

## AWS Certified Developer - Associate (DVA-C02) Guia do exame

### Introdução

O exame AWS Certified Developer – Associate (DVA-C02) destina-se a pessoas que desempenham a função de desenvolvedor. O exame valida a capacidade do candidato de demonstrar proficiência nas seguintes áreas: desenvolvimento, teste, implantação e depuração de aplicativos baseados na nuvem AWS.

O exame também valida a capacidade do candidato de concluir as seguintes tarefas:

- Desenvolver e otimizar aplicativos na AWS.
- Usar fluxos de trabalho de integração contínua e entrega contínua (CI/CD) para empacotar e implantar.
- Proteger o código e os dados do aplicativo.
- Identificar e resolver problemas de aplicativo.

### Descrição do candidato

O candidato tem um ou mais anos de experiência prática em desenvolvimento e manutenção de aplicativos por meio dos serviços da AWS.

#### Conhecimento geral de TI recomendado

O candidato deve ter:

- Proficiência em pelo menos uma linguagem de programação de alto nível
- Conhecimentos sobre gerenciamento do ciclo de vida do aplicativo
- Conhecimentos básicos de aplicativos nativos da nuvem para escrever código
- Capacidade de desenvolver aplicativos funcionais
- Experiência no uso de ferramentas de desenvolvimento

#### Conhecimento da AWS recomendado

O candidato deve ser capaz de:

- Desenvolver e proteger aplicativos por meio de APIs de serviço da AWS, da AWS CLI e de SDKs.
- Usar o pipeline de CI/CD para implantar aplicativos na AWS.

#### O que é considerado fora do escopo do candidato?

Veja a seguir uma lista (não completa) de tarefas de trabalho relacionadas as quais não se espera que o candidato seja capaz de executar. Estes itens são considerados fora do escopo do exame:

- Criação de arquiteturas (por exemplo, sistemas distribuídos, microsserviços, esquemas de banco de dados e modelagem)
- Projeto e criação de pipelines de CI/CD
- Administração de usuários e grupos do IAM
- Administração de servidores e sistemas operacionais
- Projeto de infraestrutura de redes da AWS (por exemplo, Amazon VPC, AWS Direct Connect)

Para visualizar uma lista detalhada de ferramentas e tecnologias específicas que podem ser abordadas no exame, além de listas de serviços da AWS que estão no escopo, consulte o Apêndice.

## Conteúdo do exame

### Tipos de respostas

Existem dois tipos de perguntas no exame:

- **Múltipla escolha:** tem uma resposta correta e três respostas incorretas (distratores)
- **Múltipla resposta:** tem duas ou mais respostas corretas dentre cinco ou mais opções de resposta

Selecione uma ou mais respostas que completem melhor a declaração ou respondam à pergunta. Distratores, ou respostas incorretas, são opções de resposta que um candidato com habilidades ou conhecimentos incompletos pode escolher. Geralmente, os distratores são respostas plausíveis que correspondem à área de conteúdo.

As perguntas não respondidas são pontuadas como incorretas; não há penalidade por tentar adivinhar. O exame inclui 50 perguntas que afetarão sua pontuação.

### Conteúdo não avaliado

O exame inclui 15 perguntas não pontuadas que não afetam sua pontuação. A AWS coleta informações sobre o desempenho do candidato nas perguntas não pontuadas a fim de avaliá-las para uso futuro como perguntas pontuadas. As perguntas não pontuadas não são identificadas no exame.

### Resultados do exame

O AWS Certified Developer – Associate (DVA-C02) é um exame de aprovação ou reprovação. O exame é pontuado de acordo com um padrão mínimo estabelecido por profissionais da AWS que seguem as práticas recomendadas e as diretrizes do setor de certificação.

Os resultados do exame são fornecidos como uma pontuação em escala de 100 a 1.000. A pontuação mínima de aprovação é de 720. A pontuação mostra como foi seu desempenho no exame como um todo e se você obteve aprovação. Os modelos de pontuação em escala ajudam a correlacionar as pontuações em várias formas de exame que podem ter níveis de dificuldade ligeiramente diferentes.

O relatório de pontuação pode conter uma tabela de classificação de seu desempenho em cada nível de seção. Essas informações se destinam a fornecer feedback geral sobre seu desempenho no exame. O exame usa um modelo de pontuação compensatória, o que significa que não é necessário obter uma pontuação de aprovação em cada seção. Você precisa passar apenas no exame geral.

Cada seção do exame tem um peso específico, portanto algumas seções têm mais perguntas do que outras. A tabela contém informações gerais que destacam seus pontos fortes e fracos. Tenha cuidado ao interpretar o feedback no nível da seção.

### Resumo do conteúdo

Este guia do exame inclui os pesos, os domínios do teste e as declarações de tarefas do exame. Este guia não é uma lista abrangente do conteúdo do exame. No entanto, disponibilizamos um contexto adicional para cada uma das declarações de tarefas visando ajudar a orientar sua preparação para o exame. A tabela

a seguir lista os principais domínios de conteúdo e seus pesos. A tabela precede a descrição completa do conteúdo do exame, que inclui o contexto adicional. A porcentagem de cada domínio representa apenas o conteúdo pontuado.

Domínio	% do exame
Domínio 1: Desenvolvimento com os serviços da AWS	32%
Domínio 2: Segurança	26%
Domínio 3: Implantação	24%
Domínio 4: Solução de problemas e otimização	18%
<b>TOTAL</b>	<b>100%</b>

## Domínio 1: Desenvolvimento com os serviços da AWS

Declaração de tarefa 1: Desenvolver código para aplicativos hospedados na AWS.

Conhecimento sobre:

- Padrões arquitetônicos (por exemplo, orientado por eventos, microsserviços, monolítico, coreografia, orquestração, fanout)
- Idempotência
- Diferenças entre os conceitos stateful e stateless
- Diferenças entre componentes com acoplamento rígido e acoplamento flexível
- Padrões de design tolerantes a falhas (por exemplo, novas tentativas com backoff exponencial e jitter, filas de mensagens mortas)
- Diferenças entre padrões síncronos e assíncronos

Habilidades em:

- Criar aplicativos tolerantes a falhas e resilientes em uma linguagem de programação (por exemplo, Java, C#, Python, JavaScript, TypeScript, Go)
- Criar, estender e manter APIs (por exemplo, transformações de resposta/solicitação, aplicação de regras de validação, substituição de códigos de status)
- Escrever e executar testes de unidade em ambientes de desenvolvimento [por exemplo, usando o AWS Serverless Application Model (AWS SAM)]
- Escrever código para usar serviços de sistema de mensagens
- Escrever código que interaja com os serviços da AWS por meio de APIs e SDKs da AWS
- Lidar com stream de dados usando os serviços da AWS

Declaração de tarefa 2: Desenvolver código para o AWS Lambda.

Conhecimento sobre:

- Mapeamento de origem de eventos
- Aplicativos stateless
- Teste de unidade
- Arquiteturas orientadas por eventos
- Dimensionamento

- Acesso de recursos privados em VPCs usando código do Lambda

Habilidades em:

- Configurar funções do Lambda por meio da definição de variáveis e parâmetros de ambiente (por exemplo, memória, simultaneidade, tempo limite, runtime, manipulador, camadas, extensões, gatilhos, destinos)
- Processar o ciclo de vida dos eventos e dos erros usando código (por exemplo, destinos do Lambda, filas de mensagens mortas)
- Escrever e executar código de teste usando serviços e ferramentas da AWS
- Integrar funções do Lambda aos serviços da AWS
- Ajustar as funções do Lambda para o desempenho ideal

Declaração de tarefa 3: Usar armazenamentos de dados no desenvolvimento de aplicativos.

Conhecimento sobre:

- Bancos de dados relacionais e não relacionais
- Operações de criação, leitura, atualização e exclusão (CRUD)
- Chaves de partição de alta cardinalidade para acesso balanceado à partição
- Opções de armazenamento em nuvem (por exemplo, arquivo, objeto, bancos de dados)
- Modelos de consistência de banco de dados (por exemplo, fortemente consistentes, eventualmente consistentes)
- Diferenças entre as operações de consulta e varredura
- Chaves e indexação do Amazon DynamoDB
- Estratégias de armazenamento em cache (por exemplo, write-through, read-through, lazy loading, TTL)
- Gerenciamento de níveis e ciclo de vida do Amazon S3
- Diferenças entre padrões de armazenamento de dados temporário e persistente

Habilidades em:

- Serializar e desserializar dados para fornecer persistência a um armazenamento de dados
- Usar, gerenciar e manter armazenamentos de dados
- Gerenciar os ciclos de vida dos dados
- Usar serviços de armazenamento em cache de dados

## Domínio 2: Segurança

Declaração de tarefa 1: Implementar autenticação e/ou autorização para aplicativos e serviços da AWS.

Conhecimento sobre:

- Federação de identidades [por exemplo, Security Assertion Markup Language (SAML), OpenID Connect (OIDC), Amazon Cognito]
- Tokens de portador [por exemplo, JSON Web Token (JWT), OAuth, AWS Security Token Service (AWS STS)]
- Comparação de grupos de usuários e de identidades no Amazon Cognito
- Políticas baseadas em recursos, políticas de serviço e políticas principais
- Controle de acesso baseado em função (RBAC)
- Autorização de aplicativo que usa ACLs
- Princípio de menor privilégio
- Diferenças entre as políticas gerenciadas pela AWS e as políticas gerenciadas pelo cliente

- Identity and Access Management (IAM)

Habilidades em:

- Usar um provedor de identidade para implementar o acesso federado [por exemplo, Amazon Cognito, AWS Identity and Access Management (IAM)]
- Proteger aplicativos usando tokens de portador
- Configurar o acesso programático à AWS
- Fazer chamadas autenticadas para serviços da AWS
- Assumir um perfil do IAM
- Definir permissões para entidades principais

Declaração de tarefa 2: Implementar criptografia usando os serviços da AWS.

Conhecimento sobre:

- Criptografia de dados em repouso e em trânsito
- Gerenciamento de certificados (por exemplo, AWS Certificate Manager Private Certificate Authority)
- Proteção de chaves (por exemplo, alternância de chaves)
- Diferenças entre criptografia do lado do cliente e criptografia do lado do servidor
- Diferenças entre chaves gerenciadas pela AWS e gerenciadas pelo cliente do AWS Key Management Service (AWS KMS)

Habilidades em:

- Usar chaves de criptografia para criptografar ou descriptografar dados
- Gerar certificados e chaves SSH para fins de desenvolvimento
- Usar criptografia além dos limites da conta
- Ativar e desativar a alternância de chaves

Declaração de tarefa 3: Gerenciar dados sigilosos no código do aplicativo.

Conhecimento sobre:

- Classificação de dados [por exemplo, informações de identificação pessoal (PII), informações de saúde protegidas (PHI)]
- Variáveis de ambiente
- Gerenciamento de segredos (por exemplo, AWS Secrets Manager, AWS Systems Manager Parameter Store)
- Tratamento seguro de credenciais

Habilidades em:

- Criptografar variáveis de ambiente que contêm dados sigilosos
- Usar serviços de gerenciamento de segredos para proteger dados sigilosos
- Limpar dados sigilosos

## Domínio 3: Implantação

Declaração de tarefa 1: Preparar artefatos de aplicativos para serem implantados na AWS.

Conhecimento sobre:

- Formas de acessar dados de configuração de aplicativos (por exemplo, AWS AppConfig, Secrets Manager, Parameter Store)
- Pacote de implantação do Lambda, camadas e opções de configuração
- Ferramentas de controle de versão baseadas em Git (por exemplo, Git, AWS CodeCommit)
- Imagens de contêiner

Habilidades em:

- Gerenciar as dependências do módulo de código (por exemplo, variáveis de ambiente, arquivos de configuração, imagens de contêiner) no pacote
- Organizar arquivos e uma estrutura de diretórios para implantação de aplicativos
- Usar repositórios de código em ambientes de implantação
- Aplicar requisitos de aplicativo para recursos (por exemplo, memória, núcleos)

Declaração de tarefa 2: Testar aplicativos em ambientes de desenvolvimento.

Conhecimento sobre:

- Recursos nos serviços da AWS que executam a implantação de aplicativos
- Teste de integração que usa endpoints simulados
- Versões e aliases do Lambda

Habilidades em:

- Testar código implantado usando serviços e ferramentas da AWS
- Realizar integração simulada para APIs e resolver dependências de integração
- Testar aplicativos usando endpoints de desenvolvimento (por exemplo, configurar estágios no Amazon API Gateway)
- Implantar atualizações de pilha de aplicativos em ambientes existentes (por exemplo, implantar um modelo do AWS SAM em um ambiente de teste diferente)

Declaração de tarefa 3: Automatizar testes de implantação.

Conhecimento sobre:

- Estágios do gateway de API
- Ramificações e ações no fluxo de trabalho de integração contínua e entrega contínua (CI/CD)
- Teste automatizado de software (por exemplo, teste de unidade, teste simulado)

Habilidades em:

- Criar eventos de teste de aplicativo (por exemplo, payloads JSON para testar recursos do Lambda, gateway de API, AWS SAM)
- Implantar recursos de API em vários ambientes
- Criar ambientes de aplicativo que usam versões aprovadas para testes de integração (por exemplo, aliases do Lambda, tags de imagem de contêiner, ramificações do AWS Amplify, ambientes do AWS Copilot)
- Implementar e implantar modelos de infraestrutura como código (IaC) (por exemplo, modelos do AWS SAM, modelos do AWS CloudFormation)

- Gerenciar ambientes em serviços individuais da AWS (por exemplo, diferenciação entre desenvolvimento, teste e produção no gateway de API)

Declaração de tarefa 4: Implantar código usando os serviços de CI/CD da AWS.

Conhecimento sobre:

- Ferramentas de controle de versão baseadas em Git (por exemplo, Git, AWS CodeCommit)
- Aprovações manuais e automatizadas no AWS CodePipeline
- Acesso a configurações de aplicativo do AWS AppConfig e do Secrets Manager
- Fluxos de trabalho de CI/CD que usam serviços da AWS
- Implantação de aplicativo que usa serviços e ferramentas da AWS [por exemplo, CloudFormation, AWS Cloud Development Kit (AWS CDK), AWS SAM, AWS CodeArtifact, Copilot, Amplify, Lambda]
- Opções de pacote de implantação do Lambda
- Estágios do gateway de API e domínios personalizados
- Estratégias de implantação (por exemplo, canary, azul/verde, contínua)

Habilidades em:

- Atualizar modelos de IaC existentes (por exemplo, modelos do AWS SAM, modelos do CloudFormation)
- Gerenciar ambientes de aplicativo usando os serviços da AWS
- Implantar uma versão do aplicativo usando estratégias de implantação
- Confirmar código em um repositório para invocar ações de compilação, teste e implantação
- Usar fluxos de trabalho orquestrados para implantar código em diferentes ambientes
- Realizar reversões de aplicativos usando estratégias de implantação existentes
- Usar rótulos e ramificações para o gerenciamento de versões e releases
- Usar configurações de runtime existentes para criar implantações dinâmicas (por exemplo, usando variáveis de preparação do gateway de API em funções do Lambda)

## **Domínio 4: Solução de problemas e otimização**

Declaração de tarefa 1: Auxiliar na análise da causa raiz.

Conhecimento sobre:

- Sistemas de registro em log e monitoramento
- Linguagens para consultas de log (por exemplo, Amazon CloudWatch Logs Insights)
- Visualizações de dados
- Ferramentas de análise de código
- Códigos de erro HTTP comuns
- Exceções comuns geradas por SDKs
- Mapas de serviço no AWS X-Ray

Habilidades em:

- Depurar código para identificar defeitos
- Interpretar métricas, logs e rastreamentos de aplicativos
- Consultar logs para encontrar dados relevantes
- Implementar métricas personalizadas (por exemplo, formato de métrica incorporada do CloudWatch [EMF])
- Revisar a integridade do aplicativo usando painéis e insights

- Solucionar problemas de falhas de implantação usando logs de saída de serviço

Declaração de tarefa 2: Instrumentar código para observabilidade.

Conhecimento sobre:

- Rastreamento distribuído
- Diferenças entre registro em log, monitoramento e observabilidade
- Registro em log estruturado
- Métricas de aplicativo (por exemplo, personalizadas, incorporadas, integradas)

Habilidades em:

- Implementar uma estratégia de registro em log eficaz para registrar o comportamento e o estado do aplicativo
- Implementar código que emite métricas personalizadas
- Adicionar anotações para serviços de rastreamento
- Implementar alertas de notificação para ações específicas (por exemplo, notificações sobre limites de cota ou conclusões de implantação)
- Implementar rastreamento usando serviços e ferramentas da AWS

Declaração de tarefa 3: Otimizar aplicativos usando os serviços e os recursos da AWS.

Conhecimento sobre:

- Armazenamento em cache
- Simultaneidade
- Serviços de sistema de mensagens [por exemplo, Amazon Simple Queue Service (Amazon SQS), Amazon Simple Notification Service (Amazon SNS)]

Habilidades em:

- Definir o perfil do desempenho do aplicativo
- Determinar a memória mínima e o poder computacional de um aplicativo
- Usar políticas de filtro de assinatura para otimizar o sistema de mensagens
- Armazenar conteúdo em cache com base em cabeçalhos de solicitação



## Apêndice

### **Quais são as principais ferramentas, tecnologias e conceitos que podem ser abordados no exame?**

Veja a seguir uma lista (não completa) de ferramentas e tecnologias que podem aparecer no exame. Essa lista está sujeita a alterações e é fornecida para ajudar a entender o escopo geral de serviços, recursos ou tecnologias no exame. As ferramentas e tecnologias gerais dessa lista não aparecem em nenhuma ordem específica. Os produtos da AWS são agrupados de acordo com suas funções principais. Embora algumas dessas tecnologias provavelmente sejam abordadas mais do que outras no exame, a ordem e a colocação delas na lista não são indicativos de importância nem de peso relativo:

- Análise
- Integração de aplicativos
- Computação
- Contêineres
- Gerenciamento de custos e capacidade
- Banco de dados
- Ferramentas do desenvolvedor
- Gerenciamento e governança
- Redes e entrega de conteúdo
- Segurança, identidade e conformidade
- Armazenamento

### **Recursos e produtos da AWS no escopo**

Análise:

- Amazon Athena
- Amazon Kinesis
- Amazon OpenSearch Service

Integração de aplicativos:

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

Computação:

- Amazon EC2
- AWS Elastic Beanstalk
- AWS Lambda
- AWS Serverless Application Model (AWS SAM)

Contêineres:

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

Banco de dados:

- Amazon Aurora
- Amazon DynamoDB
- Amazon ElastiCache
- Amazon MemoryDB para Redis
- Amazon RDS

Ferramentas do desenvolvedor:

- AWS Amplify
- AWS Cloud9
- AWS CloudShell
- AWS CodeArtifact
- AWS CodeBuild
- AWS CodeCommit
- AWS CodeDeploy
- Amazon CodeGuru
- AWS CodePipeline
- AWS CodeStar
- AWS X-Ray

Gerenciamento e governança:

- AWS AppConfig
- AWS Cloud Development Kit (AWS CDK)
- AWS CloudFormation
- AWS CloudTrail
- Amazon CloudWatch
- Amazon CloudWatch Logs
- AWS Command Line Interface (AWS CLI)
- AWS Systems Manager

Redes e entrega de conteúdo:

- Amazon API Gateway
- Amazon CloudFront
- Elastic Load Balancing
- Amazon Route 53
- Amazon VPC

Segurança, identidade e conformidade:

- AWS Certificate Manager (ACM)
- AWS Certificate Manager Private Certificate Authority
- Amazon Cognito
- AWS Identity and Access Management (IAM)
- AWS Key Management Service (AWS KMS)
- AWS Secrets Manager
- AWS Security Token Service (AWS STS)
- AWS WAF

Armazenamento:

- Amazon Elastic Block Store (Amazon EBS)
- Amazon Elastic File System (Amazon EFS)
- Amazon S3
- Amazon S3 Glacier

## **Recursos e produtos da AWS fora do escopo**

Veja a seguir uma lista (não completa) de recursos e produtos da AWS que não são abordados no exame. Estes recursos e produtos não representam todos os serviços da AWS que foram excluídos do conteúdo do exame. Os recursos ou os serviços que não têm nenhuma relação com as funções de trabalho às quais o exame se destina não estão nesta lista porque são considerados irrelevantes.

Os recursos e os serviços da AWS fora do escopo são:

- 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 (AMS)
- Amazon Mobile Analytics
- Amazon Polly
- Amazon QuickSight
- Amazon Rekognition
- AWS Server Migration Service (AWS SMS)
- AWS Service Catalog
- AWS Shield Standard
- AWS Shield Advanced
- Família AWS Snow
- AWS Storage Gateway
- Amazon WorkMail
- Amazon WorkSpaces