Google Cloud

Professional Data Engineer

認定試験ガイド

Professional Data Engineer は、データを収集、変換、公開することで、ほかのユーザーにとってデータを有効で価値のあるものにします。ビジネス要件と規制要件を満たすために、プロダクトやサービスを評価し、選択します。Professional Data Engineer は、堅牢なデータ処理システムを作成して管理します。これには、データ処理ワークロードを設計、構築、デプロイ、モニタリング、維持、保護する能力が含まれます。

セクション 1: データ処理システムの設計 (試験の約22%)

1.1 セキュリティとコンプライアンスを考慮した設計。以下のような点を考慮します。

- Identity and Access Management（Cloud IAM と組織のポリシーなど）
- データ セキュリティ（暗号化と鍵管理）
- プライバシー（個人を特定できる情報、Cloud Data Loss Prevention API など）
- データアクセスと保存に関する地域的な考慮事項（データ主権）
- 法令遵守、規制遵守

1.2 信頼性と確実性を考慮した設計。以下のような点を考慮します。

- データの準備とクリーニング（Dataprep、Dataflow、Cloud Data Fusion など）
- データ パイプラインのモニタリングとオーケストレーション
- 障害復旧とフォールト トレランス
- Atomicity（原子性）、Consistency（一貫性）、Isolation（独立性）、Durability（永続性）（ACID）に対するコンプライアンスと可用性に関連する意思決定
- データの検証

1.3 柔軟性とポータビリティを考慮した設計。以下のような点を考慮します。

- アーキテクチャへの現在と将来のビジネス要件のマッピング
- データとアプリケーションのポータビリティを考慮した設計（例: マルチクラウド、データ所在地の要件）
- データのステージング、カタログ化、検出（データ ガバナンス）

1.4 データ移行の設計。以下のような点を考慮します。

- 現在の関係者のニーズ、ユーザー、プロセス、技術の分析と望ましい状態を実現するための計画の策定
- Google Cloud への移行計画（BigQuery Data Transfer Service、Database Migration Service、Transfer Appliance、Google Cloud ネットワーキング、Datastream など）
- 移行検証戦略の策定
- 適切なデータ ガバナンスを確実化するためのプロジェクト、データセット、テーブル アーキテクチャの設計

セクション **2:** データの取り込みと処理 **(**試験の約**25%)**

2.1 データ パイプラインの計画。以下のような点を考慮します。

- データソースとシンクの定義
- データ変換ロジックの定義
- ネットワーキングの基礎
- データ暗号化

2.2 パイプラインの構築。以下のような点を考慮します。

- データ クレンジング
- サービスの特定（例: Dataflow、Apache Beam、Dataproc、Cloud Data Fusion、BigQuery、Pub/Sub、Apache Spark、Hadoop エコシステム、Apache Kafka など）
- 変換
    - バッチ
    - ストリーミング（例: ウィンドウ処理、受信遅延データなど）
    - 言語
    - アドホックなデータの取り込み（1 回限りまたは自動化されたパイプライン）
- データの取得とインポート
- 新しいデータソースとの統合

2.3 パイプラインのデプロイと運用化。以下のような点を考慮します。

- ジョブの自動化とオーケストレーション（例: Cloud Composer と Workflows など）
- CI / CD（継続的インテグレーションおよび継続的デプロイ）

セクション 3: データの保存 (試験の約20%)

3.1 ストレージ システムの選択。以下のような点を考慮します。

- データアクセス パターンの分析
- マネージド サービスの選択（例: Bigtable、Cloud Spanner、Cloud SQL、Cloud Storage、Firestore、Memorystore）
- ストレージの費用とパフォーマンスの計画
- データのライフサイクル管理

3.2 データ ウェアハウスを使用するための計画。以下のような点を考慮します。

- データモデルの設計
- データ正規化の度合いの決定
- ビジネス要件のマッピング
- データアクセス パターンをサポートするアーキテクチャの定義

3.3 データレイクの使用。以下のような点を考慮します。

- レイクの管理（データの検出、アクセス、費用管理の構成）
- データの処理
- データレイクのモニタリング

3.4 データメッシュを考慮した設計。以下のような点を考慮します。

- 要件に基づくデータメッシュを Google Cloud のツール（例: Dataplex、Data Catalog、BigQuery、Cloud Storage）で構築する
- データを分散チームで使用するためにセグメント化する
- 分散データシステム用の連携ガバナンス モデルを構築する

セクション 4: 分析用データの準備と使用 (試験の約15%)

4.1 可視化用データの準備。以下のような点を考慮します。

- ツールへの接続
- フィールドの事前計算
- BigQuery マテリアライズド ビュー（ビューロジック）
- 時間データの粒度の決定
- パフォーマンスの悪いクエリのトラブルシューティング

- Identity and Access Management（IAM）および Cloud Data Loss Prevention（Cloud DLP）

4.2 データの共有。以下のような点を考慮します。

- データ共有のルール定義
- データセットの公開
- レポートと視覚化の公開
- Analytics Hub

4.3 データの探索と分析。以下のような点を考慮します。

- 特徴量エンジニアリングのためのデータ準備（ML モデルのトレーニングと提供）
- データ検出の実施

セクション 5: データ ワークロードの管理と自動化 (試験の約18%)

5.1 リソースの最適化。以下のような点を考慮します。

- データに関連するビジネスニーズに従って費用を最小限に抑える
- ビジネス クリティカルなデータプロセスにとって十分なリソースを使用できるようにする
- 永続的なデータクラスタとジョブベースのデータクラスタ（例: Dataproc）のどちらを使用するかを決定する

5.2 自動化と反復性の設計。以下のような点を考慮します。

- Cloud Composer の有向非巡回グラフ（DAG）の作成
- 反復可能な方法でのジョブのスケジューリング

5.3 ビジネス要件に基づくワークロードの最適化。以下のような点を考慮します。

- Flex、オンデマンド、定額のスロット料金（（柔軟性をとるか、固定容量をとるか）
- インタラクティブ方式またはバッチ方式のクエリジョブ

5.4 プロセスのモニタリングとトラブルシューティング。以下のような点を考慮します。

- データプロセスのオブザーバビリティ（例: Cloud Monitoring、Cloud Logging、BigQuery 管理パネル）
- 計画された使用量のモニタリング
- エラー メッセージ、請求に関する問題、割り当てのトラブルシューティング

- ジョブ、クエリ、コンピューティング容量（予約）などのワークロードの管理

5.5 障害への意識の持続と影響の軽減。以下のような点を考慮します。

- フォールトトレランスを念頭に置いたシステム設計と再起動の管理
- 複数のリージョンまたはゾーンでのジョブの実行
- データの破損や欠落への準備
- データのレプリケーションとフェイルオーバー（例: Cloud SQL、Redis クラスタ）