

Prüfungsleitfaden für AWS Certified Developer - Associate (DVA-C02)

Einleitung

Die Prüfung „AWS Certified Developer - Associate (DVA-C02)“ ist für Personen gedacht, die eine Entwicklerrolle ausüben. Die Prüfung bewertet die Kenntnisse des Kandidaten in der Entwicklung, Testung, Bereitstellung und Fehlerbehebung AWS-Cloud-basierter Anwendungen.

Zudem bestätigt die Prüfung einem Kandidaten die Fähigkeit, die folgenden Aufgaben auszuführen:

- Entwickeln und Optimieren von Anwendungen auf AWS.
- Verpacken und Bereitstellen von Anwendungen mithilfe von CI/CD-Workflows (Continuous Integration und Continuous Delivery).
- Sichern von Anwendungscode und Daten.
- Identifizieren und Lösen von Anwendungsproblemen.

Beschreibung der Zielkandidaten

Zielkandidaten sollten über mindestens 1 Jahr praktische Erfahrung in der Entwicklung und Wartung von Anwendungen mithilfe von AWS-Services verfügen.

Empfohlene allgemeine IT-Kenntnisse

Die Zielkandidaten sollten folgende allgemeine IT-Kenntnisse haben:

- Fortgeschrittene Kenntnisse mindestens einer höheren Programmiersprache
- Verständnis des Anwendungslebenszyklus-Managements
- Grundlegende Kenntnisse von cloudnativen Anwendungen zum Schreiben von Code
- Kenntnisse in der Entwicklung funktionaler Anwendungen
- Erfahrung im Umgang mit Entwicklungstools

Empfohlenes AWS-Wissen

Die Zielkandidaten sollten folgende Aufgaben ausführen können:

- Entwickeln und Sichern von Anwendungen mithilfe von AWS-Service-APIs, der AWS-CLI und SDKs.
- Nutzen einer CI/CD-Pipeline zur Bereitstellung von Anwendungen auf AWS.

Aufgaben, die für Zielkandidaten nicht relevant sind

Die folgende Liste enthält Aufgaben, deren Ausführung von Zielkandidaten nicht erwartet wird. Diese Liste ist nicht vollständig. Diese Aufgaben sind für die Prüfung nicht relevant:

- Entwickeln von Architekturen (z. B. verteilte Systeme, Microservices, Datenbankschemas und Modellierung).
- Entwickeln und Erstellen von CI/CD-Pipelines.
- Verwalten von IAM-Benutzern und -Gruppen.
- Verwalten von Servern und Betriebssystemen.
- Entwickeln einer AWS-Netzwerkinfrastruktur (z. B. Amazon VPC, AWS Direct Connect).

Im Anhang findest du eine Liste der Technologien und Konzepte, die in der Prüfung abgefragt werden könnten, eine Liste der abgedeckten AWS-Services und -Funktionen sowie eine Liste der nicht abgedeckten AWS-Services und -Funktionen.

Prüfungsinhalte

Antworttypen

Es gibt zwei Arten von Fragen bei der Prüfung:

- **Multiple Choice:** beinhaltet eine richtige und drei falsche Antworten (Distraktoren)
- **Mehrfachantworten:** beinhaltet zwei oder mehr korrekte Antworten aus fünf oder mehr Antwortoptionen

Wähle eine oder mehrere Antworten aus, die die Aussage am besten beschreiben oder die Frage beantworten. Distraktoren oder falsche Antworten sind Antwortoptionen, für die sich ein Kandidat mit unzureichendem Wissen oder fehlenden Fähigkeiten entscheiden könnte. Distraktoren sind im Allgemeinen plausible Antworten, die zum Inhalt passen.

Unbeantwortete Fragen werden als falsch gewertet. Raten ist ohne Abzüge möglich. Die Prüfung umfasst 50 Fragen, aus denen sich deine Punktzahl ergibt.

Nicht gewertete Inhalte

Die Prüfung umfasst 15 nicht gewertete Fragen, die sich nicht auf deine Punktzahl auswirken. AWS sammelt mit diesen nicht gewerteten Fragen Informationen zur Leistung, um diese in Zukunft als gewertete Fragen zu verwenden. Diese nicht gewerteten Fragen werden in der Prüfung nicht ausgewiesen.

Prüfungsergebnisse

Für die Prüfung AWS Certified Developer - Associate (DVA-C02) werden die Bezeichnungen „Bestanden“ und „Nicht bestanden“ verwendet. Die Prüfung wird nach einem Mindeststandard bewertet, der von AWS-Experten festgelegt wurde, die sich an bewährten Methoden und Richtlinien der Zertifizierungsbranche orientieren.

Deine Ergebnisse für die Prüfung werden als skalierte Punktzahl von 100 bis 1.000 angezeigt. Die erforderliche Mindestpunktzahl beträgt 720. Deine Punktzahl spiegelt deine Leistung wider und gibt an, ob du bestanden hast oder nicht. Skalierte Bewertungsmodelle helfen dabei, Ergebnisse mehrerer Prüfungsformen zu vergleichen, die möglicherweise leicht unterschiedliche Schwierigkeitsgrade haben.

Dein Bewertungsbericht enthält eine Tabelle mit Klassifizierungen deiner Leistung auf jeder Abschnittsebene. Für die Prüfung wird ein kompensatorisches Bewertungsmodell verwendet, was bedeutet, dass du nicht in jedem Abschnitt die notwendige Punktzahl erreichen musst. Du musst nur die Gesamtprüfung bestehen.

Jeder Abschnitt der Prüfung hat eine bestimmte Gewichtung. Daher enthalten einige Abschnitte mehr Fragen als andere. Die Tabelle mit Klassifizierungen enthält allgemeine Informationen, die deine Stärken und Schwächen hervorheben. Bewerte das Feedback auf Abschnittsebene mit Bedacht.

Inhaltsverzeichnis

Dieser Prüfungsleitfaden enthält Gewichtungen, Inhaltsdomänen und Aufgabenstellungen für die Prüfung. Dieser Leitfaden enthält keine umfassende Liste der Prüfungsinhalte. Für jede Aufgabenstellung steht jedoch zusätzlicher Kontext zur Verfügung, der dir bei der Vorbereitung auf die Prüfung hilft.

Die Prüfung hat die folgenden Inhaltsdomänen und Gewichtungen:

- Domain 1: Entwicklung mithilfe der AWS-Services (32 % der gewerteten Inhalte)
- Domain 2: Sicherheit (26 % der gewerteten Inhalte)
- Domain 3: Bereitstellung (24 % der gewerteten Inhalte)
- Domain 4: Problembehandlung und Optimierung (18 % der gewerteten Inhalte)

Domain 1: Entwicklung mithilfe der AWS-Services

Aufgabenstellung 1: Entwickle Code für auf AWS gehostete Anwendungen.

Wissen über:

- Architektonische Muster (z. B. ereignisgesteuert, Microservices, monolithisch, Choreographie, Orchestrierung, Fanout)
- Idempotenz
- Unterschiede zwischen zustandsbehafteten und zustandslosen Konzepten
- Unterschiede zwischen eng und lose gekoppelten Komponenten
- Fehlertolerante Entwurfsmuster (z. B. Wiederholungsversuche mit exponentiellem Backoff und Jitter, Warteschlangen für unzustellbare Nachrichten)
- Unterschiede zwischen synchronen und asynchronen Mustern

Fähigkeiten im:

- Erstellen fehlertoleranter und resilienter Anwendungen in einer Programmiersprache (z. B. Java, C#, Python, JavaScript, TypeScript, Go)
- Erstellen, Erweitern und Verwalten von APIs (z. B. Antwort-/Anforderungstransformationen, Durchsetzen von Validierungsregeln, Überschreiben von Statuscodes)

- Schreiben und Ausführen von Einheitentests in Entwicklungsumgebungen (z. B. mit AWS Serverless Application Model [AWS SAM])
- Schreiben von Code zur Verwendung von Messaging-Diensten
- Schreiben von mit AWS-Services interagierendem Code mithilfe von APIs und AWS-SDKs
- Umgang mit Datenstreaming mithilfe von AWS-Services

Aufgabenstellung 2: Entwickle Code für AWS Lambda.

Wissen über:

- Ereignisquellen-Mapping
- Zustandslose Anwendungen
- Einheitentests
- Ereignisgesteuerte Architektur
- Skalierbarkeit
- Zugriff auf private Ressourcen in VPCs vom Lambda-Code aus

Fähigkeiten im:

- Konfigurieren von Lambda-Funktionen durch Definieren von Umgebungsvariablen und -parametern (z. B. Speicher, Nebenläufigkeit, Timeout, Laufzeit, Handler, Ebenen, Erweiterungen, Auslöser, Ziele)
- Umgang mit dem Ereignislebenszyklus und Fehlern mithilfe von Code (z. B. Lambda-Ziele, Warteschlangen für unzustellbare Nachrichten)
- Schreiben und Ausführen von Testcode mithilfe von AWS-Services und -Tools
- Integrieren von Lambda-Funktionen in AWS-Services
- Optimieren von Lambda-Funktionen für bestmögliche Leistung

Aufgabenstellung 3: Verwende Datenspeicher in der Anwendungsentwicklung.

Wissen über:

- Relationale und nicht relationale Datenbanken
- CRUD-Operationen (create, read, update, delete – erstellen, lesen, aktualisieren, löschen)
- Partitionsschlüssel mit hoher Kardinalität für ausgeglichenen Partitionszugriff
- Cloud-Speicheroptionen (z. B. Datei, Objekt, Datenbanken)

- Datenbankkonsistenzmodelle (z. B. stark konsistent, letztendlich konsistent)
- Unterschiede zwischen Abfrage- und Scanoperationen
- Amazon-DynamoDB-Schlüssel und -Indizierung
- Caching-Strategien (z. B. Write-Through, Read-Through, Lazy Loading, TTL)
- Amazon-S3-Stufen und Lebenszyklusmanagement
- Unterschiede zwischen kurzzeitigen und persistenten Datenspeichermustern

Fähigkeiten im:

- Serialisieren und Deserialisieren von Daten, um einem Datenspeicher Persistenz zu verleihen
- Verwenden, Verwalten und Pflegen von Datenspeichern
- Verwalten von Datenlebenszyklen
- Verwenden von Daten-Caching-Diensten

Domain 2: Sicherheit

Aufgabenstellung 1: Implementiere Authentifizierung und/oder Autorisierung für Anwendungen und AWS-Services.

Wissen über:

- Identitätsverbund (z. B. Security Assertion Markup Language [SAML], OpenID Connect [OIDC], Amazon Cognito)
- Träger-Token (z. B. JSON Web Token [JWT], OAuth, AWS Security Token Service [AWS STS])
- Vergleich von Benutzerpools und Identitätspools in Amazon Cognito
- Ressourcenbasierte Richtlinien, Servicerichtlinien und Hauptrichtlinien
- Rollenbasierte Zugriffskontrolle
- Anwendungsautorisierung, die ACLs verwendet
- Prinzip der geringsten Privilegien
- Unterschiede zwischen von AWS verwalteten Richtlinien und vom Kunden verwalteten Richtlinien
- Identity and Access Management

Fähigkeiten im:

- Verwenden eines Identitätsanbieters zur Implementierung eines Verbundzugriffs (z. B. Amazon Cognito, AWS Identity and Access Management [IAM])
- Sichern von Anwendungen mithilfe von Träger-Token
- Konfigurieren eines programmgesteuerten Zugriffs auf AWS
- Tätigen authentifizierter Anrufe an AWS-Services
- Übernehmen einer IAM-Rolle
- Definieren von Berechtigungen für Prinzipale

Aufgabenstellung 2: Implementiere eine Verschlüsselung mithilfe von AWS-Services.

Wissen über:

- Verschlüsselung im Ruhezustand und während der Übertragung
- Zertifikatsverwaltung (z. B. AWS Private Certificate Authority)
- Schlüsselschutz (z. B. Schlüsselrotation)
- Unterschiede zwischen clientseitiger und serverseitiger Verschlüsselung
- Unterschiede zwischen von AWS verwalteten und vom Kunden verwalteten AWS Key Management Service (AWS-KMS)-Schlüsseln

Fähigkeiten im:

- Verwenden von Verschlüsselungsschlüsseln zum Ver- oder Entschlüsseln von Daten
- Generieren von Zertifikaten und SSH-Schlüsseln für Entwicklungszwecke
- Verwenden der Verschlüsselung über Kontogrenzen hinweg
- Aktivieren und Deaktivieren einer Schlüsselrotation

Aufgabenstellung 3: Verwalte sensible Daten im Anwendungscode.

Wissen über:

- Datenklassifizierung (z. B. persönlich identifizierbare Informationen [PII], geschützte Gesundheitsinformationen [PHI])
- Umgebungsvariablen
- Secret-Management (z. B. AWS Secrets Manager, AWS Systems Manager-Parameterspeicher)
- Sicherer Umgang mit Anmeldeinformationen

Fähigkeiten im:

- Verschlüsseln von Umgebungsvariablen, die sensible Daten enthalten
- Verwenden von Secret-Managementdiensten zur Sicherung sensibler Daten
- Bereinigen sensibler Daten

Domain 3: Bereitstellung

Aufgabenstellung 1: Bereite Anwendungsartefakte für die Bereitstellung in AWS vor.

Wissen über:

- Möglichkeiten für den Zugriff auf Konfigurationsdaten (z. B. AWS AppConfig, Secrets Manager, Parameterspeicher)
- Lambda-Bereitstellungspakete, -Ebenen und -Konfigurationsoptionen
- Git-basierte Tools zur Versionskontrolle (z. B. Git, AWS CodeCommit)
- Container-Images

Fähigkeiten im:

- Verwalten der Abhängigkeiten des Code-Moduls (z. B. Umgebungsvariablen, Konfigurationsdateien, Container-Images) innerhalb des Pakets
- Organisieren von Dateien und einer Verzeichnisstruktur für die Anwendungsbereitstellung
- Verwenden von Code-Repositories in Bereitstellungsumgebungen
- Anwenden von Anwendungsanforderungen für Ressourcen (z. B. Speicher, Kerne)

Aufgabenstellung 2: Teste Anwendungen in Entwicklungsumgebungen.

Wissen über:

- Funktionen in AWS-Services für die Anwendungsbereitstellung
- Integrationstests mit Probeendpunkten
- Lambda-Versionen und Aliasse

Fähigkeiten im:

- Testen von bereitgestelltem Code mithilfe von AWS-Services und -Tools
- Durchführen einer Probeintegration für APIs und Auflösen von Integrationsabhängigkeiten
- Testen von Anwendungen mithilfe von Entwicklungsendpunkten (z. B. Konfigurieren von Phasen in Amazon API-Gateway)

- Bereitstellen von Updates für Anwendungs-Stacks in vorhandenen Umgebungen (z. B. Bereitstellen einer AWS-SAM-Vorlage in einer anderen Staging-Umgebung)

Aufgabenstellung 3: Automatisiere Bereitstellungstests.

Wissen über:

- API-Gateway-Phasen
- Branches und Aktionen im CI/CD-Workflow (Continuous Integration und Continuous Delivery)
- Automatisierte Softwaretests (z. B. Einheitentests, Probetests)

Fähigkeiten im:

- Erstellen von Anwendungstestereignissen (z. B. JSON-Nutzlast zum Testen von Lambda, API-Gateway, AWS-SAM-Ressourcen)
- Bereitstellen von API-Ressourcen in verschiedenen Umgebungen
- Erstellen von Anwendungsumgebungen, die genehmigte Versionen für Integrationstests verwenden (z. B. Lambda-Aliasse, Container-Image-Tags, AWS-Amplify-Branches, AWS-Copilot-Umgebungen)
- Implementieren und Bereitstellen von Infrastructure as Code (IaC)-Vorlagen (z. B. AWS-SAM-Vorlagen, AWS-CloudFormation-Vorlagen)
- Verwalten von Umgebungen in einzelnen AWS-Services (z. B. Unterscheidung zwischen Entwicklung, Test und Produktion in API-Gateway)

Aufgabenstellung 4: Stelle einen Code mithilfe von AWS-CI/CD-Diensten bereit.

Wissen über:

- Git-basierte Tools zur Versionskontrolle (z. B. Git, AWS CodeCommit)
- Manuelle und automatisierte Genehmigungen in AWS CodePipeline
- Zugriff auf Anwendungskonfigurationen von AWS AppConfig und Secrets Manager aus
- CI/CD-Workflows, die AWS-Services nutzen
- Anwendungsbereitstellung, die AWS-Services und -Tools verwendet (z. B. CloudFormation, AWS Cloud Development Kit [AWS CDK], AWS SAM, AWS CodeArtifact, AWS Copilot, Amplify, Lambda)
- Verpackungsoptionen für Lambda-Bereitstellungen

- API-Gateway-Phasen und benutzerdefinierte Domains
- Bereitstellungsstrategien (z. B. Canary, Blue/Green, Rolling)

Fähigkeiten im:

- Aktualisieren vorhandener IaC-Vorlagen (z. B. AWS-SAM-Vorlagen, CloudFormation-Vorlagen)
- Verwalten von Anwendungsumgebungen mithilfe von AWS-Services
- Bereitstellen einer Anwendungsversion mithilfe von Bereitstellungsstrategien
- Übertragen eines Code per Commit an ein Repository, um Build-, Test- und Bereitstellungsaktionen aufzurufen
- Verwenden von orchestrierten Workflows zur Bereitstellung von Code in verschiedenen Umgebungen
- Durchführen von Anwendungs-Rollbacks mithilfe vorhandener Bereitstellungsstrategien
- Verwenden von Kennzeichnungen und Branches für die Versionsverwaltung
- Verwenden vorhandener Laufzeitkonfigurationen zum Erstellen dynamischer Bereitstellungen (z. B. mithilfe von Staging-Variablen aus API-Gateway in Lambda-Funktionen)

Domain 4: Problembehandlung und Optimierung

Aufgabenstellung 1: Hilf beim Durchführen einer Ursachenanalyse.

Wissen über:

- Systeme zum Protokollieren und Überwachen
- Sprachen für Protokollabfragen (z. B. Amazon CloudWatch Logs Insights)
- Datenvisualisierungen
- Tools zur Codeanalyse
- Gängige HTTP-Fehlercodes
- Von SDKs generierte gängige Ausnahmen
- Service-Übersichten in AWS X-Ray

Fähigkeiten im:

- Beheben von Fehlern im Code zur Identifizierung von Defekten
- Interpretieren von Anwendungsmetriken, Protokollen und Traces
- Abfragen von Protokollen, um relevante Daten zu finden

- Implementieren benutzerdefinierter Metriken (z. B. eingebettetes CloudWatch-Metrikformat [Embedded Metric Format, EMF])
- Überprüfen des Anwendungszustands mithilfe von Dashboards und Erkenntnissen
- Beheben von Bereitstellungsfehlern mithilfe von Dienstausgabeprotokollen

Aufgabenstellung 2: Konfiguriere einen Code für Beobachtbarkeit.

Wissen über:

- Verteilte Nachverfolgung
- Unterschiede zwischen Protokollierung, Monitoring und Beobachtbarkeit
- Strukturiertes Protokollieren
- Anwendungsmetriken (z. B. benutzerdefiniert, eingebettet, integriert)

Fähigkeiten im:

- Implementieren einer effektiven Protokollierungsstrategie zur Aufzeichnung des Verhaltens und des Zustands von Anwendungen
- Implementieren von Code, der benutzerdefinierte Metriken ausgibt
- Hinzufügen von Anmerkungen für Nachverfolgungsdienste
- Implementieren von Warnmeldungen für bestimmte Aktionen (z. B. Benachrichtigungen über Kontingentbeschränkungen oder Bereitstellungsabschlüsse)
- Implementieren einer Nachverfolgung mithilfe von AWS-Services und -Tools

Aufgabenstellung 3: Optimierte Anwendungen mithilfe von AWS-Services und -Funktionen.

Wissen über:

- Caching
- Nebenläufigkeit
- Messaging-Dienste (z. B. Amazon Simple Queue Service [Amazon SQS] und Amazon Simple Notification Service [Amazon SNS])

Fähigkeiten im:

- Profilieren der Anwendungsleistung
- Ermitteln des Mindestspeichers und der Rechenleistung für eine Anwendung

- Verwenden von Abonnementfilterrichtlinien zur Messaging-Optimierung
- Zwischenspeichern von Inhalten basierend auf Anfrage-Headern

Anhang

Technologien und Konzepte, die in der Prüfung abgefragt werden könnten

Die folgende Liste enthält Technologien und Konzepte, die in der Prüfung abgefragt werden könnten. Diese Liste ist nicht vollständig und kann sich ändern. Die Reihenfolge und Platzierung der Elemente in dieser Liste gibt keinen Hinweis auf die relative Gewichtung oder Bedeutung in der Prüfung:

- Analytik
- Anwendungsintegration
- Computing
- Container
- Kosten- und Kapazitätsmanagement
- Datenbank
- Entwickler-Tools
- Management und Governance
- Netzwerk und Bereitstellung von Inhalten
- Sicherheit, Identität und Compliance
- Speicher

Abgedeckte AWS-Services und -Funktionen

Die folgende Liste enthält AWS-Services und -Funktionen, die für die Prüfung relevant sind. Diese Liste ist nicht vollständig und kann sich ändern. AWS-Angebote werden in Kategorien angezeigt, die den Hauptfunktionen der Angebote entsprechen:

Analytik:

- Amazon Athena
- Amazon Kinesis
- Amazon OpenSearch Service

Anwendungsintegration:

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

- AWS Step Functions

Computing:

- 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)

Datenbank:

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

Entwickler-Tools:

- 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 und Governance:

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

Netzwerk und Bereitstellung von Inhalten:

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

Sicherheit, Identität und Compliance:

- 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

Speicher:

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

AWS-Services und -Funktionen, die nicht in der Prüfung behandelt werden

Die folgende Liste enthält AWS-Services und -Funktionen, die für die Prüfung nicht relevant sind. Diese Liste ist nicht vollständig und kann sich ändern. AWS-Angebote, die in keinem Zusammenhang mit der durch die Prüfung angestrebten beruflichen Position stehen, sind von der Liste ausgeschlossen:

Analytik:

- Amazon QuickSight

Geschäftsanwendungen:

- Amazon Chime
- Amazon Connect
- Amazon WorkMail

Computing für Endbenutzer:

- Amazon AppStream 2.0
- Amazon WorkSpaces

Front-End-Web und Mobilgeräte:

- AWS Device Farm

Game Tech:

- Amazon GameLift

Maschinelles Lernen:

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

Management und Governance:

- AWS Managed Services (AMS)
- AWS Service Catalog

Medienservices:

- Amazon Elastic Transcoder

Migration und Übertragung:

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

Sicherheit, Identität und Compliance:

- AWS Shield Advanced
- AWS Shield Standard

Speicher:

- AWS Snow Family
- AWS Storage Gateway

Umfrage

Wie nützlich war dieser Prüfungsleitfaden? Lass es uns wissen, indem du [an unserer Umfrage teilnimmst](#).