple:** hay dos o más respuestas correctas entre cinco o más opciones.

Seleccione una o más respuestas que completen la afirmación o respondan a la pregunta de la mejor manera. Las distracciones o respuestas incorrectas son opciones que podría elegir un candidato con conocimientos o habilidades de un nivel inferior. Por lo general, las distracciones son respuestas verosímiles que coinciden con el área de contenido.

Las preguntas sin respuesta se califican como incorrectas. No hay penalización por adivinar. El examen incluye 50 preguntas que afectarán su puntuación.

Contenido sin puntuación

El examen incluye 15 preguntas sin puntuación que no afectan la puntuación total. AWS recopila información sobre el rendimiento de los candidatos en estas preguntas sin puntuación a fin de evaluarlas para su uso como preguntas con puntuación en el futuro. Estas preguntas sin puntuación no están identificadas en el examen.

Resultados del examen

El examen AWS Certified Developer – Associate (DVA-C01) es un examen con resultado de tipo aprobado o no aprobado. El puntaje se obtiene según un estándar mínimo que establecen los profesionales de AWS en función de las prácticas recomendadas y las pautas del sector de la certificación.

El informe de los resultados del examen es una puntuación en la escala del 100 al 1000. La puntuación mínima para aprobar es 720. La puntuación muestra cómo le fue en el examen en general y si lo aprobó o no. Los modelos de puntuación en escala ayudan a equiparar puntuaciones de varios formatos de examen que pueden tener niveles de dificultad un poco diferentes.

El informe de puntuación puede contener una tabla de clasificación de su rendimiento en cada sección. Esta información proporciona comentarios generales sobre su rendimiento en el examen. En el examen, se usa un modelo de puntuación compensatoria, lo que significa que no es necesario aprobar cada sección. Solo necesita aprobar el examen general.

Cada sección del examen tiene una ponderación específica, por lo que algunas contienen más preguntas que otras. La tabla presenta información general que resalta sus fortalezas y debilidades. Interprete los comentarios de cada sección con prudencia.

Esquema del contenido

En esta guía de examen, se incluyen ponderaciones, dominios de prueba y objetivos para el examen. No es una descripción completa del contenido del examen. Sin embargo, se encuentra disponible información adicional del contexto de cada uno de los objetivos a fin de ayudarlo a orientar la preparación para el examen. En la siguiente tabla, se enumeran los principales dominios de contenido y sus ponderaciones. La tabla precede al esquema completo del contenido del examen, que incluye el contexto adicional. El porcentaje de cada dominio solo representa el contenido que recibe una puntuación.

Dominio	% del examen
Dominio 1: Implementación	22 %
Dominio 2: Seguridad	26 %
Dominio 3: Desarrollo con servicios de AWS	30 %
Dominio 4: Refactorización	10 %
Dominio 5: Monitoreo y solución de problemas	12 %
TOTAL	100 %

Dominio 1: Implementación

1.1 Implemente código escrito en AWS mediante canalizaciones, procesos y patrones de CI/CD existentes.

- Confirme código en un repositorio e invoque acciones de creación, prueba o implementación.
- Utilice etiquetas y ramas para la administración de versiones y lanzamientos.
- Use AWS CodePipeline para organizar flujos de trabajo en diferentes entornos.
- Aplique AWS CodeCommit, AWS CodeBuild, AWS CodePipeline, AWS CodeStar y AWS CodeDeploy para fines de CI/CD.
- Ejecute un plan de reversión basado en la política de implementación de aplicaciones.

1.2 Implemente aplicaciones mediante AWS Elastic Beanstalk.

- Utilice entornos compatibles existentes para definir una nueva pila de aplicaciones.
- Empaquete la aplicación.
- Introduzca una nueva versión de la aplicación en el entorno de Elastic Beanstalk.
- Utilice una política de implementación para implementar una versión de la aplicación (es decir, todo a la vez, de forma continua, de forma continua con lote, inmutable).
- Valide el estado de la aplicación mediante el panel de Elastic Beanstalk.
- Use Amazon CloudWatch Logs para organizar el registro de aplicaciones.

1.3 Prepare el paquete de implementación de aplicaciones que se implementará en AWS.

- Administre las dependencias del módulo de código (como variables de entorno, archivos de configuración y archivos de imagen estática) dentro del paquete.
- Describa la estructura del directorio de paquetes o contenedores y organice los archivos de forma adecuada.
- Transforme los requisitos de recursos de la aplicación en parámetros de infraestructura de AWS (por ejemplo, memoria, núcleos).

1.4 Implemente aplicaciones sin servidor.

- En un caso de uso, implemente y lance una plantilla de AWS Serverless Application Model (AWS SAM).
- Administre entornos en servicios de AWS individuales (por ejemplo, diferencie entre desarrollo, prueba y producción en Amazon API Gateway).

Dominio 2: Seguridad

2.1 Realice llamadas autenticadas a los servicios de AWS.

- Comunique la política requerida en función de los privilegios mínimos que requiere la aplicación.
- Asuma un rol de IAM para acceder a un servicio.
- Utilice el proveedor de credenciales del kit de desarrollo de software (SDK) en las instalaciones o en la nube para acceder a los servicios de AWS (comparación entre credenciales locales y roles de instancia).

2.2 Implemente el cifrado mediante los servicios de AWS.

- Cifre datos en reposo (lado del cliente, lado del servidor, cifrado de sobres) mediante servicios de AWS.
- Cifre datos en tránsito.

2.3 Implemente la autenticación y autorización de aplicaciones.

- Agregue la funcionalidad de registro e inicio de sesión de usuarios para aplicaciones con grupos de usuarios o identidades de Amazon Cognito.
- Utilice las credenciales proporcionadas por Amazon Cognito para escribir código que accede a los servicios de AWS.
- Utilice la sincronización de Amazon Cognito para sincronizar perfiles de usuario y datos.
- Utilice identidades autenticadas por desarrolladores para interactuar entre los dispositivos de usuarios finales, la autenticación de backends y Amazon Cognito.

Dominio 3: Desarrollo con servicios de AWS

3.1 Escriba código para aplicaciones sin servidor.

- Compare y contraste modelos basados en servidor con modelos sin servidor (por ejemplo, microservicios, la naturaleza sin estado de las aplicaciones sin servidor, el escalado de aplicaciones sin servidor y el desacoplamiento de capas de aplicaciones sin servidor).
- Defina variables y parámetros de entorno (por ejemplo, memoria, tiempo de espera, tiempo de ejecución, controlador) para configurar las funciones de AWS Lambda.
- Cree un punto de enlace de API mediante Amazon API Gateway.
- Cree y pruebe acciones de API apropiadas como GET, POST con el punto de enlace de API.
- Aplique conceptos de Amazon DynamoDB (por ejemplo, tablas, elementos y atributos).
- Procese unidades de capacidad de lectura y escritura para Amazon DynamoDB según los requisitos de la aplicación.
- Asocie una función de AWS Lambda a una fuente de eventos de AWS (por ejemplo, Amazon API Gateway, evento de Amazon CloudWatch, eventos de Amazon S3, Amazon Kinesis).
- Invoque una función de AWS Lambda de forma síncrona y asíncrona.

3.2 Transforme requisitos funcionales en diseño de aplicaciones.

- Determine si se debe usar el procesamiento en tiempo real o el procesamiento por lotes para un caso de uso.
- Determine si se debe usar el enfoque de sincronía o asincronía para un caso de uso en particular.

- Determine si se deben usar eventos o cronogramas/encuestas para un caso de uso en particular.
 - Justifique las compensaciones de modelos de consistencia en el diseño de una aplicación.
- 3.3 Implemente el diseño de una aplicación en el código de la aplicación.
- Escriba código para utilizar servicios de mensajería (por ejemplo, SQS, SNS).
 - Use Amazon ElastiCache para crear una caché de base de datos.
 - Utilice Amazon DynamoDB para crear un índice de objetos en Amazon S3.
 - Escriba una función de AWS Lambda sin estado.
 - Escriba una aplicación web con servidores web sin estado (externalice el estado).
- 3.4 Escriba código que interactúe con los servicios de AWS mediante API, SDK y AWS CLI.
- Elija las API, los kits de desarrollo de software (SDK) y los comandos de CLI adecuados para los componentes del código.
 - Escriba código resiliente que se ocupe de errores o excepciones (es decir, reintentos con retroceso exponencial y fluctuación).

Dominio 4: Refactorización

- 4.1 Optimice las aplicaciones para aprovechar mejor los servicios y las características de AWS.
- Implemente servicios de almacenamiento en caché de AWS para optimizar el rendimiento (por ejemplo, Amazon ElastiCache, caché de Amazon API Gateway).
 - Aplique un esquema de nomenclatura de Amazon S3 para obtener un rendimiento de lectura óptimo.
- 4.2 Migre el código de la aplicación existente para ejecutarlo en AWS.
- Aísle dependencias.
 - Ejecute la aplicación como uno o varios procesos sin estado.
 - Desarrolle para permitir la escalabilidad horizontal.
 - Externalice el estado.

Dominio 5: Monitoreo y solución de problemas

- 5.1 Escriba código que pueda monitorearse.
- Cree métricas personalizadas de Amazon CloudWatch.
 - Realice el registro de una manera que esté disponible para los operadores de sistemas.
 - Organice el código fuente de la aplicación para que se habilite el seguimiento en AWS X-Ray.
- 5.2 Realice análisis de la causa raíz de las fallas encontradas en las pruebas o la producción.
- Interprete los resultados del mecanismo de registro de AWS para identificar errores en los registros.
 - Consulte el historial de creaciones y pruebas de los servicios de AWS (por ejemplo, AWS CodeBuild, AWS CodeDeploy, AWS CodePipeline) para identificar problemas.
 - Utilice servicios de AWS (por ejemplo, Amazon CloudWatch, VPC Flow Logs y AWS X-Ray) para localizar un componente defectuoso específico.

Apéndice

¿Qué herramientas, tecnologías y conceptos clave podrían incluirse en el examen?

La siguiente es una lista no exhaustiva de las herramientas y las tecnologías que podrían aparecer en el examen. Esta lista está sujeta a cambios y se proporciona para ayudarlo a comprender el alcance general de los servicios, las características o las tecnologías que se presentan en el examen. Las herramientas y las tecnologías generales de esta lista no aparecen en un orden particular. Los servicios de AWS se agrupan según sus funciones principales. Aunque es probable que el examen abarque algunas de estas tecnologías en mayor medida, el orden y su ubicación en esta lista no son indicios de su relevancia o importancia:

- Análisis
- Integración de aplicaciones
- Contenedores
- Administración de costos y capacidad
- Movimiento de datos
- Herramientas para desarrolladores
- Instancias (máquinas virtuales)
- Administración y gobernanza
- Redes y entrega de contenido
- Seguridad
- Sin servidor

Servicios y características de AWS

Análisis:

- Amazon Elasticsearch Service (Amazon ES)
- Amazon Kinesis

Integración de aplicaciones:

- Amazon EventBridge (Amazon CloudWatch Events)
- Amazon Simple Notification Service (Amazon SNS)
- Amazon Simple Queue Service (Amazon SQS)
- AWS Step Functions

Informática:

- Amazon EC2
- AWS Elastic Beanstalk
- AWS Lambda

Contenedores:

- Amazon Elastic Container Registry (Amazon ECR)
- Amazon Elastic Container Service (Amazon ECS)
- Amazon Elastic Kubernetes Service (Amazon EKS)

Base de datos:

- Amazon DynamoDB
- Amazon ElastiCache
- Amazon RDS

Herramientas para desarrolladores:

- AWS CodeArtifact
- AWS CodeBuild
- AWS CodeCommit
- AWS CodeDeploy
- Amazon CodeGuru
- AWS CodePipeline
- AWS CodeStar
- AWS Fault Injection Simulator
- AWS X-Ray

Administración y gobernanza:

- AWS CloudFormation
- Amazon CloudWatch

Redes y entrega de contenido:

- Amazon API Gateway
- Amazon CloudFront
- Elastic Load Balancing

Seguridad, identidad y conformidad:

- Amazon Cognito
- AWS Identity and Access Management (IAM)
- AWS Key Management Service (AWS KMS)

Almacenamiento:

- Amazon S3

Servicios y características de AWS fuera del alcance

A continuación, se muestra una lista no exhaustiva de los servicios y las características de AWS que no se incluyen en el examen. Estos servicios y características no representan todas las ofertas de AWS que no se incluyen en el contenido del examen. Los servicios o las características que no tienen ninguna relación con los objetivos de roles de trabajo del examen se excluyen de esta lista porque se supone que son irrelevantes.

Entre los servicios y las características de AWS fuera del alcance, se incluyen los siguientes:

- AWS Application Discovery Service
- Amazon AppStream 2.0
- Amazon Chime
- Amazon Connect
- AWS Database Migration Service (AWS DMS)
- AWS Device Farm
- Amazon Elastic Transcoder
- Amazon GameLift
- Amazon Lex
- Amazon Machine Learning (Amazon ML)
- AWS Managed Services
- Amazon Mobile Analytics
- Amazon Polly
- Amazon QuickSight
- Amazon Rekognition

- AWS Server Migration Service (AWS SMS)
- AWS Service Catalog
- AWS Shield Advanced
- AWS Shield Standard
- AWS Snow Family
- AWS Storage Gateway
- AWS WAF
- Amazon WorkMail
- Amazon WorkSpaces