

## Guide de l'examen AWS Certified Developer - Associate (DVA-C02)

### Introduction

L'examen AWS Certified Developer – Associate (DVA-C02) est destiné aux personnes ayant un rôle de développeur. L'examen valide la capacité d'un candidat à démontrer ses compétences en développement, en test, en déploiement et en débogage d'applications basées sur le Cloud AWS.

L'examen valide également la capacité d'un candidat à effectuer les tâches suivantes :

- Développer et optimiser des applications sur AWS
- Créer des packages et les déployer à l'aide de flux de travail d'intégration continue et de livraison continue (CI/CD)
- Sécuriser le code et les données d'application
- Identifier et résoudre les problèmes des applications

### Description du candidat cible

Le candidat cible doit posséder au moins 1 an d'expérience pratique dans le développement et la maintenance d'applications à l'aide des services AWS.

#### Connaissances informatiques générales recommandées

Le candidat cible doit avoir les connaissances générales suivantes en matière d'informatique :

- Maîtriser au moins un langage de programmation de haut niveau
- Comprendre la gestion du cycle de vie des applications
- Avoir une compréhension de base des applications natives cloud pour écrire du code
- Pouvoir développer des applications fonctionnelles
- Savoir utiliser les outils de développement

## Connaissances AWS recommandées

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

- Développer et sécuriser des applications à l'aide des API de service AWS, de la CLI AWS et des SDK
- Utiliser un pipeline CI/CD pour déployer des applications sur AWS

## Tâches n'entrant pas dans le cadre des exigences pour le candidat cible

La liste suivante contient les tâches que le candidat cible n'est pas censé être en mesure d'effectuer. Cette liste n'est pas exhaustive. Ces tâches sont hors du champ de l'examen :

- Concevoir des architectures (par exemple, systèmes distribués, microservices, schémas de bases de données et modélisation)
- Concevoir et créer des pipelines CI/CD
- Administrer des utilisateurs et des groupes IAM
- Administrer des serveurs et des systèmes d'exploitation
- Concevoir une infrastructure réseau AWS (par exemple, Amazon VPC, AWS Direct Connect)

Vous trouverez en annexe une liste de technologies et de concepts susceptibles de figurer dans l'examen, une liste des services et fonctions AWS inclus dans le champ de l'examen et une liste des services et fonctions AWS exclus du champ de l'examen.

## Contenu de l'examen

### Types de réponses

L'examen comporte deux types de questions :

- **Choix multiple** : une réponse correcte et trois réponses incorrectes (réponses piège)
- **Réponse multiple** : plusieurs réponses correctes sur cinq options de réponse ou plus

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

Les questions sans réponse sont notées comme incorrectes ; aucune pénalité n'est appliquée lorsque le candidat devine une réponse. L'examen comporte 50 questions qui ont une incidence sur votre score.

### **Contenu non noté**

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

### **Résultats de l'examen**

L'examen AWS Certified Developer - Associate (DVA-C02) est de type « réussite ou échec ». L'examen est évalué selon une norme minimale établie par les professionnels d'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 vos performances 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 au niveau de chaque section. L'examen utilise un modèle de notation 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 des classifications contient des informations générales qui mettent en évidence vos points forts et vos points faibles. Les commentaires au niveau des sections doivent être interprétés avec précaution.

## **Aperçu du contenu**

Ce guide de l'examen inclut les pondérations, les domaines du contenu, ainsi que les énoncés de tâche de l'examen. Ce guide ne fournit pas une liste exhaustive du contenu de l'examen. Cependant, chaque énoncé de tâche est accompagné d'un contexte supplémentaire pour vous aider à vous préparer à l'examen.

L'examen comporte les domaines de contenu et les pondérations suivants :

- Domaine 1 : Développement avec les services AWS (32 % du contenu évalué)
- Domaine 2 : Sécurité (26 % du contenu évalué)
- Domaine 3 : Déploiement (24 % du contenu évalué)
- Domaine 4 : Dépannage et optimisation (18 % du contenu évalué)

## **Domaine 1 : Développement avec les services AWS**

Énoncé de tâche 1 : Développez du code pour des applications hébergées sur AWS.

Connaissance des domaines suivants :

- Modèles architecturaux (par exemple, orientés événements, microservices, monolithiques, chorégraphie, orchestration, ramifiés)
- Idempotence
- Différences entre les concepts avec état et sans état
- Différences entre les composants étroitement couplés et faiblement couplés
- Modèles de conception tolérants aux pannes (par exemple, nouvelles tentatives avec backoff exponentiel et fluctuations, files d'attente de lettres mortes)
- Différences entre les modèles synchrones et asynchrones

Compétences dans les domaines suivants :

- Créer des applications résilientes et tolérantes aux pannes dans un langage de programmation (par exemple, Java, C#, Python, JavaScript, TypeScript, Go)
- Créer, étendre et gérer des API (par exemple, transformations de réponse/demande, application de règles de validation, remplacement des codes d'état)
- Rédiger et exécuter des tests unitaires dans des environnements de développement (par exemple, en utilisant AWS Serverless Application Model [AWS SAM])
- Écrire du code pour utiliser les services de messagerie
- Écrire du code interagissant avec les services AWS à l'aide d'API et de SDK AWS
- Gérer le flux de données à l'aide des services AWS

Énoncé de tâche 2 : Développez du code pour AWS Lambda.

Connaissance des domaines suivants :

- Mappage de source d'événement
- Applications sans état
- Tests unitaires
- Architectures orientées événements
- Capacité de mise à l'échelle
- Accès aux ressources privées dans les VPC à partir du code Lambda

Compétences dans les domaines suivants :

- Configurer des fonctions Lambda en définissant des variables et des paramètres d'environnement (par exemple, mémoire, simultanéité, délai d'inactivité, runtime, gestionnaire, couches, extensions, déclencheurs, destinations)
- Gérer le cycle de vie des événements et les erreurs en utilisant du code (par exemple, destinations Lambda, files d'attente de lettres mortes)
- Écrire et exécuter du code de test à l'aide des services et des outils AWS
- Intégrer des fonctions Lambda aux services AWS
- Régler des fonctions Lambda pour des performances optimales

Énoncé de tâche 3 : Utilisez les magasins de données dans le développement d'applications.

Connaissance des domaines suivants :

- Bases de données relationnelles et non relationnelles
- Opérations de création, de lecture, de mise à jour et de suppression (CRUD)
- Clés de partition à cardinalité élevée pour un accès équilibré aux partitions
- Options de stockage dans le cloud (par exemple, fichiers, objets, bases de données)
- Modèles de cohérence des bases de données (par exemple, très cohérents, cohérents à terme)
- Différences entre les opérations de requête et d'analyse
- Clés et indexation Amazon DynamoDB
- Stratégies de mise en cache (par exemple, écriture directe, lecture directe, chargement différé, TTL)
- Niveaux Amazon S3 et gestion du cycle de vie
- Différences entre les modèles de stockage de données éphémères et persistants

Compétences dans les domaines suivants :

- Sérialiser et désérialiser des données pour assurer la persistance d'un magasin de données
- Utiliser, gérer et maintenir des magasins de données
- Gérer les cycles de vie des données
- Utiliser des services de mise en cache des données

## **Domaine 2 : Sécurité**

Énoncé de tâche 1 : Mettez en œuvre l'authentification et/ou l'autorisation pour les applications et les services AWS.

Connaissance des domaines suivants :

- Fédération d'identité (par exemple, Security Assertion Markup Language [SAML], OpenID Connect [OIDC], Amazon Cognito)
- Jetons au porteur (par exemple, JSON Web Token [JWT], OAuth, AWS Security Token Service [AWS STS])

- Comparaison des groupes d'utilisateurs et des groupes d'identités dans Amazon Cognito
- Politiques basées sur les ressources, politiques de service et politiques principales
- Contrôle d'accès basé sur les rôles (RBAC)
- Autorisation d'application utilisant des ACL
- Principe du moindre privilège
- Différences entre les politiques gérées par AWS et les politiques gérées par le client
- Gestion des identités et des accès

Compétences dans les domaines suivants :

- Utiliser un fournisseur d'identité pour mettre en œuvre un accès fédéré (par exemple, Amazon Cognito, AWS Identity and Access Management [IAM])
- Sécuriser des applications en utilisant des jetons au porteur
- Configurer l'accès programmatique à AWS
- Réaliser des appels authentifiés vers les services AWS
- Assumer un rôle IAM
- Définir des autorisations pour les principaux

Énoncé de tâche 2 : Mettez en œuvre le chiffrement à l'aide des services AWS.

Connaissance des domaines suivants :

- Chiffrement au repos et en transit
- Gestion des certificats (par exemple, AWS Private Certificate Authority)
- Protection des clés (par exemple, rotation des clés)
- Différences entre le chiffrement côté client et le chiffrement côté serveur
- Différences entre les clés AWS Key Management Service (AWS KMS) gérées par AWS et gérées par le client

Compétences dans les domaines suivants :

- Utiliser des clés de chiffrement pour chiffrer ou déchiffrer les données
- Générer des certificats et des clés SSH à des fins de développement
- Utiliser le chiffrement au-delà des limites du compte
- Activer et désactiver la rotation des clés

Énoncé de tâche 3 : Gérez les données sensibles dans le code d'application.

Connaissance des domaines suivants :

- Classification des données (par exemple, informations d'identification personnelle [PII], informations de santé protégées [PHI])
- Variables d'environnement
- Gestion des secrets (par exemple, AWS Secrets Manager, AWS Systems Manager Parameter Store)
- Gestion sécurisée des justificatifs

Compétences dans les domaines suivants :

- Chiffrer des variables d'environnement contenant des données sensibles
- Utiliser des services de gestion des secrets pour sécuriser les données sensibles
- Nettoyer les données sensibles

### **Domaine 3 : Déploiement**

Énoncé de tâche 1 : Préparez les artefacts d'application à déployer sur AWS.

Connaissance des domaines suivants :

- Méthodes d'accès aux données de configuration des applications (par exemple, AWS AppConfig, Secrets Manager, Parameter Store)
- Package, couches et options de configuration du déploiement Lambda
- Outils de gestion des versions basés sur Git (par exemple, Git, AWS CodeCommit)
- Images de conteneurs

Compétences dans les domaines suivants :

- Gérer les dépendances du module de code (par exemple, variables d'environnement, fichiers de configuration, images de conteneur) dans le package
- Organiser la structure de fichiers et d'annuaires pour le déploiement des applications
- Utiliser des annuaires de code dans les environnements de déploiement
- Appliquer des exigences d'applications pour les ressources (par exemple, mémoire, cœurs)



Énoncé de tâche 2 : Testez les applications dans les environnements de développement.

Connaissance des domaines suivants :

- Fonctions des services AWS qui effectuent le déploiement d'applications
- Tests d'intégration utilisant des points de terminaison fictifs
- Versions et alias Lambda

Compétences dans les domaines suivants :

- Tester le code déployé à l'aide des services et des outils AWS
- Effectuer une intégration fictive pour les API et résoudre les dépendances d'intégration
- Tester des applications à l'aide de points de terminaison de développement (par exemple, configuration des étapes dans Amazon API Gateway)
- Déployer des mises à jour de la pile d'applications dans des environnements existants (par exemple, déploiement d'un modèle AWS SAM dans un environnement intermédiaire différent)

Énoncé de tâche 3 : Automatisez les tests de déploiement.

Connaissance des domaines suivants :

- Étapes API Gateway
- Branches et actions dans le flux de travail d'intégration continue et de livraison continue (CI/CD)
- Tests logiciels automatisés (par exemple, tests unitaires, tests fictifs)

Compétences dans les domaines suivants :

- Créer des événements de test d'application (par exemple, des charges utiles JSON pour tester les ressources Lambda, API Gateway, AWS SAM)
- Déployer des ressources d'API dans divers environnements
- Créer des environnements d'applications utilisant des versions approuvées pour les tests d'intégration (par exemple, alias Lambda, identifications d'images de conteneurs, branches AWS Amplify, environnements AWS Copilot)
- Mettre en œuvre et déployer des modèles d'Infrastructure as Code (IaC) (par exemple, modèles AWS SAM, modèles AWS CloudFormation)

- Gérer des environnements dans les services AWS individuels (par exemple, différenciation entre le développement, le test et la production dans API Gateway)

Énoncé de tâche 4 : Déployez du code à l'aide des services AWS CI/CD.

Connaissance des domaines suivants :

- Outils de gestion des versions basés sur Git (par exemple, Git, AWS CodeCommit)
- Approbations manuelles et automatisées dans AWS CodePipeline
- Accès aux configurations des applications à partir d'AWS AppConfig et de Secrets Manager
- Flux de travail CI/CD qui utilisent les services AWS
- Déploiement d'applications utilisant les services et les outils AWS (par exemple, CloudFormation, AWS Cloud Development Kit [AWS CDK], AWS SAM, AWS CodeArtifact, AWS Copilot, Amplify, Lambda)
- Options de package de déploiement Lambda
- Étapes API Gateway et domaines personnalisés
- Stratégies de déploiement (par exemple, canary, bleu/vert, en continu)

Compétences dans les domaines suivants :

- Mettre à jour des modèles IaC existants (par exemple, modèles AWS SAM, modèles CloudFormation)
- Gérer des environnements d'applications à l'aide des services AWS
- Déployer une version de l'application en utilisant des stratégies de déploiement
- Valider du code dans un référentiel pour appeler des actions de génération, de test et de déploiement
- Utiliser des flux de travail orchestrés pour déployer du code dans différents environnements
- Restaurer des applications en utilisant des stratégies de déploiement existantes
- Utiliser des étiquettes et des branches pour la gestion des versions et des éditions

- Utiliser des configurations de runtime existantes pour créer des déploiements dynamiques (par exemple, en utilisant des variables d'étape d'API Gateway dans les fonctions Lambda)

## **Domaine 4 : Dépannage et optimisation**

Énoncé de tâche 1 : Participez à une analyse des causes profondes.

Connaissance des domaines suivants :

- Systèmes de journalisation et de surveillance
- Langages pour les interrogations de journaux (par exemple, Amazon CloudWatch Logs Insights)
- Visualisations des données
- Outils d'analyse de code
- Codes d'erreur HTTP courants
- Exceptions courantes générées par les SDK
- Cartes de service dans AWS X-Ray

Compétences dans les domaines suivants :

- Déboguer du code pour identifier les défauts
- Interpréter les métriques, les journaux et les traces des applications
- Interroger des journaux pour trouver des données pertinentes
- Mettre en œuvre des métriques personnalisées (par exemple, le format de métrique intégré CloudWatch [EMF])
- Examiner l'état des applications à l'aide de tableaux de bord et d'insights
- Dépanner des échecs de déploiement à l'aide des journaux de sortie

Énoncé de tâche 2 : Instrumentez du code pour l'observabilité.

Connaissance des domaines suivants :

- Suivi distribué
- Différences entre la journalisation, la surveillance et l'observabilité
- Journalisation structurée
- Métriques d'application (par exemple, personnalisées, incorporées, intégrées)

Compétences dans les domaines suivants :

- Mettre en œuvre une stratégie de journalisation efficace pour enregistrer le comportement et l'état des applications
- Mettre en œuvre un code qui émet des métriques personnalisées
- Ajouter des annotations pour les services de suivi
- Mettre en œuvre des alertes de notification pour des actions spécifiques (par exemple, des notifications concernant les limites de quota ou la fin du déploiement)
- Mettre en œuvre un suivi à l'aide des services et des outils AWS

Énoncé de tâche 3 : Optimisez les applications en utilisant les services et les fonctions AWS.

Connaissance des domaines suivants :

- Mise en cache
- Simultanéité
- Services de messagerie (par exemple, Amazon Simple Queue Service [Amazon SQS], Amazon Simple Notification Service [Amazon SNS])

Compétences dans les domaines suivants :

- Profiler les performances de l'application
- Déterminer la mémoire et la puissance de calcul minimales pour une application
- Utiliser des politiques de filtrage des abonnements pour optimiser la messagerie
- Mettre en cache le contenu en fonction des en-têtes de requête

## Annexe

### Technologies et concepts susceptibles de figurer dans l'examen

La liste suivante contient les technologies et les concepts susceptibles de figurer dans l'examen. Cette liste n'est pas exhaustive et peut faire l'objet de modifications. L'ordre et l'emplacement des éléments de cette liste ne constituent pas une indication de leur pondération ou de leur importance relative dans le cadre de l'examen :

- Analytique
- Intégration d'applications
- Calcul
- Conteneurs
- Gestion des coûts et de la capacité
- Base de données
- Outils pour développeur
- Management et gouvernance
- Mise en réseau et diffusion de contenu
- Sécurité, identité et conformité
- Stockage

### Services et fonctions AWS inclus dans le champ de l'examen

La liste suivante contient les services et fonctions AWS qui font partie du champ de l'examen. Cette liste n'est pas exhaustive et peut faire l'objet de modifications. Les offres AWS apparaissent dans des catégories correspondant à leurs fonctions principales :

Analytique :

- Amazon Athena
- Amazon Kinesis
- Amazon OpenSearch Service

Intégration d'applications :

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

- Amazon Simple Queue Service (Amazon SQS)
- AWS Step Functions

Calcul :

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

Conteneurs :

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

Base de données :

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

Outils pour développeur :

- 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

#### Management et gouvernance :

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

#### Mise en réseau et diffusion de contenu :

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

#### Sécurité, identité et conformité :

- 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

#### Stockage :

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

## Services et fonctions AWS hors du champ de l'examen

La liste suivante contient les services et fonctions AWS qui sont hors du champ de l'examen. Cette liste n'est pas exhaustive et peut faire l'objet de modifications. Les offres AWS qui ne sont aucunement liées aux rôles cibles de l'examen sont exclues de cette liste :

### Analytique :

- Amazon QuickSight

### Applications métier :

- Amazon Chime
- Amazon Connect
- Amazon WorkMail

### Informatique pour l'utilisateur final :

- Amazon AppStream 2.0
- Amazon WorkSpaces

### Applications web et mobiles front-end :

- AWS Device Farm

### Jeux :

- Amazon GameLift

### Machine Learning :

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

### Management et gouvernance :

- AWS Managed Services (AMS)
- AWS Service Catalog



#### Services multimédias :

- Amazon Elastic Transcoder

#### Migration et transfert :

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

#### Sécurité, identité et conformité :

- AWS Shield Advanced
- AWS Shield Standard

#### Stockage :

- Gamme AWS Snow
- AWS Storage Gateway

#### Enquête

Ce guide de l'examen a-t-il été utile ? Dites-nous ce que vous en pensez en [répondant à notre enquête](#).