

HCL Volt MX Go VMXGo-DEV-100

Lesson 3 - Design Import Part 2



もくじ

はじめに.....	3
前提条件.....	4
Lesson 3 – Design Import Part 2.....	5
まとめ	13
法的ステートメント	14
免責事項.....	15

はじめに

HCL Volt MX Go VMXGo-DEV-100 トレーニングコースは、現在のリリースである Volt MX Go v2.0.1 (<https://opensource.hcltechsw.com/voltxgo-documentation/references/whatsnew.html>) 以降の HCL Volt MX Go ツールを学ぶための開発者向けスタートコースです。

Volt MX Go の開発者が知っておかなければならない 2 つの主要なツールは、1) Design Import と 2) VoltFormula です。Design Import は、Volt MX Go Iris (別名 Volt Iris) のプロジェクト、UI、Domino とデータを交換するロジックをゼロから作成する必要がなく、開発者の時間と労力を大幅に削減します。VoltFormula を使用すると、依存する Domino ロジックを HCL Volt MX プロジェクトで使用できるため、JavaScript でロジックを書き直す時間と労力を節約できます。

Design Import は、Domino バージョン 12.0.2 以降でホストされている HCL Domino アプリケーション (Domino REST API、別名 DRAPI に公開済み) を Volt MX Go の Volt Iris プロジェクトと Volt MX Go の Volt Foundry アプリ (Foundry ミドルウェアサービスのコレクション) にインポートします。Design Import の最終的な出力は、Volt Iris Web アプリで、すぐに機能し、完全に開発され、Domino アプリケーションを表すすべての Volt Iris フォームとウィジェットが含まれ、OAuth2 Identity サービス、Integration サービス、Volt MX Go の Foundry Domino アダプターを使用する Object サービスを持つ Foundry アプリに関連付けられています。

Design Import 後における Volt MX Go アプリ開発の一般的な流れは、組織のブランディングや UI 要件に合わせて UI のリブランド/リファクタリングを行い、Domino 文書のリストにソートやフィルタリング機能を追加することです。

この HCL Volt MX Go VMXGo-DEV-100 トレーニングには、上記を扱う 6 つのレッスンが含まれています。レッスンは以下の通りです。

1. Lesson 1 - Domino REST API 必須情報
2. Lesson 2 - Design Import のセットアップ
3. Lesson 3 - Design のインポート
4. Lesson 4 - VoltFormula
5. Lesson 5 - UI のリブランディング
6. Lesson 6 - セグメントのソートとフィルタリング

このコースでは、HCL Volt MX Go First Touch Recipe Catalog アプリとその資産 (Domino DB (レシピ保存用)、First Touch Recipe Domino REST API スキーマ、スコープ、DRAPI アプリ

(<https://opensource.hcltechsw.com/voltxgo-documentation/tutorials/firsttouch.html>) を含む) を活用します。DRAPI First Touch Recipe アプリで Design Import を実行し、Volt Iris アプリに VoltFormula を追加し、Iris アプリのログイン画面とダッシュボード画面/フォームをリブランドし、Iris アプリにソートとフィルタリング機能を追加します。

前提条件

このコースを修了するには、HCL Volt MX Go の Volt Foundry (ミドルウェア) と Volt Iris (IDE) に加え、Domino REST API を含む Domino 環境が必要です。Domino と Volt MX Go サーバーのオンプレミスインストールの代わりに、HCL SoFy プラットフォーム (<https://hclsofy.com>) の HCL Volt MX Go サンドボックスを使用できます。HCL SoFy サンドボックスには、Domino、Domino REST API、および Volt MX Go Foundry が含まれます。SoFy が提供するトライアルサンドボックスを使用するには、付録 I を参照してください。

オンプレミス

- HCL Domino server 12.0.2+
- HCL Domino REST API (DRAPI) サービス (タスクとサービスが稼動)
- HCL Domino REST API Console URL
- HCL Domino REST API Admin User Credentials (ユーザーID とパスワード)
- HCL Volt MX Go Foundry v2.0.1
- HCL Volt MX Go Foundry Console URL
- HCL Volt MX Go Foundry Admin User Credentials (ユーザーID とパスワード)
- HCL Volt MX Go Iris v2.0.1

HCL SoFy プラットフォーム (<https://hclsofy.com>)

- HCL Volt MX Go サンドボックス

Lesson 3 – Design Import Part 2

Volt MX Go Design Import ウィザードには、Recipe Domino DB を Go Volt Iris Web/デスクトッププロジェクトにインポートするために必要な 6 つのステップがあります。下の画像を参照してください。

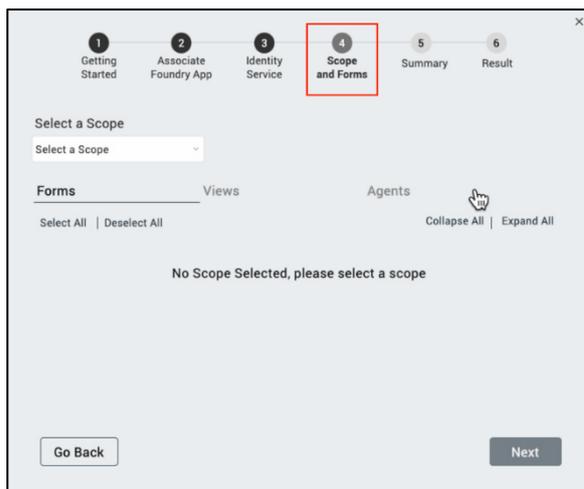


引き続き Volt MX Go Design Import Part 2 のレッスンを行います。このパート 2 では Design Import プロセスを完了します。

Lesson 3 – Design Import Part 2 では、前のレッスンの Design Import プロセスの Step 4 – Scope and Forms を継続します。ここでは、Volt に変換するフォーム、ビュー、エージェントを First Touch Recipe Domino DB から選択し、Iris デスクトップ/Web プロジェクトに変換します。Design Import のステップ 5 および 6 に続き、プロセスを完了します。

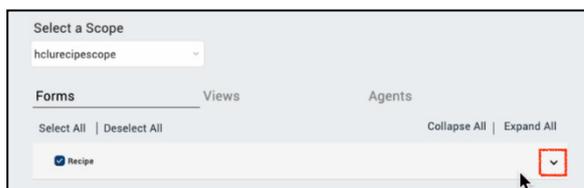
Design Import が完了すると、**Foundry App** に Objects サービスが追加され、Volt MX Go の HCL-Domino-Adapter を使用してアプリが Domino データを送受信できるようになります。Foundry Objects サービスには選択したすべての Domino フォームとビューが含まれます。

Volt Iris 側では、Design Import が Domino DB に一致するすべての Iris フォームとビューを作成します。Iris は、フォームとビューが Domino ドキュメントを表示し、Domino ドキュメントを作成、編集、削除できるように、すべてのロジックとアクションスクリプトを作成します。Iris プロジェクトは、Foundry アプリにも自動的にリンクされます。プレビュービルドやビルド & パブリッシュを実行して、Web アプリをテストできる状態になります。



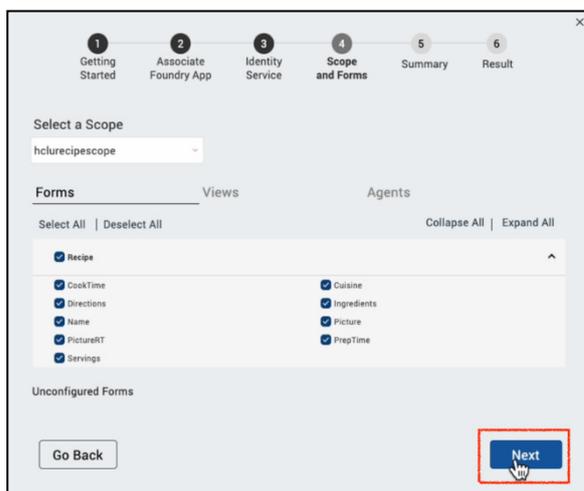
ステップ

1. Volt Go Iris IDE から、プルダウンコントロール Select a Scope をクリックします。
2. 複数のスコープが表示されるかもしれません。「Lesson 1 - Domino REST API 必須情報」で記録した First Touch Recipes アプリのスコープを選択します。
3. Forms タブの下にチェックマーク付きの Recipe が表示されます。レシピの右端にある下向き矢印をクリックして Domino フォームを展開します

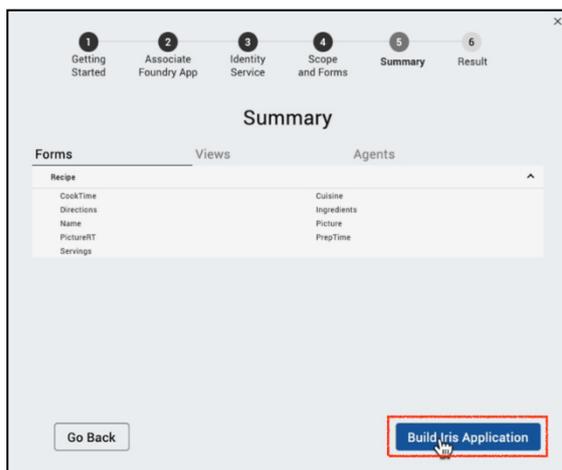


4. CookTime、Directions、Name などの Recipes データフィールドがチェックマーク付きで表示されます。
5. \$FILES の Domino フィールドのチェックを外します。Design Import 中に、必要なフィールドを選択または選択解除できます。このレッスンでは\$FILES は必要ありません。
6. Views タブをクリックすると既にチェックされている単一の\$All ビューが表示されます。
7. エージェントタブをクリックし、Recipes Domino アプリにエージェントがないことに注意してください。

8. チェックマークは、Domino フォーム、フィールド、ビュー、およびエージェントが Volt Iris 相当のウィジェットに変換されることを示しています。変換したくない項目のチェックを外すことができます。このレッスンではすべてを変換します（すべてのチェックマークをオンにします）。
9. Next をクリックして **Step 5 – Summary** に進みます。



10. Summary で、選択したフォーム、データフィールド、ビュー、エージェントがインポートされようとしていることを確認できます。
11. **Build Iris Application** ボタンをクリックしてインポート処理を開始します。



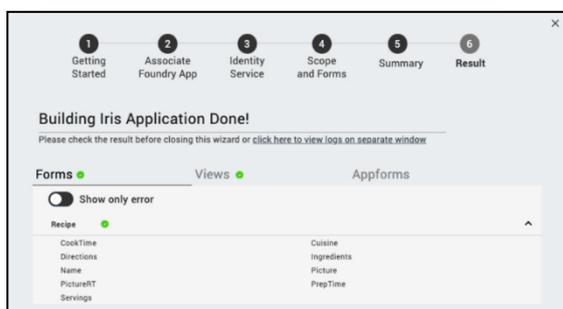
12. インポート中、Iris プロジェクトと Foundry アプリの両方が影響を受けます。Foundry アプリには、Volt MX Go Foundry Domino アダプターを使用して、選択したすべての Domino フォーム、データフィールド、ビューを生成する新しい Foundry Object サービスが含まれま

す。インポートにより、Foundry アプリが再パブリッシュされます。Volt Iris では、インポートによって、Domino ビューを表す Iris フォームを含め、レシピ アプリを表示し、レシピを一覧表示、作成、編集、削除するために必要なすべての Iris フォーム（Volt Iris 画面）が作成されます。

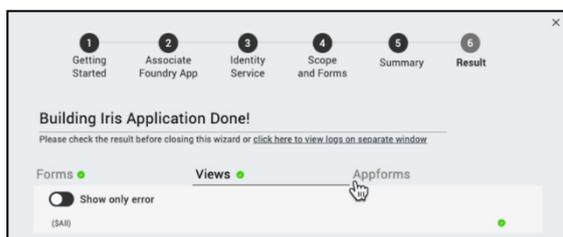
13. Design Import が完了すると **Building Iris Application Done!**が表示されます。

14. Forms、Views、Appforms の3つのタブが表示されます

15. Forms タブには、インポートした Iris フォームと、各フォームのデータフィールドが一覧表示されます。



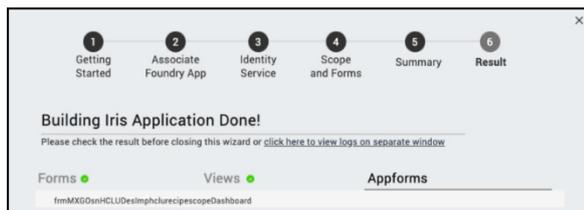
16. Views タブには、インポートした Domino ビューが一覧表示されます。このケースでは (\$All) フォームが一つみえているはずです。



17. Appforms タブには、「ダッシュボード」 Iris フォームが一覧表示されます。これらのダッシュボードフォームは、すべての Domino 文書を表示したり、文書を作成・管理したりするためのメイン画面となります。レシピアプリでは Appform は一つだけです。

- “frmMXGO”
- 前のレッスン「Lesson 3 - Design Import パート 1」の Step 5 で設定したサービス名
- Domino REST API スコープ名（「レッスン 1 - Domino REST API 必須情報」で設定）

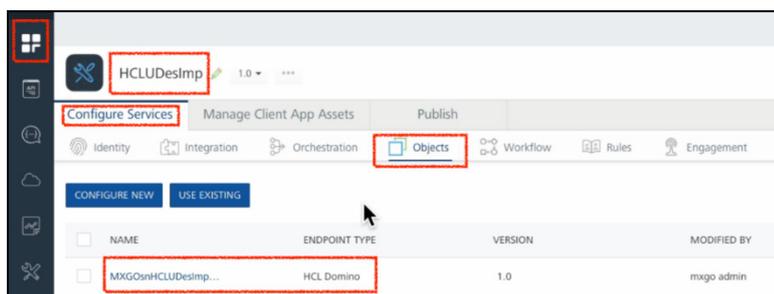
- “Dashboard”



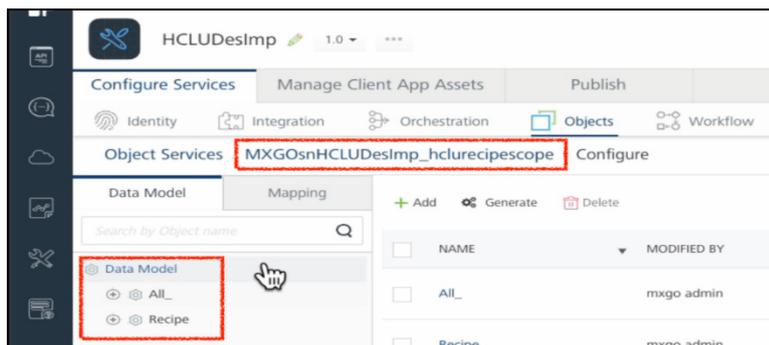
18. DONE をクリックして Design Import を終了し、Volt Iris キャンバスに戻ります。

19. Design Import によって、Foundry と Domino REST API の間でデータを交換するための Foundry Object サービスが Foundry アプリに作成されたことを確認するには、次の手順を実行します。

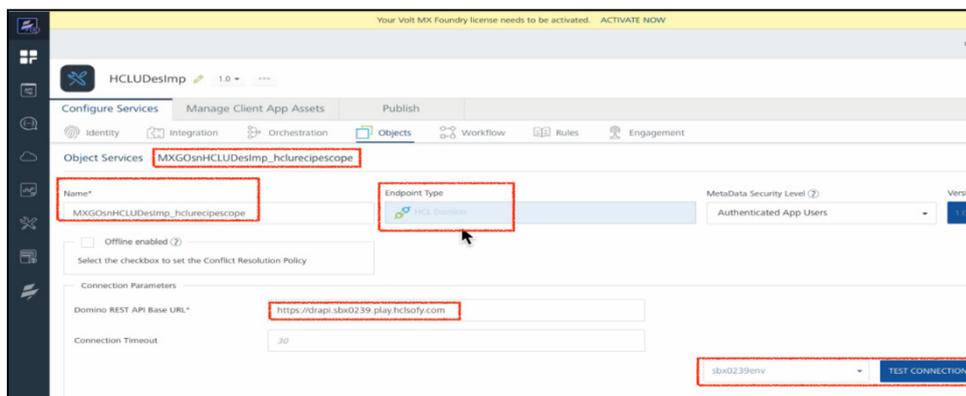
- Volt Go Foundry コンソールを起動し、Sofy から提供された、または自分のローカルの Foundry から提供された Volt MX クレデンシャルを使用してログインします。
- Foundry コンソールで次の場所に移動します。Apps -> Foundry アプリ（Volt Iris プロジェクトと同じ名前） -> Configure Services -> Objects



- エンドポイントタイプが HCL Domino の新しい Foundry Object サービスが表示されるはずですが、新しい Object サービスの名前は次のようになります。
 - “MXGO”
 - 前のレッスン (Lesson 3 - Design Import Part 1) の Step 5 で指定したサービス名
 - アンダースコア
 - Domino REST API スコープ名 (「Lesson 1 - Domino REST API 必須情報」で設定)
- Objects 画面から **Object service name** をクリックして **Data Models** を表示します。All_と Recipe Data Models が表示されます。



- 各データモデルを展開して、すべてのデータフィールドを表示し、Mapping タブの各データモデルで利用可能な操作を確認できます。
- オブジェクトサービスの詳細を表示するには、"Object Services"と"Configure"の2つのラベルの中央にあるオブジェクト名をクリックします。

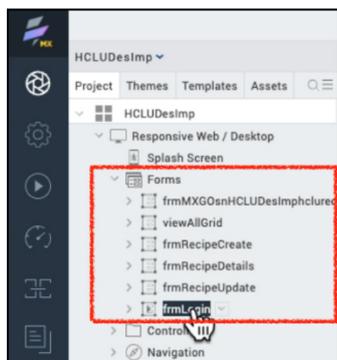


- エンドポイントタイプとサービス名が正しいことを確認します。
- Domino REST API Base URL が Sofy から提供されたもの、または独自の環境であることを確認します。
- Foundry 環境を選択し、**Test Connection** をクリックして、Foundry が Domino REST API サーバーにアクセスできることを確認できます。

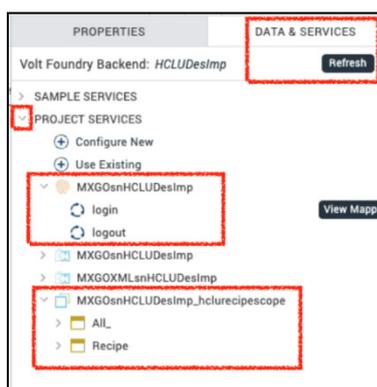
20. Design Import で正しい Iris フォームが作成され、Volt Iris プロジェクトが最新バージョンの Foundry アプリにリンクされていることを以下の手順で確認します。

- Volt Iris IDE の左端にある Project タブの下に、ツリー構造 <Iris プロジェクト名> → Responsive Web / Desktop → Forms が表示されます。

- Forms の下に、Design Import が Recipes アプリ用に作成した以下のフォームが表示されているはずです：



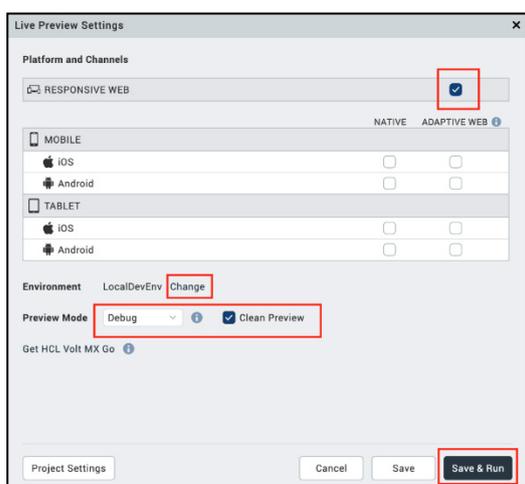
- 各 Iris フォームを展開すると Iris のセンターキャンバスにその画面が表示されます。
- Iris で Foundry アプリの最新バージョンにリンクされていることを確認するには Iris の右端の **Data & Services** タブを使用します。Foundry アプリの最新バージョンは Design Import によって、Foundry Domino アダプターを使用し、Domino フォームとビューのデータ モデルを備えた新しい Foundry オブジェクト サービスに更新されています。
- Data & Services** タブ -> **Refresh** をクリックします。
- PROJECT SERVICES** の左にある下矢印をクリックするとすべての Foundry サービスが表示されます。



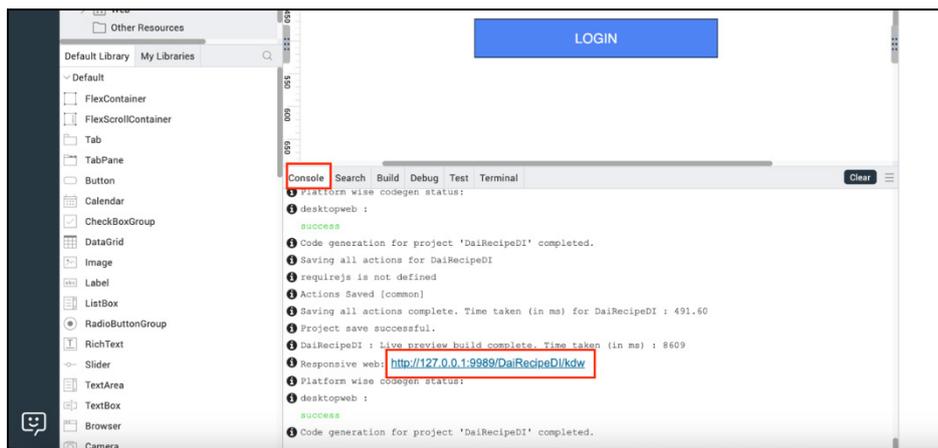
- Foundry OAuth2 ID サービスと Foundry Object サービスが表示されているはずです。

これでプレビューWeb アプリをビルドし、テストする準備ができました。

1. Iris のトップレベルメニューから **Project** -> **Save** または **Save All** をクリックして Iris プロジェクトを保存します。
2. これで、プロジェクトのプレビュービルドを行う準備ができました。Iris のトップレベルメニューから、**Build** -> **Live Preview** 設定をクリックします。
3. ライブプレビューの設定で、**Responsive Web** チャンネルのみがチェックされていることを確認します。**Clean Preview** にチェックを入れます。Preview Mode は Debug です。Foundry 環境が正しいことを確認し、必要であれば **Change** をクリックして設定します。最後に **Save & Run** をクリックして、Web プレビューアプリをビルドして実行します。



4. プレビュービルドが完了すると、Volt MX Go Iris Preview デバッグブラウザのポップアップが表示されます。ログインして Recipes アプリをテストできます。別のブラウザでテストしますのでデバッグブラウザは閉じてかまいません。
5. ブラウザからプレビューアプリをテストするには、次のようにします。
 - Volt Iris IDE の中央下のセクションに、Console、Search、Build、Debug、Test、Terminal のタブがあります。**Console** をクリックします。
 - メッセージウィンドウを一番下までスクロールします。Response web: <http://127.0.0.1:9989/<あなたの Iris プロジェクト名>/kdw> というリンクがあるはずです。



- このリンクをクリックし、任意のブラウザで Web アプリを起動してください。

Domino の All ビューですべてのレシピを一覧表示できますか？レシピの一覧は、First Touch Recipe Web-app (Lesson 1 Appendix II - Volt MX Go Get Started) にあるものと一致していますか？レシピの詳細を表示できますか？Domino レシピフォームを使って新しいレシピを作成できますか？

まとめ

Design Import が完了しました。この時点で、Design Import によって完全な Iris プロジェクトが作成され、アプリの実行と Recipes 文書の管理に必要なロジックがすべて含まれた Web アプリが作成されました。また、Foundry アプリには、最低でも OAuth2 ID サービスが1つと Object サービスが1つあります。これで、Responsive Web チャンネルをターゲットにした Live Preview ビルドを実行したり、Build and Publish Web を実行して Web アプリをビルドして一般公開できます。

法的ステートメント

このエディションは、HCL Volt MX Go のリリース 2.0.1、および新しいエディションで別段の記載がない限り、それ以降のすべてのリリースおよび変更に適用されます。

あなたが HCL Technologies Ltd.に情報を送信する場合、あなたは HCL Technologies Ltd.に、あなたに対していかなる義務を負うことなく、適切と思われる方法で情報を使用または配布する非独占的な権利を付与します。

©2023 Copyright HCL Technologies Ltd and others. 無断複写・転載を禁じます。

米国政府ユーザーへの注意 - 制限された権利に関連する文書 - 使用、複製、または開示は、HCL Technologies Ltd.との GSA ADP スケジュール契約に規定された制限に従うものとしします。

免責事項

本レポートは、HCL 利用規約 (<https://www.hcl.com/terms-of-use>) および以下の免責事項の対象となります：

本レポートに含まれる情報は、情報提供のみを目的としています。本レポートに含まれる情報は、情報提供のみを目的として提供されるものであり、本書に含まれる情報の完全性および正確性を確認するよう努めたが、商品性、非侵害性、特定目的への適合性の黙示保証を含むがこれに限定されない、明示または黙示を問わずいかなる保証もなく、現状のまま提供されるものである。また、本情報は、HCL 社の現在の製品計画および戦略に基づいており、HCL 社により予告なく変更される場合があります。HCL は、本レポートまたはその他の資料の使用またはその他の関連から生じる直接的、間接的、偶発的、結果的、特別またはその他の損害について責任を負わないものとします。本書に含まれるいかなる内容も、HCL 社またはその供給業者やライセンサーによる保証や表明を意図するものではなく、またそのような効果をもたらすものでもありません。

本レポートにおける HCL の製品、プログラム、サービスへの言及は、HCL が事業を展開するすべての国でそれらが利用可能になることを意味するものではありません。本プレゼンテーションで言及されている製品のリリース日や機能は、市場機会やその他の要因に基づき、HCL の独自の裁量で随時変更される可能性があり、将来の製品や機能の提供を約束するものではありません。これらのレポートをサポートするために使用される基礎データベースは、毎週更新されます。この Web ツールを使用して生成されたレポートと他の HCL ドキュメンテーションソースの間に見られる不一致は、このツールと他のソースの公開および更新サイクルが異なることに起因する場合も、そうでない場合もあります。本レポートに含まれるいかなる内容も、あなたが行った活動が特定の売上、収益の増加、節約、またはその他の結果をもたらすことを意図したものではなく、またそのような効果を持つものでもありません。利用者は、本レポートの結果として利用者が得た結果または利用者が行った決定について、単独で責任を負うものとします。HCL 利用規約 (<https://www.hcl.com/terms-of-use>)にかかわらず、本サイトの利用者は、本ツールから生成されたレポートを利用者自身の内部業務目的のためにコピーおよび保存することが許可されています。それ以外の使用は許可されません。