

Guida all'esame AWS Certified Developer - Associate (DVA-C02)

Introduzione

L'esame AWS Certified Developer - Associate (DVA-C02) è stato concepito per chi ricopre il ruolo di sviluppatore. L'esame valuta le competenze dei candidati relative a sviluppo, test, distribuzione e debug di applicazioni AWS basate sul cloud.

Inoltre, durante l'esame viene valutata la capacità dei candidati di completare le seguenti attività:

- Sviluppo e ottimizzazione di applicazioni su AWS.
- Pacchettizzazione e distribuzione utilizzando flussi di lavoro di integrazione continua e consegna continua (CI/CD).
- Protezione del codice e dei dati delle applicazioni.
- Identificazione e risoluzione di problemi delle applicazioni.

Descrizione del candidato target

Il candidato target dovrebbe avere uno o più anni di esperienza diretta nello sviluppo e nella manutenzione di applicazioni che utilizzano i servizi AWS.

Conoscenze IT generali consigliate

Il candidato target dovrebbe possedere le seguenti conoscenze IT generali:

- Conoscenza approfondita di almeno un linguaggio di programmazione di alto livello
- Comprensione della gestione del ciclo di vita delle applicazioni
- Conoscenze di base delle applicazioni native per il cloud adeguate per completare la scrittura di un codice
- Capacità di sviluppare applicazioni funzionali
- Esperienza nell'uso di strumenti di sviluppo

Conoscenze AWS consigliate

Il candidato target dovrebbe essere in grado di completare le seguenti attività:

- Sviluppare e proteggere applicazioni utilizzando le API dei servizi AWS, la AWS CLI e gli SDK.
- Utilizzare la pipeline CI/CD per la distribuzione di applicazioni su AWS.

Attività lavorative che non rientrano nelle competenze del candidato target

Il seguente elenco illustra le attività lavorative che non rientrano nelle competenze previste per il candidato target. L'elenco non è esaustivo. Le seguenti attività non rientrano nell'ambito dell'esame:

- Architetture di progettazione (ad esempio sistemi distribuiti, microservizi, schemi di database e modellazione).
- Progettazione e creazione di pipeline CI/CD.
- Amministrazione di utenti e gruppi IAM.
- Amministrazione di server e sistemi operativi.
- Progettazione di un'infrastruttura di rete AWS (ad esempio Amazon VPC, AWS Direct Connect).

Consulta l'Appendice per un elenco di tecnologie e concetti che potrebbero essere inclusi nell'esame, nonché per un elenco di servizi e funzionalità AWS trattati in sede d'esame e un elenco di quelli esclusi.

Contenuto dell'esame

Tipi di risposta

L'esame prevede due tipi di domande:

- **Scelta multipla:** una risposta corretta e tre risposte errate (distrattori)
- **Risposta multipla:** due o più risposte corrette su cinque o più opzioni di risposta

Selezionare una o più risposte che meglio completano l'affermazione o rispondono alla domanda. I distrattori, o risposte errate, sono opzioni di risposta che possono essere scelte da candidati con conoscenze o competenze insufficienti. Solitamente, i distrattori sono risposte plausibili che rientrano nell'ambito dei contenuti trattati.

Le domande senza risposta sono valutate come errate; non è applicata alcuna penalità se il candidato tenta una risposta. L'esame prevede 50 domande che influiscono sul punteggio finale.

Contenuto senza punteggio

L'esame include 15 domande alle quali non viene assegnato un punteggio e che non influiscono sul risultato finale. AWS raccoglie informazioni sulle prestazioni relativamente a queste domande, al fine di valutare la possibilità di convertirle in futuro in domande a punteggio. In sede di esame, le domande che non influiscono sul punteggio non verranno distinte dalle altre.

Risultati dell'esame

L'esame AWS Certified Developer - Associate (DVA-C02) prevede un esito netto, superamento o mancato superamento. La valutazione avviene in base a uno standard minimo stabilito da professionisti AWS che seguono le best practice e le linee guida del settore delle certificazioni.

I risultati dell'esame sono espressi da un punteggio compreso tra 100 e 1.000. Il punteggio minimo richiesto per il superamento della prova è 720. Il punteggio riflette le prestazioni complessive del candidato durante l'esame e indica se l'esame è stato superato o meno. I modelli di punteggio scalare aiutano a equiparare i punteggi tra moduli dell'esame che potrebbero presentare livelli di difficoltà leggermente diversi.

Il report relativo al punteggio può contenere una tabella di classificazione del rendimento in ogni sezione. Per l'esame viene impiegato un modello di punteggio compensativo; ciò significa che non è necessario ottenere un punteggio sufficiente in ogni sezione. L'esame viene superato se il punteggio complessivo ottenuto corrisponde almeno al minimo richiesto.

Ogni sezione ha un proprio peso specifico, quindi alcune di esse presentano più domande di altre. La seguente tabella delle classificazioni include informazioni generali che evidenziano i punti forti e deboli del candidato. Interpreta con la massima attenzione il feedback relativo a ogni sezione.

Descrizione del contenuto

Questa guida all'esame include informazioni sui pesi, sui domini dei contenuti e sugli obiettivi dell'esame. Non fornisce un elenco esaustivo dei contenuti dell'esame. Tuttavia, per ogni obiettivo è disponibile maggiore contesto come aiuto durante la preparazione all'esame.

Di seguito sono elencati i domini dei contenuti e i pesi dell'esame:

- Dominio 1: Sviluppo utilizzando i servizi AWS (32% dei contenuti a punteggio)
- Dominio 2: Sicurezza (26% dei contenuti a punteggio)
- Dominio 3: Implementazione (24% dei contenuti a punteggio)
- Dominio 4: Risoluzione dei problemi e ottimizzazione (18% dei contenuti a punteggio)

Dominio 1: Sviluppo utilizzando i servizi AWS

Obiettivo 1: Sviluppo di un codice per applicazioni ospitate su AWS.

Conoscenza di:

- Modelli di architettura (ad esempio basati su eventi, su microservizi, monolitici, coreografia, orchestrazione, fan-out)
- Idempotenza
- Differenze tra concetti stateful e stateless
- Differenze tra componenti strettamente accoppiati e debolmente accoppiati
- Modelli di progettazione tolleranti ai guasti (ad esempio nuovi tentativi con backoff esponenziale e jitter, code di messaggi non recapitabili)
- Differenze tra modelli sincroni e asincroni

Competenze in:

- Creazione di applicazioni resilienti e tolleranti ai guasti in un linguaggio di programmazione (ad esempio Java, C#, Python, JavaScript, TypeScript, Go)
- Creazione, estensione e gestione di API (ad esempio trasformazioni di risposte/richieste, applicazione di regole di convalida, sovrascrittura dei codici di stato)
- Scrittura ed esecuzione di unit test in ambienti di sviluppo (ad esempio utilizzando AWS Serverless Application Model [AWS SAM])

- Scrittura di codici per l'utilizzo di servizi di messaggistica
- Scrittura di codici che interagiscano con i servizi AWS utilizzando API e SDK AWS
- Gestione dello streaming di dati utilizzando i servizi AWS

Obiettivo 2: Sviluppo di un codice per AWS Lambda.

Conoscenza di:

- Mappatura delle origini degli eventi
- Applicazioni stateless
- Esecuzione di unit test
- Architetture basate su eventi
- Scalabilità
- Accesso alle risorse private nei VPC dal codice Lambda

Competenze in:

- Configurazione delle funzioni Lambda definendo variabili e parametri di ambiente (ad esempio memoria, concorrenza, timeout, runtime, gestore, livello, estensioni, trigger, destinazioni)
- Gestione del ciclo di vita degli eventi e degli errori utilizzando il codice (ad esempio destinazioni Lambda, code di messaggi non recapitabili)
- Scrittura ed esecuzione di un codice di test utilizzando servizi e strumenti AWS
- Integrazione delle funzioni Lambda con i servizi AWS
- Messa a punto delle funzioni Lambda per prestazioni ottimali

Obiettivo 3: Utilizzo di archivi dati nello sviluppo di applicazioni.

Conoscenza di:

- Database relazionali e non relazionali
- Operazioni CRUD (creazione, lettura, aggiornamento ed eliminazione)
- Chiavi di partizione ad alta cardinalità per un accesso bilanciato alle partizioni
- Opzioni di archiviazione nel cloud (ad esempio file, oggetti, database)
- Modelli di consistenza del database (ad esempio consistenza elevata, consistenza finale)
- Differenze tra operazioni di query e di scansione

- Chiavi di Amazon DynamoDB e indicizzazione
- Strategie di memorizzazione nella cache (ad esempio write-through, read-through, caricamento lazy, TTL)
- Livelli e gestione del ciclo di vita di Amazon S3
- Differenze tra modelli di archiviazione dei dati effimeri e persistenti

Competenze in:

- Serializzazione e deserializzazione dei dati per fornire persistenza a un archivio dati
- Utilizzo, gestione e manutenzione di archivi dati
- Gestione dei cicli di vita dei dati
- Utilizzo dei servizi di memorizzazione di dati nella cache

Dominio 2: Sicurezza

Obiettivo 1: Implementazione di autenticazione e/o autorizzazione per applicazioni e servizi AWS.

Conoscenza di:

- Federazione delle identità (ad esempio Security Assertion Markup Language [SAML], OpenID Connect [OIDC], Amazon Cognito)
- Token di connessione (ad esempio JSON Web Token [JWT], OAuth, AWS Security Token Service [AWS STS])
- Confronto tra pool di utenti e pool di identità in Amazon Cognito
- Policy basate sulle risorse, policy del servizio e policy delle entità
- Controllo degli accessi basato sul ruolo (RBAC)
- Autorizzazione degli accessi all'applicazione attraverso ACL
- Principio del privilegio minimo
- Differenze tra le policy gestite da AWS e dal cliente
- Gestione dell'identità e degli accessi

Competenze in:

- Utilizzo di un provider di identità per implementare l'accesso federato (ad esempio Amazon Cognito, AWS Identity and Access Management [IAM])
- Protezione delle applicazioni utilizzando i token di connessione
- Configurazione dell'accesso programmatico ad AWS
- Esecuzione di chiamate autenticate ai servizi AWS

- Assunzione di un ruolo IAM
- Definizione delle autorizzazioni per le entità

Obiettivo 2: Implementazione della crittografia utilizzando i servizi AWS.

Conoscenza di:

- Crittografia dei dati inattivi e in transito
- Gestione dei certificati (ad esempio AWS Private Certificate Authority)
- Protezione delle chiavi (ad esempio rotazione delle chiavi)
- Differenze tra crittografia lato client e crittografia lato server
- Differenze nella gestione delle chiavi AWS Key Management Service (AWS KMS) da parte di AWS e del cliente

Competenze in:

- Utilizzo delle chiavi di crittografia per crittografare o decrittografare i dati
- Generazione di certificati e chiavi SSH per scopi di sviluppo
- Utilizzo della crittografia oltre i confini dell'account
- Abilitazione e disabilitazione della rotazione delle chiavi

Obiettivo 3: Gestione dei dati sensibili nel codice dell'applicazione.

Conoscenza di:

- Classificazione dei dati (ad esempio informazioni di identificazione personale [PII], informazioni sanitarie protette [PHI])
- Variabili di ambiente
- Gestione dei segreti (ad esempio AWS Secrets Manager, Archivio dei parametri AWS Systems Manager)
- Gestione sicura delle credenziali

Competenze in:

- Crittografia delle variabili di ambiente che contengono dati sensibili
- Utilizzo di servizi di gestione dei segreti per proteggere i dati sensibili
- Sanificazione dei dati sensibili

Dominio 3: Implementazione

Obiettivo 1: Preparazione degli artefatti delle applicazioni da distribuire in AWS.

Conoscenza di:

- Modalità di accesso ai dati di configurazione delle applicazioni (ad esempio AWS AppConfig, Secrets Manager, Archivio dei parametri)
- Pacchettizzazione, livelli e opzioni di configurazione della distribuzione Lambda
- Strumenti di controllo delle versioni basati su Git (ad esempio Git, AWS CodeCommit)
- Immagini del container

Competenze in:

- Gestione delle dipendenze del modulo di codice (ad esempio variabili di ambiente, file di configurazione, immagini del container) all'interno del pacchetto
- Organizzazione dei file e di una struttura di directory per la distribuzione delle applicazioni
- Utilizzo di repository di codice negli ambienti di distribuzione
- Applicazione dei requisiti di risorsa per le app (ad esempio memoria, core)

Obiettivo 2: Test delle applicazioni negli ambienti di sviluppo.

Conoscenza di:

- Funzionalità dei servizi AWS che permettono di eseguire la distribuzione delle applicazioni
- Test dell'integrazione eseguita tramite endpoint fittizi
- Versioni e alias di Lambda

Competenze in:

- Test del codice distribuito utilizzando servizi e strumenti AWS
- Esecuzione di integrazioni fittizie per le API e risoluzione delle dipendenze dell'integrazione
- Test delle applicazioni utilizzando endpoint di sviluppo (ad esempio configurazione delle fasi in Amazon API Gateway)

- Distribuzione di aggiornamenti degli stack di applicazioni in ambienti esistenti (ad esempio distribuzione di un template AWS SAM in un ambiente di staging diverso)

Obiettivo 3: Automatizzazione dei test di distribuzione.

Conoscenza di:

- Fasi di API Gateway
- Rami e azioni nel flusso di lavoro di integrazione continua e consegna continua (CI/CD)
- Test automatizzati di software (ad esempio unit test, test simulati)

Competenze in:

- Creazione di eventi di test delle applicazioni (ad esempio payload in formato JSON per eseguire il test di risorse Lambda, API Gateway, AWS SAM)
- Distribuzione di risorse API in vari ambienti
- Creazione di ambienti delle applicazioni che utilizzano versioni approvate per i test di integrazione (ad esempio alias Lambda, tag dell'immagine del container, rami di AWS Amplify, ambienti AWS Copilot)
- Implementazione e distribuzione di template di Infrastructure as code (IaC) (ad esempio template AWS SAM, AWS CloudFormation)
- Gestione di ambienti nei singoli servizi AWS (ad esempio differenziazione tra sviluppo, test e produzione in API Gateway)

Obiettivo 4: Distribuzione di un codice tramite i servizi CI/CD di AWS.

Conoscenza di:

- Strumenti di controllo delle versioni basati su Git (ad esempio Git, AWS CodeCommit)
- Approvazioni manuali e automatiche in AWS CodePipeline
- Accesso alle configurazioni delle applicazioni da AWS AppConfig e Secrets Manager
- Flussi di lavoro CI/CD che utilizzano i servizi AWS
- Distribuzione di applicazioni che utilizzano servizi e strumenti AWS (ad esempio CloudFormation, AWS Cloud Development Kit [AWS CDK], AWS SAM, AWS CodeArtifact, AWS Copilot, Amplify, Lambda)

- Opzioni di pacchettizzazione della distribuzione Lambda
- Fasi di API Gateway e domini personalizzati
- Strategie di distribuzione (ad esempio canary, blu/verde, continua)

Competenze in:

- Aggiornamento di template IaC esistenti (ad esempio template AWS SAM, CloudFormation)
- Gestione degli ambienti applicativi utilizzando i servizi AWS
- Distribuzione di una versione dell'applicazione utilizzando strategie di distribuzione
- Esecuzione del commit di un codice in un repository per richiamare azioni di creazione, test e distribuzione
- Utilizzo di flussi di lavoro orchestrati per distribuire un codice in ambienti diversi
- Ripristino dello stato precedente delle applicazioni utilizzando strategie di distribuzione esistenti
- Utilizzo di tag e rami per la gestione delle versioni e dei rilasci
- Utilizzo di configurazioni di runtime esistenti per creare distribuzioni dinamiche (ad esempio utilizzando variabili di staging da API Gateway nelle funzioni Lambda)

Dominio 4: Risoluzione dei problemi e ottimizzazione

Obiettivo 1: Assistenza durante l'analisi della causa principale.

Conoscenza di:

- Sistemi di registrazione e monitoraggio
- Linguaggi per le query di log (ad esempio Amazon CloudWatch Logs Insights)
- Visualizzazioni dei dati
- Strumenti di analisi del codice
- Codici di errore HTTP comuni
- Eccezioni comuni generate dagli SDK
- Mappe dei servizi in AWS X-Ray

Competenze in:

- Debug di un codice per identificare i difetti
- Interpretazione di metriche, log e tracce delle applicazioni
- Query sui log per trovare dati pertinenti
- Implementazione di metriche personalizzate (ad esempio il formato metrico incorporato di CloudWatch [EMF])
- Revisione dello stato delle applicazioni utilizzando dashboard e informazioni
- Risoluzione degli errori di distribuzione utilizzando i registri di output del servizio

Obiettivo 2: Inclusione nel codice di strumenti per l'osservabilità.

Conoscenza di:

- Tracciamento distribuito
- Differenze tra registrazione, monitoraggio e osservabilità
- Registrazione strutturata
- Metriche delle applicazioni (ad esempio personalizzate, integrate, incorporate)

Competenze in:

- Implementazione di una strategia di registrazione efficace per registrare il comportamento e lo stato delle applicazioni
- Implementazione di un codice per la creazione di metriche personalizzate
- Aggiunta di annotazioni per i servizi di tracciamento
- Implementazione di avvisi di notifica per azioni specifiche (ad esempio notifiche relative ai limiti delle quote o al completamento della distribuzione)
- Implementazione del tracciamento utilizzando servizi e strumenti AWS

Obiettivo 3: Ottimizzazione delle applicazioni utilizzando i servizi e le funzionalità di AWS.

Conoscenza di:

- Memorizzazione nella cache
- Simultaneità
- Servizi di messaggistica (ad esempio Amazon Simple Queue Service [Amazon SQS], Amazon Simple Notification Service [Amazon SNS])

Competenze in:

- Profilazione delle prestazioni delle applicazioni
- Determinazione della memoria e della potenza di calcolo minime per un'applicazione
- Utilizzo di policy di filtro delle sottoscrizioni per ottimizzare la messaggistica
- Memorizzazione nella cache dei contenuti in base alle intestazioni della richiesta

Appendice

Tecnologie e concetti che potrebbero essere inclusi nell'esame

Il seguente elenco contiene tecnologie e concetti che potrebbero essere inclusi nell'esame. Si tratta di un elenco non esaustivo e soggetto a modifiche. L'ordine e la posizione delle voci nell'elenco non sono rappresentativi del loro peso o importanza in relazione all'esame:

- Analisi
- Integrazione di applicazioni
- Calcolo
- Container
- Gestione di costi e capacità
- Database
- Strumenti di sviluppo
- Gestione e governance su AWS
- Reti e distribuzione di contenuti
- Sicurezza, identità e conformità
- Archiviazione

Servizi e funzionalità AWS trattati in sede di esame

Il seguente elenco contiene i servizi e le funzionalità AWS trattati nell'esame. Si tratta di un elenco non esaustivo e soggetto a modifiche. Le offerte AWS sono suddivise in categorie in funzione delle loro funzioni principali:

Analisi:

- Amazon Athena
- Amazon Kinesis
- Servizio OpenSearch di Amazon

Integrazione di applicazioni:

- AWS AppSync
- Amazon EventBridge
- Amazon Simple Notification Service (Amazon SNS)
- Amazon Simple Queue Service (Amazon SQS)

- AWS Step Functions

Calcolo:

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

Container:

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

Database:

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

Strumenti di sviluppo:

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

Gestione e governance su AWS:

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

Reti e distribuzione di contenuti:

- Amazon API Gateway
- Amazon CloudFront
- Elastic Load Balancing (ELB)
- Amazon Route 53
- Amazon VPC

Sicurezza, identità e conformità:

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

Archiviazione:

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

Servizi e funzionalità AWS non trattati in sede di esame

Il seguente elenco contiene i servizi e le funzionalità AWS non trattati nell'esame. Si tratta di un elenco non esaustivo e soggetto a modifiche. Sono esclusi da questo elenco le offerte AWS del tutto estranee ai ruoli target per l'esame:

Analisi:

- Amazon QuickSight

Applicazioni aziendali:

- Amazon Chime
- Amazon Connect
- Amazon WorkMail

Servizi informatici per utenti finali:

- Amazon AppStream 2.0
- Amazon WorkSpaces

Front-end per il web e i dispositivi mobili:

- AWS Device Farm

Tecnologie di gioco:

- Amazon GameLift

Machine learning:

- Amazon Lex
- Amazon Machine Learning (Amazon ML)
- Amazon Polly
- Amazon Rekognition

Gestione e governance su AWS:

- AWS Managed Services (AMS)
- AWS Service Catalog

Servizi multimediali:

- Amazon Elastic Transcoder

Migrazione e trasferimento:

- AWS Application Discovery Service
- AWS Application Migration Service
- AWS Database Migration Service (AWS DMS)

Sicurezza, identità e conformità:

- AWS Shield Advanced
- AWS Shield Standard

Archiviazione:

- AWS Snow Family
- AWS Storage Gateway

Sondaggio

Quanto è stata utile questa guida all'esame? Per farci sapere cosa ne pensi, [partecipa al sondaggio](#).