

AWS Certified Developer – Associate (DVA-C01) Guide de l'examen

Introduction

L'examen AWS Certified Developer – Associate (DVA-C01) est destiné aux personnes ayant un rôle de développeur. L'examen valide la capacité du candidat à :

- démontrer qu'il comprend les principaux services et cas d'utilisation AWS, ainsi que les bonnes pratiques de base en matière d'architecture AWS ;
- démontrer ses compétences en développement, déploiement et débogage des applications basées sur le cloud à l'aide d'AWS.

Description du candidat cible

Le candidat cible doit posséder au moins un an d'expérience pratique en matière de développement et de gestion d'applications basées sur AWS.

Connaissances informatiques générales recommandées

Le candidat cible doit posséder les connaissances suivantes :

- Maîtrise d'au moins un langage de programmation de haut niveau
- Compréhension de la gestion du cycle de vie des applications
- Capacité à écrire du code pour les applications sans serveur
- Compréhension de l'utilisation des conteneurs dans le processus de développement

Connaissances recommandées concernant AWS

Le candidat cible doit être capable d'effectuer les opérations suivantes :

- Utiliser les API de service AWS, l'interface de ligne de commande (CLI) et les kits de développement logiciel (SDK) pour écrire des applications
- Identifier les fonctions clés des services AWS
- Comprendre le modèle de responsabilité partagée AWS
- Utiliser un pipeline d'intégration continue et de livraison continue (CI/CD) pour déployer des applications sur AWS
- Utiliser des services AWS et interagir avec ces derniers
- Appliquer une compréhension de base des applications natives cloud pour écrire du code
- Écrire du code à l'aide des bonnes pratiques de sécurité AWS (par exemple, utiliser des rôles IAM au lieu de clés secrètes et de clés d'accès dans le code)
- Créer, entretenir et déboguer des modules de code sur AWS

Qu'est-ce qui est considéré comme hors du champ des exigences pour le candidat cible ?

Voici une liste non exhaustive des tâches connexes que le candidat cible ne doit pas impérativement être capable d'effectuer. Ces éléments sont considérés comme hors champ pour l'examen :

- Concevoir des architectures (par exemple, système distribué, microservices)
- Concevoir et implémenter des pipelines CI/CD
- Administrer des utilisateurs et des groupes IAM
- Administrer Amazon Elastic Container Service (Amazon ECS)

- Concevoir une infrastructure réseau AWS (par exemple, Amazon VPC, AWS Direct Connect)
- Comprendre la conformité et les licences

Pour consulter une liste détaillée des outils et technologies spécifiques susceptibles d'être couverts par l'examen, ainsi que la liste des services AWS concernés et hors champ, reportez-vous à l'annexe.

Contenu de l'examen

Types de réponses

L'examen comporte deux types de questions :

- **Questions à choix multiples** : une réponse correcte et trois réponses incorrectes (distracteurs)
- **Questions à réponses multiples** : deux réponses correctes ou plus sur cinq réponses ou plus

Sélectionnez une ou plusieurs réponses qui complètent l'affirmation ou répondent à la question. Les distracteurs, ou réponses incorrectes, sont des options qu'un candidat ayant des connaissances ou compétences incomplètes est susceptible de choisir. Les distracteurs sont généralement des réponses plausibles qui correspondent au thème du contenu.

Les questions sans réponse sont considérées comme étant incorrectes ; les réponses au hasard ne sont pas pénalisées. L'examen comporte : 50 questions qui ont une incidence sur votre score.

Contenu non scoré

L'examen comporte 15 questions non scorées qui n'ont pas d'incidence sur votre score. AWS recueille des informations sur les performances des candidats dans le cadre de ces questions non scorées. L'objectif est d'évaluer ces questions en vue d'une utilisation ultérieure en tant que questions scorées. Ces questions non scorées ne sont pas identifiées comme telles dans l'examen.

Résultats de l'examen

L'examen AWS Certified Developer – Associate (DVA-C01) est un examen éliminatoire. L'examen est évalué en fonction d'une norme minimale établie par les professionnels AWS. Ceux-ci observent les bonnes pratiques et directives en matière de certification.

Vos résultats à l'examen sont présentés sous la forme d'un score gradué de 100 à 1 000. Le score minimal pour réussir est de 720. Votre score indique votre performance lors de l'examen dans son ensemble et si vous avez réussi l'examen. Les modèles de score gradué permettent de mettre en correspondance des scores de différents formulaires d'examen qui peuvent présenter des niveaux de difficulté légèrement différents.

Votre compte-rendu de score peut contenir un tableau des classifications de vos performances à chaque niveau de section. Ces informations sont destinées à fournir un feedback général sur vos performances à l'examen. L'examen utilise un modèle de score compensatoire, ce qui signifie que vous n'avez pas besoin d'obtenir une note minimale dans chaque section. Vous devez seulement réussir l'examen dans son ensemble.

Chaque section de l'examen présente une pondération spécifique. Certaines sections comportent donc plus de questions que d'autres. Le tableau contient des informations générales qui mettent en évidence vos points forts et vos points faibles. Le feedback au niveau des sections doit être interprété avec précaution.

Description du contenu

Ce guide d'examen inclut les pondérations, les domaines couverts par le test, ainsi que les objectifs de l'examen. Il ne s'agit pas d'une liste complète du contenu de l'examen. Cependant, pour chacun des objectifs, un contexte complémentaire est disponible afin de vous aider à préparer votre examen. Le tableau suivant répertorie les principaux domaines du contenu et leurs pondérations. Le tableau est suivi d'un plan complet du contenu de l'examen, avec le contexte complémentaire. Dans chaque domaine, le pourcentage représente uniquement le contenu noté.

Domaine	% de l'examen
Domaine 1 : Déploiement	22 %
Domaine 2 : Sécurité	26 %
Domaine 3 : Développement avec les services AWS	30 %
Domaine 4 : Refactorisation	10 %
Domaine 5 : Surveillance et dépannage	12 %
TOTAL	100 %

Domaine 1 : Déploiement

- 1.1 Déployer du code écrit dans AWS à l'aide des processus, modèles et pipelines CI/CD existants.
 - Valider du code dans un référentiel et invoquer des actions de génération, de test et/ou de déploiement
 - Utiliser des étiquettes et des branches pour la gestion des versions
 - Utiliser AWS CodePipeline pour orchestrer les flux de travail dans différents environnements
 - Appliquer AWS CodeCommit, AWS CodeBuild, AWS CodePipeline, AWS CodeStar et AWS CodeDeploy à des fins CI/CD
 - Exécuter un plan de restauration basé sur une stratégie de déploiement d'applications
- 1.2 Déployer des applications à l'aide d'AWS Elastic Beanstalk
 - Utiliser les environnements pris en charge existants pour définir une nouvelle pile d'applications
 - Créer un package de l'application
 - Introduire une nouvelle version de l'application dans l'environnement Elastic Beanstalk
 - Utiliser une stratégie de déploiement pour déployer une version de l'application (c'est-à-dire, simultané, en continu, en continu par lots, inaltérable)
 - Valider l'état de l'application à l'aide du tableau de bord Elastic Beanstalk
 - Utiliser Amazon CloudWatch Logs pour instrumenter la journalisation des applications
- 1.3 Préparer le package de déploiement de l'application qui sera déployé sur AWS
 - Gérer les dépendances du module de code (variables d'environnement, fichiers de configuration et fichiers image statiques) au sein du package
 - Décrire la structure du répertoire package/conteneur et organiser les fichiers de manière appropriée
 - Traduire les besoins en ressources des applications en paramètres d'infrastructure AWS (par exemple, mémoire, cœurs)

1.4 Déployer des applications sans serveur.

- Dans le cadre d'un cas d'utilisation, implémenter et lancer un modèle AWS SAM (AWS Serverless Application Model)
- Gérer les environnements dans des services AWS individuels (par exemple, différencier les environnements de développement, de test et de production dans Amazon API Gateway)

Domaine 2 : Sécurité

2.1 Réaliser des appels authentifiés vers les services AWS.

- Communiquer la stratégie requise en fonction des moindres privilèges requis par l'application
- Assumer un rôle IAM pour accéder à un service
- Utiliser le fournisseur d'informations d'identification du kit de développement logiciel (SDK) sur site ou dans le cloud pour accéder aux services AWS (informations d'identification locales ou rôles d'instance)

2.2 Implémenter le chiffrement à l'aide des services AWS.

- Chiffrer les données au repos (côté client, côté serveur, chiffrement par enveloppement) à l'aide des services AWS
- Chiffrer les données en transit

2.3 Implémenter une authentification et une autorisation d'application.

- Ajouter une fonction d'inscription et de connexion des utilisateurs pour les applications dotées d'une identité Amazon Cognito ou de groupes d'utilisateurs
- Utiliser les informations d'identification fournies par Amazon Cognito pour écrire du code qui accède aux services AWS.
- Utiliser la synchronisation Amazon Cognito pour synchroniser les profils et les données utilisateur
- Utiliser les identités authentifiées par les développeurs pour permettre l'interaction entre les appareils des utilisateurs finaux, l'authentification backend et Amazon Cognito

Domaine 3 : Développement avec les services AWS

3.1 Écrire du code pour les applications sans serveur.

- Comparer les modèles basés sur serveur et sans serveur et en exposer les différences (par exemple, microservices, nature sans état des applications sans serveur, mise à l'échelle des applications sans serveur et découplage des couches des applications sans serveur)
- Configurer les fonctions AWS Lambda en définissant les variables et les paramètres d'environnement (par exemple, mémoire, délai d'exécution, gestionnaire)
- Créer un point de terminaison d'API à l'aide d'Amazon API Gateway
- Créer et tester des actions d'API appropriées telles que GET et POST à l'aide du point de terminaison d'API
- Appliquer les concepts Amazon DynamoDB (par exemple, les tables, les éléments et les attributs)
- Calculer les unités de capacité de lecture/écriture pour Amazon DynamoDB en fonction des exigences de l'application
- Associer une fonction AWS Lambda à une source d'événements AWS (par exemple, Amazon API Gateway, Amazon CloudWatch, événements Amazon S3, Amazon Kinesis)
- Invoquer une fonction AWS Lambda de manière synchrone et asynchrone

3.2 Traduire les exigences fonctionnelles en éléments de conception d'une application.

- Déterminer le traitement en temps réel par rapport au traitement par lots pour un cas d'utilisation donné
- Déterminer l'utilisation d'une action synchrone ou asynchrone pour un cas d'utilisation donné

- Déterminer l'utilisation d'un événement par rapport à la planification/l'interrogation pour un cas d'utilisation donné
- Prendre en compte les compromis nécessaires pour les modèles de cohérence dans la conception d'une application

3.3 Implémenter la conception d'application dans le code d'application.

- Écrire du code pour utiliser les services de messagerie (par exemple SQS, SNS)
- Utiliser Amazon ElastiCache pour créer un cache de base de données
- Utiliser Amazon DynamoDB pour indexer des objets dans Amazon S3
- Écrire une fonction AWS Lambda sans état
- Écrire une application web avec des serveurs web sans état (externaliser l'état)

3.4 Écrire du code interagissant avec les services AWS à l'aide d'API, de kits SDK et de l'interface de ligne de commande AWS (AWS CLI).

- Choisir les API, kits de développement logiciel (SDK) et commandes CLI appropriées pour les composants de code
- Écrire du code résilient qui traite les échecs ou les exceptions (c.-à-d. nouvelles tentatives avec interruption exponentielle et instabilité)

Domaine 4 : Refactorisation

4.1 Optimiser les applications pour une meilleure utilisation des fonctions et services AWS.

- Implémenter les services de mise en cache AWS pour optimiser les performances (par exemple, Amazon ElastiCache, cache Amazon API Gateway)
- Appliquer un schéma de dénomination Amazon S3 pour des performances de lecture optimales

4.2 Migrer le code d'application existant pour l'exécuter sur AWS.

- Isoler les dépendances
- Exécuter l'application comme un ou plusieurs processus sans état
- Développer pour permettre une capacité de mise à l'échelle horizontale
- Externaliser l'état

Domaine 5 : Surveillance et dépannage

5.1 Écrire du code pouvant être surveillé.

- Créer des métriques personnalisées Amazon CloudWatch
- Effectuer la journalisation d'une manière disponible pour les opérateurs système
- Instrumenter le code source de l'application pour activer le suivi dans AWS X-Ray

5.2 Analyser les causes premières des pannes détectées lors des tests ou de la production.

- Interpréter les sorties du mécanisme de journalisation dans AWS pour identifier les erreurs dans les journaux
- Vérifier l'historique de génération et de test dans les services AWS (par exemple, AWS CodeBuild, AWS CodeDeploy, AWS CodePipeline) pour identifier les problèmes
- Utiliser les services AWS (par exemple Amazon CloudWatch, les journaux de flux VPC et AWS X-Ray) pour localiser un composant défectueux spécifique

Annexe

Quels outils, technologies et concepts clés sont susceptibles d'être abordés lors de l'examen ?

Voici une liste non exhaustive des outils et technologies susceptibles de figurer dans l'examen. Sous réserve de modification, cette liste a pour but de vous aider à comprendre la portée générale des services, fonctions ou technologies abordés dans le cadre de l'examen. Les outils et technologies généraux de cette liste n'apparaissent pas dans un ordre particulier. Les services AWS sont regroupés selon leurs fonctions principales. Certaines de ces technologies seront probablement davantage couvertes que d'autres lors de l'examen. Cependant, leur ordre et leur position dans cette liste ne sont pas une indication de pondération ni d'importance relative :

- Analytique
- Intégration d'applications
- Conteneurs
- Gestion des coûts et de la capacité
- Mouvement des données
- Outils pour développeurs
- Instances (machines virtuelles)
- Gestion et gouvernance
- Réseaux et diffusion de contenu
- Sécurité
- Sans serveur

Services et fonctions AWS

Analytique :

- Amazon Elasticsearch Service (Amazon ES)
- Amazon Kinesis

Intégration d'applications :

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

Calcul :

- Amazon EC2
- AWS Elastic Beanstalk
- AWS Lambda

Conteneurs :

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

Base de données :

- Amazon DynamoDB
- Amazon ElastiCache
- Amazon RDS

Outils pour développeurs :

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

Gestion et gouvernance :

- AWS CloudFormation
- Amazon CloudWatch

Réseaux et diffusion de contenu :

- Amazon API Gateway
- Amazon CloudFront
- Elastic Load Balancing

Sécurité, identité et conformité :

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

Stockage :

- Amazon S3

Services et fonctions AWS non concernés

Vous trouverez ci-dessous une liste non exhaustive des services et fonctions AWS qui ne sont pas abordés dans l'examen. Ces services et fonctions ne représentent pas toutes les offres AWS exclues du contenu de l'examen. Les services ou fonctions qui ne sont aucunement liés aux fonctions cible pour l'examen sont exclus de cette liste car ils sont considérés comme non pertinents.

Les services et fonctions AWS non concernés sont les suivants :

- 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