

MuleSoft.MCIA-Level-1-Maintenance.v2023-07-07.q42

試験コード:	MCIA-Level-1-Maintenance
試験名称:	MuleSoft Certified Integration Architect - Level 1 MAINTENANCE
認定資格:	MuleSoft
無料問題数:	42
バージョン:	v2023-07-07
アクセス数:	452
ページビュー数:	420
https://www.jpnpdf.com/MuleSoft.MCIA-Level-1-Maintenance.v2023-07-07.q42-mondaishu.html	

最新問題: 1

ある企業は、Oracle ERP でホストされている REST API 対応の請求支払いルール エンジンによって駆動される一連の重要な機能をサポートする、新しい Mule アプリケーションを実装しています。設計どおり、Mule アプリケーションはバッチ処理ロジックを実行する際に多くのデータ変換操作を必要とします。

同社は、既存の Java ベースの機能 (クラス、オブジェクト、データ モデルなど) をできるだけ多く活用して再利用したいと考えています。新しい Mule で Mule アプリケーションと Oracle ERP の間で必要なデータ マッピングと変換を実装する際に、どのようなアプローチを考慮する必要がありますか。応用？

- A. Mule で適切な Java オブジェクトから新しいメタデータ RAML クラスを作成し、Dataweave 経由で変換を実行します。
- B. 適切な Java メソッドのいずれかを直接呼び出し、メタデータ RAML クラスを作成し、Dataweave 経由で必要な変換を実行します。
- C. Mule アプリケーションから、XSLT モデル経由で変換します。
- D. Dataweave から適切な Java クラスを呼び出して変換します。

Answer: ([解答を表示する](#))

最新問題: 2

ある企業は、導入環境をオンプレミス クラスターからランタイム ファブリック (RTF) クラスターに移行することを計画しています。また、Mule ランタイム インスタンスにデプロイされた Mule アプリケーションがアプリケーションのレプリカと再起動間でデータを保存および共有できるようにするための要件もあります。

これらの要件はどのように満たされるのでしょうか？

- A. クラスター ノードの 1 つにオブジェクトストア ポッドをインストールします。
- B. RTF で永続的ゲートウェイを構成する
- C. RTF クラスター内のレプリカ間でデータを共有する Anypoint オブジェクトストア V2

D. Mule Object Store を使用していずれかのサーバーで永続化ゲートウェイを設定します。

Answer: B ([メッセージを残す](#))

最新問題: 3

組織は、パフォーマンス プロファイルを備えた 8 音符クラスターにランタイム ファブリックを導入しました。API は、状態データの一部を維持するために非永続オブジェクトストアを使用します。サーバーがクラッシュした場合、古いデータにはどのような影響がありますか？

- A. 状態データは以前に保存されたバージョンにロールバックされます
- B. 状態データは、複数のデータがクラッシュの影響を受けない限り保持されます。
- C. 状態データは保存されます
- D. 状態データが失われています

Answer: B ([メッセージを残す](#))

最新問題: 4

ある組織は、従来のバックエンドに接続する Mule アプリケーションを設計しています。バックエンド サービスの可用性は高くなく、頻繁にダウンタイムが発生することが報告されています。統合アーキテクトとして、高信頼性の目標を達成するには、次のどのアプローチを提案しますか？

- A. Mule ランタイムでアラートを設定できるため、サービスがダウンしたときにバックエンド チームに通信できるようになります。
- B. バックエンド API の呼び出し中に成功スコープが実装されるまで
- C. 再度エラーが発生した場合に呼び出すために使用される On Error Continue スコープ
- D. バックエンド API の可用性に応じて、そのジョブを使用してバックエンドに送信されるすべてのリクエストを含むバッチ ジョブを作成します。

Answer: ([解答を表示する](#)**)**

説明

正解は、バックエンド API の呼び出し中に、Until Successful スコープを実装できることです。until Successful スコープは、すべてが成功するか、最大再試行回数を超えるまで、スコープのコンポーネント (フロー参照を含む) を繰り返しトリガーします。このスコープには、最大再試行回数を制御するオプションが用意されています。再試行と再試行間の間隔 このスコープは、何らかの理由で失敗する可能性があり、再試行で成功する可能性があるプロセッサの任意のシーケンスを実行できます。

最新問題: 5

組織は、コード、ビルド、テストの各段階を通じて Mule アプリケーションを促進する継続的インテグレーション (CI) ライフサイクルを実装しました。組織の CI の取り組みを標準化するために、依存関係、バージョン管理、ビルド プロモーションなどの情報を含む成果物を保存するための新しい依存関係制御アプローチが設計されています。

これらのプロセス改善を実装するために、組織は開発者に対し、Mule アプリケーション コードに関連するすべての依存関係を共有の場所に維持することを要求します。

Mule アプリケーション コードに関連するすべての依存関係を標準化するために組織が共有場所で使用する必要がある、最も慣用的な (本来の目的で使用される) タイプのシステムは何ですか？

- A. cloudhub.io の Anypoint Object Store サービス
- B. MuleSoft が管理するリポジトリ atrepository.mulesoft.org
- C. API コミュニティマネージャー
- D. バイナリ アーティファクト リポジトリ

Answer: C ([メッセージを残す](#))

最新問題: 6

Mule アプリケーションは APIkit for SOAP を使用して SOAP Web サービスを実装します。Mule アプリケーションは、テスト環境の CloudHub ワーカーにデプロイされています。

統合テスト チームは、SOAP クライアントを使用して統合テストを実行したいと考えています。統合テストを実行するには、統合チームは SOAP Web サービスのインターフェイス定義を取得する必要があります。

SOAP クライアントとの統合テストを実行するために、統合テスト チームがデプロイされた SOAP Web サービスのインターフェイス定義を取得する最も慣用的な (本来の目的で使用される) 方法は何ですか？

- A. デプロイされた Mule アプリケーションから WSDL ファイルを取得します。
- B. Runtime Manager から XML ファイルを取得します。
- C. デプロイされた Mule アプリケーションから RAML ファイルを取得します。
- D. API Manager から OpenAPI 仕様ファイルを取得します。

Answer: ([解答を表示する](#)**)**

最新問題: 7

AnyAirline 組織の旅客予約センターは、ビジネス トランザクションで 3 つの異なるシステム API (bookFlight、bookHotel、bookCar) の呼び出しを組み合わせる統合ソリューションを設計しています。

各システム API は単一のデータベースを呼び出します。

少なくとも 1 つの API が失敗した場合は、ビジネス トランザクション全体をロールバックする必要があります。

これらの API をほぼリアルタイムで統合し、一貫性、パフォーマンス、信頼性の最適なバランスを実現する最も慣用的な (本来の目的に使用される) 方法は何ですか？

- A. 各API実装でローカルトランザクションを実装

Saga パターンを使用して API 実装間を調整する

障害が発生した場所に応じてさまざまな補償アクションを適用します

- B. 各 API 実装内でローカル トランザクションを実装します。

同じ eXtended Architecture (XA) トランザクションにも参加するように各 API 実装を構成する 各 API 実装にキャッシュを実装してパフォーマンスを向上させる

- C. API 実装間に eXtended Architecture (XA) トランザクションを実装する Saga パターンを使用して API 実装間を調整する 各 API 実装にキャッシュを実装してパフォーマンスを向上させる

D. Saga パターンを使用して Mule アプリケーションに eXtended Architecture (XA) トランザクション マネージャを実装する XA トランザクションを使用して各 API 実装を Mule アプリケーションに接続する 障害が発生した場所に応じて、さまざまな補償アクションを適用する

Answer: A ([メッセージを残す](#))

最新問題: 8

保険会社には、現在顧客が使用している既存の API があります。API は顧客がホストする Mule ランタイム クラスタにデプロイされます。Mule クラスタ上の API にアクセスするために使用されるロードバランサは、サーバ上のポート 443 でホストされているアプリケーションを指すようにのみ設定されています。

企業の Mule アプリケーション チームは、ポート 443 を使用して 2 番目の API をデプロイしようとしたが、アプリケーションは起動せず、ログを確認すると、アドレスがすでに使用されていることを示すエラーが表示されました。

このエラーを解決し、顧客が両方の API にアクセスできるようにするには、組織はどのような手順を実行する必要がありますか？

- A. 2 番目の API の HTTP リスナーが最初の API で使用されたポートとは異なるポートを使用するように設定します。
- B. 2 番目の API の HTTP リスナー設定のベース パスを最初の API とは異なるパスに変更します。
- C. 両方の API で HTTP リスナー構成を設定し、複数のポートからの接続を許可します。
- D. HTTP リスナー設定を API から移動し、ポート 443 を使用して Mule ドメイン プロジェクトにパッケージ化します。

Answer: ([解答を表示する](#))

最新問題: 9

Clondhub で公開されている SOAP V1.2 API には、ルール制限ポリシーが適用されています。API 実装は、HTTP のメイン フローでエラーが伝播すると、グローバル エラー ハンドラーでエラーを捕捉します。HTTP ステータスが 429 に設定された RETRY_EXHAUSTED と、HTTP ステータスが 500 に設定された any です。

クライアントが API 呼び出しの割り当てを超過した場合に予想される HTTP ステータスは何ですか？

- A. ポリシー違反のため HTTP ステータス 401 が許可されていません
- B. 呼び出しがバックエンドに到達しないため、レート制限ポリシー違反による HTTP ステータス 400
- C. API の HTTP:RETRY EXHAUSTED エラー ハンドラで定義された HTTP ステータス 429 B. API:RETRY_EXHAUSTED が生成されるため、API の ANY エラー ハンドラで定義された HTTP ステータス 500

Answer: C ([メッセージを残す](#))

最新問題: 10

商社は 1 日に何百万ものリクエストを処理します。ビジネスの性質上、アプリケーション内で優れたパフォーマンスと信頼性が必要です。

この目的のために、同社は、ネットワーク内にある共有メッセージ キューを介して通信するさまざまな Mule クラスタでホストされる多数のイベントベースの API を使用しています。

企業のシステム要件を満たすにはどのような方法を使用する必要がありますか？

- A. JMS トランザクション
- B. XA トランザクションと XA 接続コンポーネント
- C. 信頼性パターンを持つ VM キュー
- D. 信頼性パターンを使用した JMS 手動確認応答

Answer: D ([メッセージを残す](#))

最新問題: 11

重要な支払い関連の Mule アプリケーションの 1 つでは、トランザクションが使用されています。実装の拡張機能として、トランザクション グループの一部でもあるスキャッター ギャザー ルートが導入されました。スキャッターギャザールートは4つのルートがあります。

スキャッター/ギャザー ルーターの 4 番目のルートでエラーが発生し、トランザクションをロールバックする必要がある場合、Mule アプリケーションはどのように動作しますか？

- A. エラーが発生したルートのみがロールバックされます。
- B. すべてのルートがロールバックされます
- C. Scatter Gather ルーターをトランザクションに含めることはできません

Answer: B ([メッセージを残す](#))

説明

*Scatter Gather: トランザクション内で実行する場合、Scatter Gather は並列実行されません。これは、最初のルートが処理された後に 2 番目のルートが実行され、2 番目のルートが処理された後に 3 番目のルートが実行されるということを意味します。エラーが発生した場合、すべてのルートがロールバックされます。

最新問題: 12

組織は API 主導の接続をうまく活用していますが、アプリケーション ネットワークが拡大するにつれて、API の公開、共有、検出、登録、ポリシーの適用、デプロイなどの手動で実行されるすべてのタスクが反復的な作業となり、組織はこのプロセスを自動化する必要に迫られています。効率的な CI/CD パイプライン。

Anypoint プラットフォームの機能を考慮すると、組織は API ライフサイクルの自動化にどのようにアプローチすべきでしょうか？

- A. Groovy などのスクリプト言語で Anypoint CLI または Anypoint Platform REST API を使用する
- B. API ライフサイクルに必要なカスタム構成で Maven を使用する
- C. API 管理と MavenforAPI デプロイメントに Exchange REST API を使用する
- D. API 管理と API デプロイメントにランタイム マネージャー REST API を使用する

Answer: C ([メッセージを残す](#))

最新問題: 13

旅行会社は、明確に定義された予約サービス API を公開して、ビジネス パートナーと共有したいと考えています。

これらのビジネス パートナーは、SOAP サービスのみを利用することに同意しており、開発を開始する前に簡単に利用できる方法でサービス契約を取得したいと考えています。旅行会社は初期設計ドキュメントを Anypoint Exchange に公開し、それらのドキュメントをビジネス パートナーと共有します。API 主導のアプローチを使用する場合、旅行会社がビジネス パートナーに提供する必要があります最初の設計ドキュメントは何ですか？

- A. 任意の XML エディタを使用して WSDL 仕様を作成します。
- B. 任意のテキスト エディタを使用して RAML API 仕様を作成します。
- C. Design Center で OAS API 仕様を作成する
- D. Design Center で SOAP API 仕様を作成する

Answer: A ([メッセージを残す](#))

説明

SOAP API 仕様は WSDL として提供されます。デザイン センターには、WSDL ファイルを作成する機能はありません。したがって、WSDLはXMLエディタを使用して作成する必要があります。

最新問題: 14

お客様は、お客様がホストするさまざまな Mule ランタイムに Mule アプリケーションをデプロイしました。Mule アプリケーションは Anypoint プラットフォームから管理されます。

Anypoint モニタリングからこれらの Mule アプリケーションをモニタリングするには何を設定する必要がありますか？また、Anypoint モニタリングにモニタリング データを送信するには何を設定する必要がありますか？

- A. 各 Mule ランタイムにランタイム マネージャー エージェントをインストールします。
ランタイム マネージャー エージェントは、Mule アプリケーションからのデータを監視してから Anypoint 監視に移行します。
- B. デフォルトでは、Anypoint モニタリング エージェントは各 Mule ランタイムにインストールされます。Anypoint モニタリング エージェントは、Mule アプリケーションから Anypoint モニタリングにモニタリング データを自動的に送信します。
- C. ランタイム マネージャー アプリケーション設定から個々のアプリケーションの監視を有効にするランタイム マネージャー エージェントは、Mule アプリケーションからの監視データを Anypoint 監視に送信します
- D. 各 Mule ランタイム上の Anypoint モニタリング エージェント
Anypoint モニタリング エージェントは、Mule アプリケーションから Anypoint モニタリングにモニタリング データを送信します。

Answer: D ([メッセージを残す](#))

最新問題: 15

頻繁なプラグイン バージョンのアップグレードと外部プラグイン プロジェクトの依存関係に悩まされている組織。チームは、すべての新規プロジェクトと進行中のプロジェクトにわたってデフォルトの依存関係のセットを定義するベスト プラクティスを作成することで、アプリケーションへの影響を最小限に抑えたいと考えています。

最小限の責任を持つアプリケーションでこれらのベスト プラクティスをどのように達成できるでしょうか？

- A. POM.xml ファイルで定義されているすべての依存関係を含む Mule ドメイン プロジェクトを作成し、各アプリケーションをドメイン プロジェクトに追加します。
- B. 各アプリケーションの POM.xml ファイルにすべての依存関係を追加します。
- C. 必要なすべての依存関係の親 POM を作成し、各アプリケーションの POM.xml ファイルでそれぞれを参照します。
- D. すべての依存関係を含む Mule プラグイン プロジェクトを作成し、それを各アプリケーションの POM.xml ファイルに依存関係として追加します。

Answer: ([解答を表示する](#))

最新問題: 16

ある企業は、ユーザーが自社の内部ユーザー認証情報を使用して Anypoint Platform にログインすることを望んでいます。

これを実現するには、企業は外部アイデンティティ プロバイダー (IdP) を企業の Anypoint Platform マスター組織と統合する必要がありますが、SAML 2.0 は使用できません。SAML 2.0 以外に、企業は IdP を Anypoint Platform マスター組織と統合するためにどのようなシングル サインオン標準を使用できますか？

- A. SAML 1.0
- B. OAuth 2.0
- C. ベーシック認証
- D. OpenID Connect

Answer: D ([メッセージを残す](#))

説明

Anypoint Platform 組織管理者は、Anypoint Platform でアイデンティティ管理を設定して、シングル サインオン (SSO) 用にユーザーを設定できます。

次のシングル サインオン標準のいずれかを使用して ID 管理を構成します。

- 1) OpenID Connect: SSO を含む認可サーバーによるエンドユーザー ID 検証
- 2) SAML 2.0: クロスドメイン SSO を含む Web ベースの認証

有効な **MCIA-Level-1-Maintenance** 問題集は GoShiken.com が提供された合格しやすい MCIA-Level-1-Maintenance 試験問題集！ GoShiken.com が最新の **MCIA-Level-1-Maintenance** 試験問題集を提供しています。GoShiken.com MCIA-Level-1-Maintenance 試験問題は最新で、解答が正確でございます。最新の GoShiken.com MCIA-Level-1-Maintenance 問

題集をゲットする人はこちら: <https://www.goshiken.com/MuleSoft/MCIA-Level-1-Maintenance-mondaishu.html> (11830%OFF問題集溶と正解付きで 30%w 特別割引コード: **Freepdfdumps**)

最新問題: 17

自動化エンジニアは、Anypoint Platform での API と実装の作成、公開、デプロイ、管理の手順を含む、API ライフサイクルの手順を自動化するスクリプトを作成する必要があります。

Anypoint Platform REST API を直接呼び出す必要がなく、最も簡単な方法でスクリプト内のこれらすべてのアクションの実行を自動化するために使用できる Anypoint Platform の機能はどれですか？

- A. API Manager の自動ポリシー
- B. ランタイムマネージャーエージェント
- C. Mule Maven プラグイン
- D. Anypoint CLI

Answer: D ([メッセージを残す](#))

説明

Anypoint Platform は、Anypoint Platform と Anypoint Platform Private Cloud Edition (Anypoint Platform PCE) の両方にスクリプトおよびコマンドライン ツールを提供します。コマンドライン インターフェイス (CLI) は、対話型シェル モードと標準 CLI モードの両方をサポートし、以下と連携します。Anypoint Exchange アクセス管理 Anypoint Runtime Manager

最新問題: 18

Mule アプリケーションからの同期ロギングと非同期ロギングの主な違いは何ですか？

- A. 同期ロギングは単一のロギング スレッドにログ メッセージを書き込みますが、次のイベント プロセッサによって処理されている Mule イベントをブロックしません。
- B. 非同期ロギングにより、ESB Mule イベントの処理スループットが向上し、同時に各 Mule イベントの処理時間が短縮されます。
- C. 非同期ログにより、より正確なタイムスタンプを備えた、より信頼性の高い監査証跡が生成されます。
- D. 進行中のトランザクション内の同期ログは、現在の Mule イベントを処理するのと同じスレッドにログ メッセージを書き込みます。

Answer: B ([メッセージを残す](#))

説明

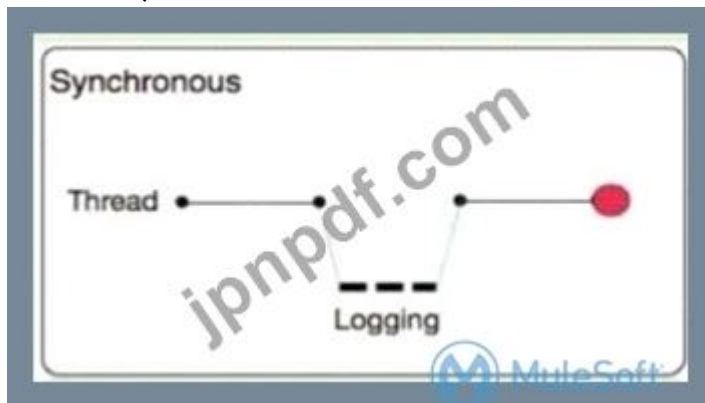
ロギングの種類:

A) 同期: メッセージを処理しているスレッドの実行は中断され、ログ メッセージが完全に処理されるまで待機してから続行されます。

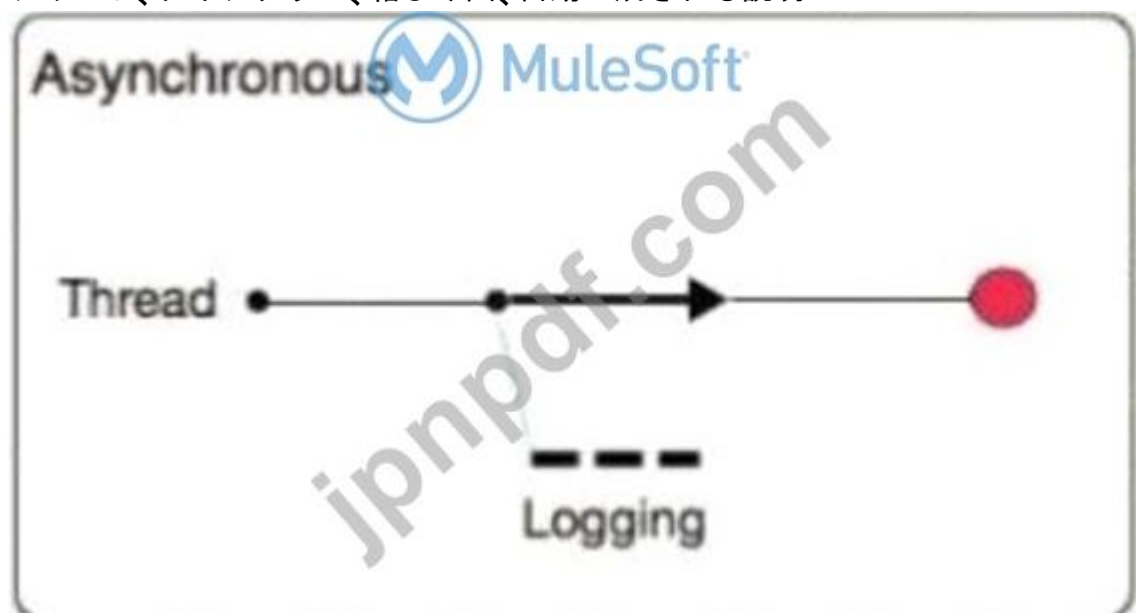
メッセージを処理しているスレッドの実行は、続行する前にログ メッセージが完全に出力されるまで待機するために中断されます。同期ログが原因でパフォーマンスが低下します。ログが監査証跡として使用される場合、またはエラー/クリティカル メッセージをログに記録する場合に使用されます。ロガーがディスクへの書き込みに失敗すると、現在 Mule イベントを処理しているのと

同じスレッドで例外が発生します。ログ記録が重要な場合は、トランザクションをロールバックできます。

チャート、図表の説明が自動生成されます



チャート、ダイアグラム、箱ひげ図、自動生成される説明



B) 非同期:

ロギング操作は別のスレッドで行われるため、メッセージの実際の処理がロギングの完了を待つために遅れることはありません。メッセージ処理のスループットとレイテンシが大幅に改善されます。Mule ランタイム エンジン (Mule) 4 は、Log4j 2 非同期ロギングを使用します。デフォルト非同期ログの欠点は、エラー処理です。

ロガーがディスクへの書き込みに失敗した場合、処理を実行しているスレッドはディスクへの書き込みの問題を認識しないため、何もロールバックできなくなります。ログの実際の書き込みは異なるため、バッファがフラッシュされる前に Mule がクラッシュした場合、ログ メッセージがディスクに到達せずに失われる可能性があります。

つまり、正解は次のとおりです。非同期ロギングにより、ESB Mule イベントの処理スループットが向上し、同時に各 Mule イベントの処理時間が短縮されます。

最新問題: 19

Mule アプリケーションは、単一データベース上での一連の操作のローカル トランザクションをサポートするように構築されています。Mule アプリケーションには、ローカル トランザクションに参加する Scatter-Gather があります。

このローカル トランザクション内で実行されている場合、Scatter-Gather はどのような動作をしますか？

- A. スキャッタギャザー内のすべてのルートの実行が並行して行われます。スキャッタギャザー内でエラーが発生すると、すべてのデータベース操作がロールバックされます。
- B. スキャッタギャザー内の各ルートの実行は順番に行われます。スキャッタギャザー内で発生したエラーはデータベース操作のロールバックにはなりません。
- C. スキャッタギャザー内の各ルートの実行は順番に行われます。スキャッタギャザー内でエラーが発生すると、すべてのデータベース操作がロールバックされます。
- D. スキャッタギャザー内の各ルートの実行は並行して行われます。スキャッタギャザー内で発生したエラーはデータベース操作のロールバックにはなりません。

Answer: C ([メッセージを残す](#))

最新問題: 20

Mule アプリケーションには、複数のバッチ ステップ スコープを持つバッチ ジョブ スコープが含まれています。バッチ ジョブのスコープは、バッチ ブロック サイズ 25 で構成されています。

4,000 レコードを含むペイロードがバッチ ジョブ スコープによって受信されます。

エラーがない場合、バッチ ジョブ スコープはバッチ ステップ スコープ内およびバッチ ステップ スコープ間でレコードをどのように処理しますか？

- A. バッチ ジョブ スコープは、各レコード ブロックを一度に 1 つずつ順番に処理します。各バッチ ステップ スコープは、受信した Mule イベントのペイロード内の 1 つのレコードを使用して呼び出されます。各バッチ ステップ スコープでは、ブロック内の 25 レコードすべてが順番に処理されます。一度に 1 つずつレコードのブロックが次のバッチ ステップ スコープで使用可能になる前に、4000 レコードすべてが完了する必要があります
- B. バッチ ジョブ スコープは複数のレコード ブロックを並行して処理し、25 レコードのブロックは前のレコード ブロックを超えて次のバッチ ステップ スコープにジャンプできます。各バッチ ステップ スコープは、ペイロード内の 1 つのレコードで呼び出されます。Mule イベントを受信しました バッチ ステップ スコープごとに、ブロック内の 25 レコードすべてが一度に 1 レコードずつ順番に処理されます 25 レコードのブロックが次のバッチ ステップ スコープで使用可能になる前に、ブロック内のすべてのレコードが完了する必要があります
- C. バッチ ジョブ スコープは複数のレコード ブロックを並行して処理し、25 レコードのブロックは前のレコード ブロックを超えて次のバッチ ステップ スコープにジャンプできます。各バッチ ステップ スコープは、ペイロード内の 1 つのレコードで呼び出されます。Mule イベントを受信しました。バッチ ステップ スコープごとに、ブロック内の 25 レコードすべてが並列処理されます。25 レコードのブロックが次のバッチ ステップ スコープで使用可能になる前に、ブロック内のすべてのレコードが完了する必要があります。
- D. バッチ ジョブ スコープは複数のレコード ブロックを並行して処理します。

各バッチ ステップ スコープは、受信した Mule イベントのペイロード内の 25 レコードのバッチで呼び出されます。各バッチ ステップ スコープでは、4000 レコードすべてが並列処理されます。個々のレコードは、残りのレコードよりも前に次のバッチ ステップ スコープにジャンプできます。現在のバッチ ステップ スコープでの処理を終了する

Answer: C ([メッセージを残す](#))

最新問題: 21

Anypoint Exchange の API は、受け入れられたセマンティック バージョニングの実践に従って、API プロデューサーによってバージョン 3.1.1 から 3.2.0 に更新され、変更は API のパブリック ポータル経由で通知されました。

API エンドポイントは新しいバージョンでも変更されません。API クライアントの開発者はこの変化にどのように対応すべきでしょうか？

- A. 更新はプロジェクトのリスクとして特定され、この API を使用する機能の完全な回帰テストを実行する必要があります。
- B. 既存の機能への変更を理解するには、API プロデューサーに連絡する必要があります。
- C. API プロデューサーは、古いバージョンと新しいバージョンを並行して実行するように要求される必要があります。
- D. API クライアント コードは、新しい機能を利用する必要がある場合にのみ変更する必要があります。

Answer: D ([メッセージを残す](#))

説明

* セマンティック バージョニングは、XYZ 形式の 3 要素の数値です。ここで、:

X はメジャー バージョンを表します。

Y はマイナー バージョンを表します。

Z はパッチを表します。

したがって、SemVer は Major.Minor.Patch の形式になります。私たちの質問に来ると、API のマイナー バージョンが変更され、下位互換性があります。したがって、API クライアント側で必要な変更はありません。マイナー バージョン変更の一部として追加された新しい機能を利用したい場合は、最終的にコードを変更する必要がある場合があります。したがって、正しい答えは、新しい機能を利用する必要がある場合にのみ、API クライアント コードを変更する必要があります。

自動生成される図の説明



最新問題: 22

Mule アプリケーションは Customer Hosted Runtime にデプロイされます。システムのスループットを向上させるために、非同期ログが実装されました。しかし、トランザクションのロールバックに使用される重要な例外ログ メッセージの一部が期待どおりに機能せず、組織に多大な損失をもたらしていることが長期間にわたって観察されました。組織はこうした損失を避けたいと考えています。アプリケーションには、スループットをあまり犠牲にすることができない制約もあります。この場合、考えられる選択肢は何でしょうか？

- A. ロギングを非同期から同期に変更する必要があります
- B. この場合、外部ログ アペンダを使用する必要があります。
- C. このようなシナリオでは永続メモリ ストレージを使用する必要があります。
- D. 同期方法を介して例外をログに記録するには、非同期ロガーまたは同期ロガーの混合構成を使用する必要があります。

Answer: D ([メッセージを残す](#))

説明

正しいアプローチは、非同期ロガーまたは同期ロガーの混合構成を使用することです。同期方法で例外をログに記録するには、非同期ロガーを使用する必要があります。非同期ロギングには、パフォーマンスと信頼性のトレードオフが伴います。ロギングバッファがディスクにフラッシュされる前に Mule がクラッシュすると、一部のメッセージが失われる可能性があります。この場合、アプリ内で非同期ロガーまたは同期ロガーを混合構成できることを考慮してください。ベストプラクティスは、運用アプリケーションでは、最小ログ レベル WARN で同期ログではなく非同期ログを使用することです。場合によっては、ポリシーのインストールの成功などのイベントを確認する必要がある場合や、トラブルシューティングを実行する必要がある場合に、INFO ログ レベルを有効にします。アプリケーションの `src/main/resources/log4j2.xml` ファイルを編集して、ロギング戦略を構成します。

最新問題: 23

Mule アプリケーションは、単一データベース上での一連の操作のローカル トランザクションをサポートするように構築されています。Mule アプリケーションには、ローカル トランザクションに参加する Scatter-Gather スコープがあります。

このローカル トランザクション内で実行されている場合、Scatter-Gather はどのような動作をしますか？

- A. Scatter-Gather 内のすべてのルートが並行して実行されます。Scatter-Gather 内でエラーが発生すると、すべてのデータベース操作がロールバックされます。
- B. Scatter-Gather 内のすべてのルートの実行は順番に行われます。Scatter-Gather 内でエラーが発生すると、すべてのデータベース操作がロールバックされます。
- C. Scatter-Gather 内のすべてのルートの実行は並行して行われます。Scatter-Gather 内で発生したエラーはエラー ハンドラーによって処理され、ロールバックは発生しません。
- D. Scatter-Gather 内のすべてのルートの実行は順番に行われます。Scatter-Gather 内で発生したエラーはエラー ハンドラーによって処理され、ロールバックは発生しません。

Answer: A ([メッセージを残す](#))

最新問題: 24

2 つの要件を満たすように設計された Mule アプリケーション

a) ファイルの処理は、VM イベントの負荷分散のための VM 中間キューを使用して、FTPS サーバーからバックエンド データベースまで同期的行われます。b) バッチ ジョブ スコープを使用して、ソースからターゲット システムへの中程度の速度のレコードを処理します。FTPS ファイル。アプリケーションが Cloudhub ワーカーにデプロイされている場合、ファイルの処理および バッチ ジョブのスコープに対して VM キューをどのように構成する必要がありますか？

A. FTPS ファイル処理にクラウド ハブの永続キューを使用する

デフォルトで VM キューイングにワーカーのディスクを使用するため、バッチ ジョブ スコープに対して VM キューを構成する必要はありません。

B. FTPS ファイル処理に VM コネクタの永続キューを使用する バッチ ジョブ スコープの VM キューを無効にする

C. FTPS ファイル処理にクラウド ハブの永続 VM キューを使用する

デフォルトでワーカーの JVM メモリを VM キューイングに使用するため、バッチ ジョブ スコープに対して VM キューを構成する必要はありません。

D. FTPS ファイル処理にクラウド ハブの永続 VM キューを使用する

バッチ ジョブ スコープの VM キューを無効にする

Answer: D ([メッセージを残す](#))

最新問題: 25

保険プロバイダーは、アプリケーション インフラストラクチャを管理するために Anypoint プラットフォームを実装しており、満たさなければならない特定の財務要件のため、顧客がホストするランタイムをビジネスに使用しています。多数の同期 API を構築しており、現在これらを 1

台のサーバー上の Mule ランタイムでホストしています。これらのアプリケーションは、オブジェクトストアや VM キューの多用など、多数のコンポーネントを利用しています。

昨年ビジネスは急速に成長し、保険会社はその申請から信頼性の問題に関する報告を受け始めています。

DevOps チームは、現在 API が処理するリクエストが多すぎて、サーバーに過負荷がかかっていることを示しています。チームはまた、サーバーがメンテナンスのためにダウンしている場合には大幅なダウンタイムが発生することにも言及しました。

統合アーキテクトとして、これらの問題を軽減するにはどのオプションを提案しますか？

- A. ロードバランサーを追加し、クラスター構成にサーバーを追加します。
- B. イベント駆動型モデルを使用してアプリケーションを変更する
- C. ロードバランサーを追加し、サーバーグループ構成にサーバーを追加します。
- D. サーバーの CPU メモリとネットワークの物理仕様を増やす

Answer: A ([メッセージを残す](#))

最新問題: 26

プロジェクトチームは、RAML 定義を出発点として使用して API 実装に取り組んでいます。チームは新しい操作を含むように定義を更新し、交換する新しいバージョンを公開しました。一方、別のチームは、同じ API 実装を使用する Mule アプリケーションに取り組んでいます。

開発中に、新しく追加された操作を利用するために Mule アプリケーション チームは何を実行する必要がありますか？

- A. クライアント アプリケーションの交換から REST コネクタを更新します
- B. 新しい定義を持つ Scaffold API 実装アプリケーション
- C. API 実装の API コネクタを更新し、Exchange に公開します。
- D. 新しい定義を使用してクライアント アプリケーションをスキャフォールディングします。

Answer: A ([メッセージを残す](#))

最新問題: 27

組織には、Orders API という HTTPS 対応の Mule アプリケーションがあり、Process Orders という別の Mule アプリケーションからリクエストを受け取ります。

これら 2 つの Mule アプリケーション間の通信は、TLS 相互認証 (双方向 TLS) によって保護される必要があります。

各 Mule アプリケーションのキーを適切に保護しながら、2 つの Mule アプリケーション間の双方向 TLS を適切にサポートするには、これら 2 つの Mule アプリケーションの各トラストストアとキーストアに少なくとも何を保存する必要がありますか？

- A. Orders API トラストストア: Process Orders 公開キー
Orders API キーストア: Orders API の秘密キーと公開キー
注文処理トラストストア: Orders API 公開キー
注文処理キーストア: 注文処理の秘密キーと公開キー
- B. Orders API トラストストア: Orders API の秘密キーと公開キー
注文処理キーストア: 注文処理秘密キー公開キー

C. Orders API トラストストア: Process Orders 公開キー

Orders API キーストア: Orders API 秘密キー

注文処理トラストストア: Orders API 公開キー

注文処理キーストア: 注文処理秘密キー

D. Orders API トラストストア: Orders API 公開キー

注文処理キーストア: 注文処理の秘密キーと公開キー

Answer: A ([メッセージを残す](#))

最新問題: 28

プロジェクトチームは、RAML 仕様を使用して API の機能要件を文書化し、API 定義を提供します。現在の法的要件に従って、定義されたサービス レベル アグリーメントに従って、高レートのリクエストからサービスを保護するために、追加の非機能要件を追加して API 定義を強化するように設計されています。

プロジェクトが Mulesoft API のガバナンスとポリシーに従っていると仮定すると、プロジェクトチームは必要な非機能要件を関係者にどのように伝えるべきでしょうか？

A. 非機能要件に対して API マネージャーでさまざまな SLA を作成し、Exchange に公開します。

B. 機能以外の要件をすべてコメントとして RAML 仕様に追加し、交換用に公開します。

C. 適切なポリシーのフラグメントを使用して API 定義を更新し、Exchange に公開します。

D. 非機能要件に対して API マネージャーでプロキシを作成し、Exchange に公開します。

Answer: C ([メッセージを残す](#))

最新問題: 29

ある組織は、販売注文レコードを含む新しいファイルの SFTP ロケーションを定期的にポーリングし、それらの販売注文を処理する Mule アプリケーションを設計しています。各販売注文は 1 回だけ処理する必要があります。

この要件をサポートするには、Mule アプリケーションは、各販売注文レコードに含まれる一意の ID に基づいて重複する販売注文を識別およびフィルタし、新しい販売注文のみを下流システムに送信する必要があります。

各販売注文レコードに含まれる一意の ID フィールドに基づいて重複する販売注文をフィルタリングするために Mule アプリケーションで設定できる、最も慣用的な (本来の目的で使用される) Anypoint コネクタ、バリデータ、またはスコープはどれですか？

A. 注文 ID で各レコードをフィルターして保存するようにデータベース コネクタを構成します。

B. 新規または更新されたファイル イベント ソースでウォーターマークを構成し、注文 ID で一意のレコードをフィルターします。

C. 受信したファイルの各レコードを注文 ID でフィルターして保存するようにキャッシュ スコープを構成します。

D. 各レコードを注文 ID でフィルターするように冪等メッセージ検証コンポーネントを構成します。

Answer: ([解答を表示する](#))

最新問題: 30

機械設備の予知保全を実装するために、ACME Tractors は、各機械資産のデータを JMS メッセージのシーケンスとして、JMS サーバー上の SENSOR_DATA という名前の JMS キューにほぼリアルタイムで送信する数千の IoT センサーを設置しました。Mule アプリケーションには、JMS サーバーの SENSOR_DATA JMS キューから受信メッセージを受信するように設定された JMS リスナー オペレーションが含まれています。Mule アプリケーションは受信した各 JMS メッセージを保持し、対応する Mule イベントの変換されたバージョンを機械装置のバックエンドシステムに送信します。

Mule アプリケーションは、顧客がホストするマルチノードの Mule ランタイム クラスタにデプロイされます。通常の場合では、各 JMS メッセージは 1 回だけ処理される必要があります。

JMS キューのパフォーマンスと同時メッセージ処理を最大化するには、JMS リスナーをどのように構成する必要がありますか？

A. numberOfConsumers を 1 より大きい値に設定します。

PrimaryNodeOnly = false に設定します

B. numberOfConsumers を 1 より大きい値に設定します。

PrimaryNodeOnly = true に設定します

C. 消費者の数 = 1 を設定します。

PrimaryNodeOnly = true に設定します

D. 消費者の数 = 1 を設定します。

PrimaryNodeOnly = false に設定します

Answer: ([解答を表示する](#))

最新問題: 31

プロセス API に適用される可能性が最も低い API ポリシーはどれですか？

A. カスタムサーキットブレーカー

B. クライアント ID の強制

C. レート制限

D. JSON 脅威からの保護

Answer: D ([メッセージを残す](#))

説明

この質問の鍵は、Process API がクライアントから直接アクセスされることを意図していないという事実にあります。オプションを 1 つずつ分析してみましょう。クライアント ID の強制：これは通常、API クライアントの ID が常に認識され、API ベースの分析に利用できるようにするために、プロセス API レベルで適用されます。レート制限：このポリシーは、発生する可能性のあるサービスの低下から API を保護するために、プロセス レベルの API に適用されます。受信した負荷が処理できる量を超えている場合に備えて カスタム サーキット ブレーカー：これは、API クライアントが失敗した API を呼び出す無駄な時間と労力を節約できるため、プロセス レベル API でも非常に便利な機能です。JSON 脅威からの保護：

このポリシーは Process API では必須ではなく、Experience API として実装されます。このポリシーは、JSON オブジェクトに悪意のあるコードを挿入することにより、悪意のある攻撃からアプ

リケーションを保護するために使用されます。理想的には、プロセス API は外部から呼び出されないため、このポリシーはプロセス API で使用されることはありません。したがって、正解は JSON 脅威保護 MuleSoft ドキュメント リファレンス です。

<https://docs.mulesoft.com/api-manager/2.x/policy-mule3-json-threat>

有効な **MCIA-Level-1-Maintenance** 問題集は GoShiken.com が提供された合格しやすい MCIA-Level-1-Maintenance 試験問題集！ GoShiken.com が最新の **MCIA-Level-1-Maintenance** 試験問題集を提供しています。GoShiken.com MCIA-Level-1-Maintenance 試験問題は最新で、解答が正確でございます。最新の GoShiken.com MCIA-Level-1-Maintenance 問題集をゲットする人はこちら: <https://www.goshiken.com/MuleSoft/MCIA-Level-1-Maintenance-mondaishu.html> (11830%OFF問題集溶と正解付きで 30%w 特別割引コード: **Freepdfdumps**)

最新問題: 32

Anypoint Platform では、企業は複数の基幹業務 (LOB) に対して複数の ID プロバイダー (IdP) を構成したいと考えています。これらの LOB には、複数のビジネス グループ、チーム、および環境が定義されています。

会社のビジネス グループ、チーム、環境全体で複数の IdP を使用できる Anypoint Platform 機能はどれですか？

- A. MuleSoft がホストする (CloudHub) 専用ロード バランサ
- B. クライアント(アプリケーション)管理
- C. 仮想プライベートクラウド
- D. 権限

Answer: ([解答を表示する](#))

説明

環境内で専用のロード バランサーを使用するには、まず Anypoint VPC を作成する必要があります。複数の環境を同じ Anypoint VPC に関連付けることができるため、異なる環境に同じ専用のロード バランサーを使用できます。

最新問題: 33

大手電子商取引大手は、ランタイム ファブリック (RTF) 上の Mulesoft API を使用して顧客の注文を処理する予定です。

クレジットカード情報などの一部の顧客の機密情報も API ペイロードの一部として含まれています。

機密データと元のデータを一致させるリスクを最小限に抑え、必要なときにいつでもどこでも元の値に戻ることができるアプローチは何ですか？

- A. マスキングとトークン化の両方を使用
- B. マスキングを適用して機密情報を非表示にしてから API を使用します

- C. API ゲートウェイでフィールド レベルの暗号化ポリシーを適用します。
- D. マスキング形式を非トークン化し、元の値を返すマネージャー
- E. トークン化フォーマットを作成し、API ゲートウェイにトークン化ポリシーを適用します。

Answer: ([解答を表示する](#))

最新問題: 34

Mule アプリケーションが設計されています。SFTP 経由で外部ベンダーから数百万のレコードを含む CSV ファイルを毎晩受信するには、ファイルのレコードを検証し、変換する必要があります。そしてデータベースに書き込まれます。レコードは任意の順序でデータベースに挿入できます。

この使用例では、Mule コンポーネントのどの組み合わせが、これらのレコードをデータベースに書き込む最も効果的でパフォーマンスの高い方法を提供しますか？

- A. Parallel for Each スコープを使用してレコードを 1 つずつデータベースに挿入します
- B. Scatter-Gather を使用してデータベースにレコードを一括挿入します。
- C. バッチ ジョブ スコープを使用して、データベースにレコードを一括挿入します。
- D. DataWeave マップ操作と非同期スコープを使用して、レコードを 1 つずつデータベースに挿入します。

Answer: C ([メッセージを残す](#))

説明

正解は、「バッチ ジョブ スコープを使用してデータベースにレコードを一括挿入する」です。

* バッチ ジョブは、数百万のレコードを管理する最も効率的な方法です。

ここで注意すべき点は次のとおりです。

信頼性: レコードの処理中に信頼性が必要な場合、つまり、実行時のクラッシュやその他の不幸なシナリオでも処理が存続し、再起動時に残りのすべてのレコードが処理される必要がある場合は、永続キューを使用するバッチを使用します。

エラー処理: 特定のルートでエラーが発生するたびに、そのルートの残りのレコードの処理が並行して停止されます。そのような場合は、on error continue を使用して処理する必要があります。バッチ処理は、そのようなエラー中に停止しません。代わりに、次のようにすることができます。障害に対するステップとして、専用の処理を用意します。

メモリ フットプリント: 質問では、処理するレコードが数百万あるとのことなので、Parallel for each を実行すると、最後に処理されたすべてのレコードが集約され、メモリ不足が発生する可能性があります。

代わりに、バッチ ジョブは完了フェーズで BatchResult を提供し、失敗と成功の数を取得できます。巨大なファイルを処理する場合、順序を気にしない場合は、バッチ ジョブを使用してください。

最新問題: 35

ある組織は、すべての Mule アプリケーションをランタイム ファブリック (RTF) に移行しています。Mule アプリケーションはいずれも Mule ドメイン プロジェクトを使用しません。

現在、すべての Mule アプリケーションは、顧客がホストする複数の Mule ランタイム間のサーバー グループに手動でデプロイされています。

これらの Mule アプリケーション デプロイメント間のポートの競合は、現在、Mule アプリケーションのプロパティ ファイルを注意深く管理する DevOps チームによって管理されています。Mule アプリケーションを現在の顧客ホストのサーバー グループからランタイム ファブリック (RTF) に移行する場合、Mule アプリケーションを書き直す必要がありますが、DevOps ポート設定の責任はどのように変更されるのか、それとも変わらないのでしょうか？

A. はい、Mule アプリケーションを書き直す必要があります

DevOps では Mule アプリケーション間のポート競合を管理する必要がなくなりました

B. はい、Mule アプリケーションを書き直す必要があります

DevOps は引き続きポートの競合を管理する必要があります。

C. いいえ、Mule アプリケーションを書き直す必要はありません

DevOps は依然としてポートの競合を管理しなければなりません

D. いいえ、Mule アプリケーションを書き直す必要はありません

DevOps では、Mule アプリケーション間のポートの競合を管理する必要はなくなりました。

Answer: C ([メッセージを残す](#))

説明

* Anypoint Runtime Fabric は、Mule アプリケーションとゲートウェイのデプロイとオーケストレーションを自動化するコンテナ サービスです。

* ランタイム ファブリックは、AWS、Azure、仮想マシン (VM)、またはベアメタル サーバー上の顧客管理のインフラストラクチャ上で実行されます。

* Mule アプリケーションはいずれも Mule ドメイン プロジェクトを使用しないため、アプリケーションを書き直す必要はありません。

また、アプリケーションが RTF にデプロイされている場合、デフォルトでは、インGRESは 8081 でのみ許可されます。

* したがって、DevOps チームがポートの競合を管理する必要はありません

最新問題: 36

プロジェクト要件の一部として、クライアントはデータのストリームを Mule アプリケーションに送信します。ペイロードのサイズは 10mb から 5GB の間で変化します。データを変換して複数の SFTP サーバー間で送信するには、Mule アプリケーションが必要です。組織のコスト削減のため、Mule アプリケーションには 0.2 仮想コアのサイズのワーカーを 1 つだけ割り当てることができます。

統合アーキテクトとして、このシナリオに対処するためにどのストリーミング戦略を提案しますか？

A. メモリ内の反復不可能なストリーム

B. ファイルベースの反復不可能なストリーム

C. メモリ内の反復可能なストリーム

D. ファイルベースの反復可能なストレージ

Answer: D ([メッセージを残す](#))

説明

質問にあるように、データは複数の SFTP サーバー間で送信する必要があるため、反復不可能なストリームを使用することはできません。反復不可能な戦略では反復可能なストリームが無効になり、入力ストリームを 1 回だけ読み取ることができます。

0.2 vcore では 1 GB のヒープ メモリしか取得できないため、メモリ ストレージでは使用できません。したがって、1 GB を超えるファイルに対してアプリケーションはエラーになります。

したがって、正しいオプションはファイルベースの反復可能なストリームです

最新問題: 37

ある製造会社は、Mule アプリケーションを自社の Azure Kubernetes Service インフラストラクチャにデプロイすることを計画しています。

この組織は、MuleSoft がホストするコントロール プレーンからすべての Mule アプリケーションを管理しながら、各 Mule アプリケーションを Docker コンテナ内の分離された Mule ランタイムにデプロイすることで、Mule アプリケーションの可用性と堅牢性を高めたいと考えています。これらの組織の要件を満たすランタイム プレーンの最も慣用的な (本来の目的に使用される) 選択は何ですか？

- A. Anypoint ランタイム ファブリック
- B. エニーポイント サービス メッシュ
- C. Anypoint Platform プライベート クラウド エディション
- D. クラウドハブ

Answer: ([解答を表示する](#))

最新問題: 38

システム API EmployeeSAPI は、基盤となる SQL データベースから従業員のデータを取得するために使用されます。

アーキテクトは、データベースで処理される冗長トランザクションの数を最小限に抑えるために、従業員の安定した更新がある場合にのみデータベースにクエリを実行するか、それ以外の場合はキャッシュされた応答を返すキャッシュ戦略を設計する必要があります。

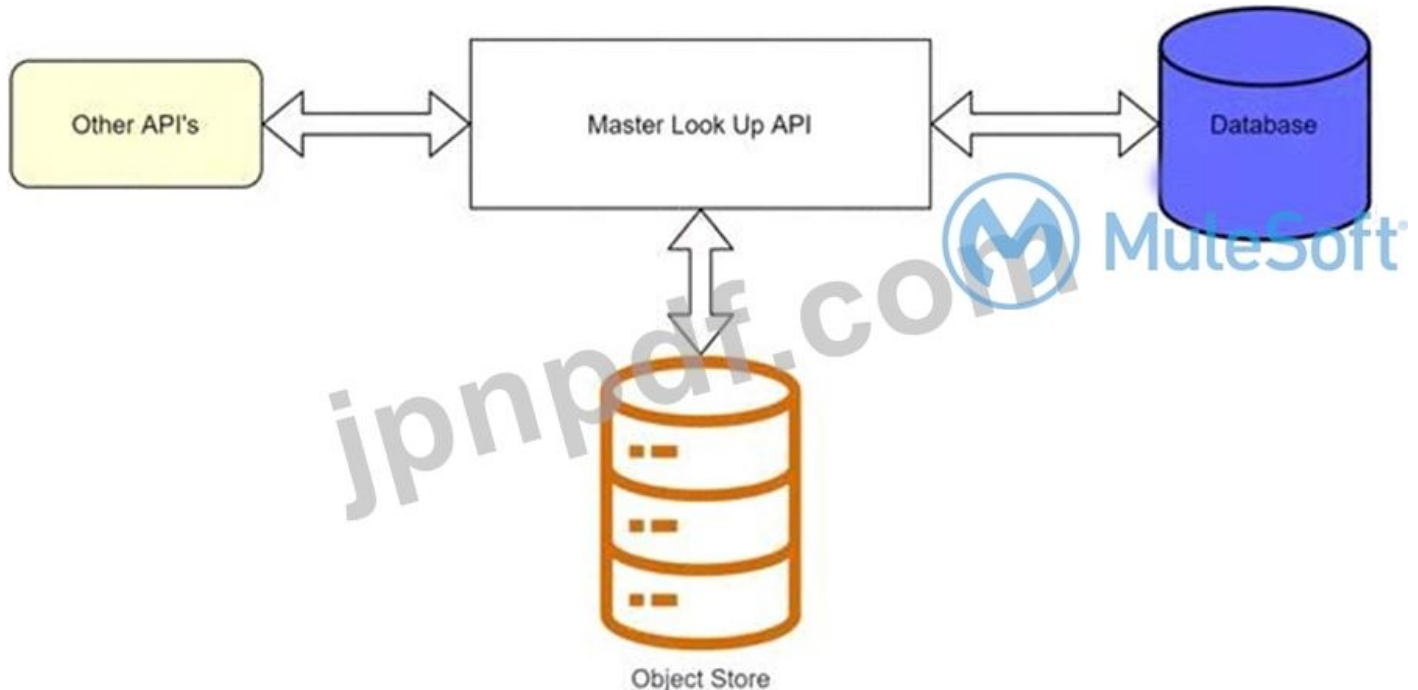
キャッシュの目的を達成するにはアーキテクトは何をしなければなりませんか？

- A. 1 時間ごとに固定頻度でスケジューラを使用し、無効化キャッシュ フローをトリガーします。オブジェクトストア キャッシュ戦略を使用し、有効期限間隔を 1 時間に設定します。
- B. 1 時間ごとに固定頻度でスケジューラを使用し、無効化キャッシュ フローをトリガーします。オブジェクトストアのキャッシュ戦略と有効期限間隔を使用して空にします。
- C. 従業員テーブル呼び出しでテーブル ルールを使用し、キャッシュを無効にし、キャッシュする新規従業員データをオブジェクトストア キャッシュ戦略を使用し、有効期限間隔を 1 時間に設定します。
- D. 従業員テーブルの On Table Row を使用し、キャッシュの無効化を呼び出します。オブジェクトストアのキャッシュ戦略と有効期限間隔を使用して空にします

Answer: D ([メッセージを残す](#))

最新問題: 39

ある銀行会社は、オンライン ビジネス用の新しい API セットを開発しています。重要な API の 1 つは、システム API であるマスター ルックアップ API です。このマスター検索 API は永続オブジェクトストアを使用します。この API は、マスター ルックアップ データを提供するために他のすべての API によって使用されます。



キャッシュされるマスター データが大量にあるため、マスター ルックアップ API はそれぞれ 0.1 仮想コアの 2 つの Cloudhub ワーカーにデプロイされます。マスター ルックアップ データはキーと値のペアとして保存されます。キーがキャッシュ内に見つからない場合、キャッシュは更新されます。

パフォーマンス テストを行うと、マスター ルックアップ データを取得するためのデータベースクエリの実行により、マスター ルックアップ API の応答時間が長くなることがわかりました。このパフォーマンスの問題により、オンライン ビジネスの稼働が保留されており、銀行に潜在的な経済的損失を引き起こす可能性があります。

統合アーキテクトとして、パフォーマンスの問題を解決するために次のオプションのうちどれをお勧めしますか？

- A. マスター ルックアップ API のすべての GET エンドポイントに HTTP キャッシュ ポリシーを実装します。
- B. 追加の Cloudhub ワーカーを追加して追加の容量を提供します
- C. 仮想コア サイズを 0.1 仮想コアから 0.2 仮想コアにアップグレードします。
- D. マスター ルックアップ API のすべての GET エンドポイントに HTTP キャッシュ ポリシーを実装し、オブジェクトストアへのアクセスを同期するためのロックを実装します。

Answer: D ([メッセージを残す](#))

最新問題: 40

保険会社は、MuleSoft がホストするランタイム プレーンに Mule アプリケーションを展開することを計画しています。要件の一部として、アプリケーションはスケーラブルである必要があります。可用性が高い。また、ログを少なくとも 2 年間保持することを要求する規制要件もあります。統合アーキテクトとして、これを達成するためにどのようなステップを推奨しますか？

- A. CloudHub 導入ではログを 2 年間保存することはできません。外部ログ管理システムが必要です。
- B. アプリケーションを CloudHub にデプロイする場合、ログの保持期間は 2 年として選択する必要があります
- C. アプリケーションを CloudHub にデプロイする場合、ワーカー サイズは 2 年分のデータを保存するのに十分である必要があります
- D. ロギング戦略は、アプリケーションとともにデプロイされた log4j ファイルで適切に構成する必要があります。

Answer: A ([メッセージを残す](#))

説明

正解は、CloudHub デプロイメントではログを 2 年間保存することはできません。外部ログ管理システムが必要です。ドキュメントに記載されているように、CloudHub には特定のログ保持ポリシーがあります。プラットフォームは、アプリおよびワーカーごとに最大 100 MB、または最大 30 日間のいずれかの制限に達した方のログを保存します。この制限に達すると、最も古いログ情報がまとめて削除され、回復不能に失われます。推奨されるアプローチは、ログ アペンダーを使用して、選択した外部ログ システム (Splunk など) にログを保存することです。この解決策により、ログが当社のプラットフォームに保存されなくなるため、サポート ケースを提出する場合は、レビューとケース解決のために適切なログを提供する必要があることに注意してください。

最新問題: 41

プロセス API の実装を変更する必要があります。API クライアントに対するこの変更の影響を最小限に抑える有効なアプローチは何ですか？

- A. プロセス API の RAML 定義が可能な限り変更されないように、プロセス API 実装に必要な変更を実装します。
- B. 現在のプロセス API の RAML 定義を更新し、更新された RAML 定義へのリンクを送信して API クライアント開発者に通知します。
- C. 新しいプロセス API または API バージョンに移行する準備ができたことを API コンシューマーが認識するまで、変更を延期します。
- D. 新しい API 実装でプロセス API の変更を実装し、古い API 実装が HTTP ステータス コード 301 - Moved Permanently を返すようにして、新しい API 実装を呼び出す必要があることを API クライアントに通知します。

Answer: A ([メッセージを残す](#))

説明

- * オプション B は、RAML が変更された場合、クライアントが変更に対応する必要があるため、非常に必要な場合を除き、使用しないでください。質問は、クライアントへの影響を最小限に抑えることに関するものです。したがって、これは有効な選択ではありません。
- * オプション C は、消費者の承認のためにビジネスを停止できないため、無効です。
- * オプション D は、クライアントが変更に対応する必要があるため、実行可能なオプションではありません。
- * 最良の選択は A です。ここでは、RAML 定義は変更されず、クライアントに依存せず、クライアントに影響を与えることなく、下線付きの機能が変更されます。

最新問題: 42

Mule アプリケーションは、HTTP リクエスト操作を使用して外部 API を関与させます。

外部 API は、ステータス コードを適切に使用するための HTTP 仕様に従います。

外部 API からの HTTP リクエスト操作に対して 3xx ステータス コードが返された場合、考えられる原因は何ですか？

- A. 外部 API でリクエストが受け付けられませんでした。
- B. リクエストは外部 API によって別の URL にリダイレクトされました
- C. リクエストは外部 API によって受信されませんでした
- D. リクエストは外部 API によって受け入れられました

Answer: B ([メッセージを残す](#))

説明

3xx HTTP ステータス コードは、ユーザー エージェント (Web ブラウザまたはクローラー) が特定のリソースにアクセスしようとするときにさらなるアクションを実行する必要があることを示すリダイレクトを示します。

Valid MCIA-Level-1-Maintenance Dumps shared by GoShiken.com for Helping Passing MCIA-Level-1-Maintenance Exam! GoShiken.com now offer the **newest MCIA-Level-1-Maintenance exam dumps**, the GoShiken.com MCIA-Level-1-Maintenance exam **questions have been updated** and **answers have been corrected** get the **newest** GoShiken.com MCIA-Level-1-Maintenance dumps with Test Engine here:
<https://www.goshiken.com/MuleSoft/MCIA-Level-1-Maintenance-mondaishu.html> (118 Q&As Dumps, **30%OFF Special Discount: Freepdfdumps**)