

日々是Oracle APEX

Oracle APEXを使った作業をしていて、気の付いたところを忘れないようにメモをとります。

2022年3月3日 木曜日

Autonomous Databaseの複数ドメインのサポートを確認する

オラクルの公式ブログと先日のOffice Hourにて、製品マネージャーのTodd Bottgerさんより Autonomous DatabaseのAPEXで複数ドメインが利用できるようになった、と告知がありました。

ブログの記事は以下になります。

[Oracle Autonomous Database Vanity URLs Part 4: Multiple Domain Names](#)

Office Hourは以下より録画を視聴できます。

[Vanity URL Extensions for APEX & ORDS \(2022年2月17日\)](#)

複数ドメインの確認は難しいため、単一のAutonomous DatabaseのAPEXに複数のホスト名を割り当てる設定を行ってみました。

以前の記事[OCIの証明書サービスを使う](#)に従って構築した環境を使います。元の英語の記事では Autonomous Databaseはプライベート・エンドポイントで構成されていますが、それは有料なので使わず、Customer Managed ORDSを構成することにより独自ホスト名を割り当てられるようにしています。

ロード・バランサはtestsrvlbとして作成しています。割り当てられたパブリックIPアドレスは、DNSより2つのホスト名webprod.apexugj.devおよびwebtest.apexugj.devから参照されるようにしています。手順が分かりにくくなるので本記事では検証に使用したホスト名をそのまま記載しますが、**ホスト名の部分はそれぞれの環境に合わせて置き換えて作業する必要があります。**

構成されている環境の概要は以下です。仮想プライベート・ネットワークをMyAPEXDEVNetとして作成していますが、データベース、コンピュート、ロード・バランサなど全てが仮想プライベート・ネットワーク内の同一のパブリック・ネットワークに含まれているので、説明からは省きます。

1. Always FreeのAutonomous Transaction ProcessingをAPEXDEVとして作成。
2. コンピュート・インスタンスを作成し、Customer Managed ORDSを構成。
3. OCIの証明書サービスを使い、webprod.apexugj.devのサーバー証明書を生成。
4. 作成したコンピュート・インスタンスをバックエンドとするロード・バランサtestsrvlbを作成。
5. APEXのワークスペースとしてAPEXDEVを作成。

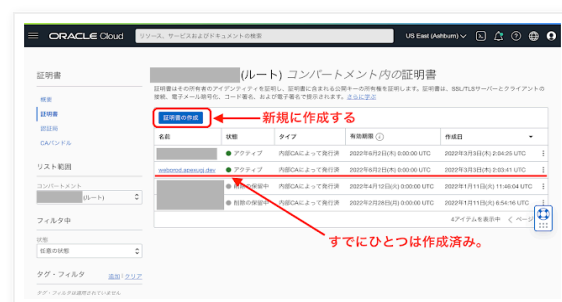
ここまでの手順は以前の記事通りですが、ホスト名はtestsrv.apexugj.devではなく、webprod.apexugj.devとしてアクセスできるように構成しています。

証明書の準備

サーバー証明書はOCIの証明書サービスを使わなくても準備できます。とはいえ、最も準備が容易でかつ無料です。以前の記事でもOCIの証明書サービスを使用しているため、同じ手順で追加の証明書を作成します。

受け付けるホスト名を**共通名(Common Name)**および**サブジェクトの代替名(Subject Alternative Name)**とした証明書を発行します。

OCIのコンソールより**証明書サービス**を開きます。

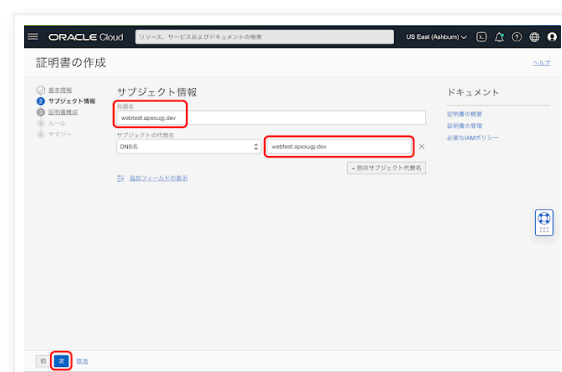


ロード・バランサを作成するにあたって、すでにひとつは証明書が発行済みになっているはず。まったく同じ手順にそって、新たに証明書を発行します。

コンパートメントにルート、証明書タイプに**内部CAによって発行済**を選択します。名前は**webtest.apexugj.dev**としていますが、この名前はホスト名である必要はありません。説明を入力し、次へ進みます。

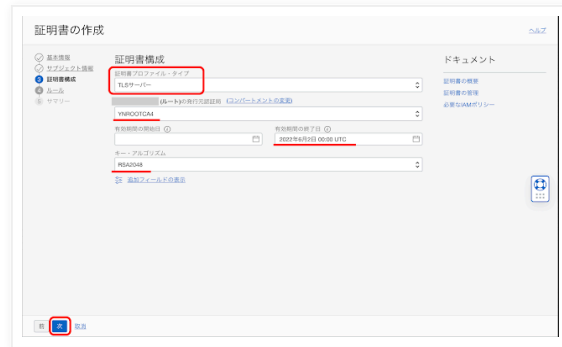


サブジェクト情報の共通名として、追加するホスト名**webtest.apexugj.dev**を入力します。サブジェクトの代替名としても**webtest.apexugj.dev**を入力し、次へ進みます。

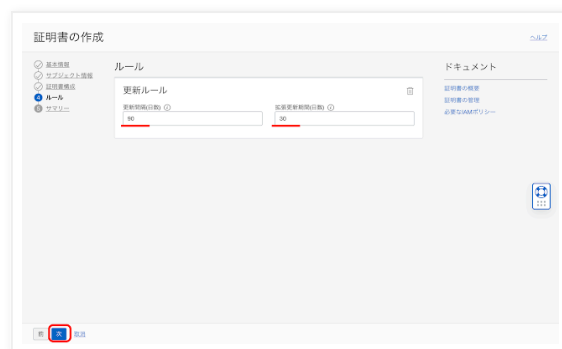


同じパブリックIPアドレスに割り当てられている異なるホスト名であれば、最初に **webprod.apexugj.dev** 向けの証明書を発行する際に、**別のサブジェクト代替名**として **webtest.apexugj.dev** も追加しておけばよいのですが、今回の作業はドメイン名が異なる場合の検証も含むため、新たに証明書を発行しています。

証明書構成の証明書プロファイル・タイプとして**TLSサーバー**を選択します。それ以外はデフォルトの設定のまま、**次**へ進みます。



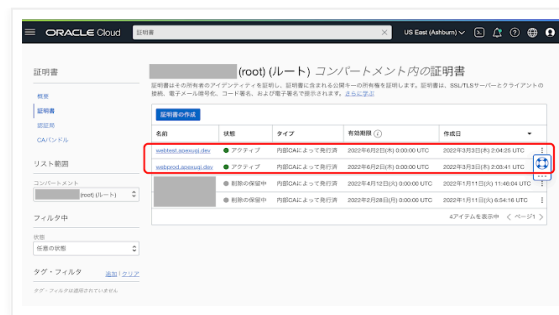
更新ルールの更新間隔(日数)と**拡張更新期間(日数)**はともに、デフォルトの**90**、**30**のまま変更せず、**次**へ進みます。



サマリーを確認し、**証明書の作成**を実行します。



以上で、2つのホスト名 **webprod.apexugj.dev**、**webtest.apexugj.dev** で利用できるサーバー証明書が発行済になりました。

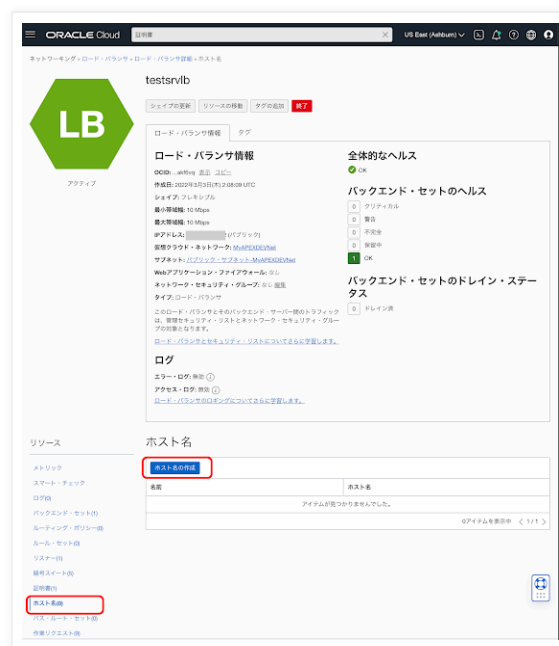


ホスト名の登録

ロード・バランサにホスト名を登録します。

作成済みのロード・バランサtestsvrlbのロード・バランサ詳細ページを開き、リソースからホスト名を選択します。

ホスト名の作成をクリックします。

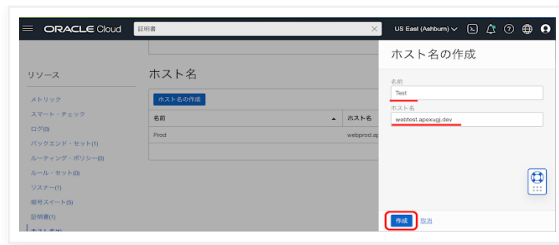


画面右にドロワーが開きます。名前にProd、ホスト名としてwebprod.apexugj.devを入力します。

作成をクリックします。



ホスト名としてwebprod.apexugj.devが登録されます。再度、ホスト名の作成をクリックし、名前としてTest、ホスト名としてwebtest.apexugj.devを入力し、作成をクリックします。



結果としてロード・バランサtestsrvlbに、ホスト名webprod.apexugj.devおよびwebtest.apexugj.devが登録されます。



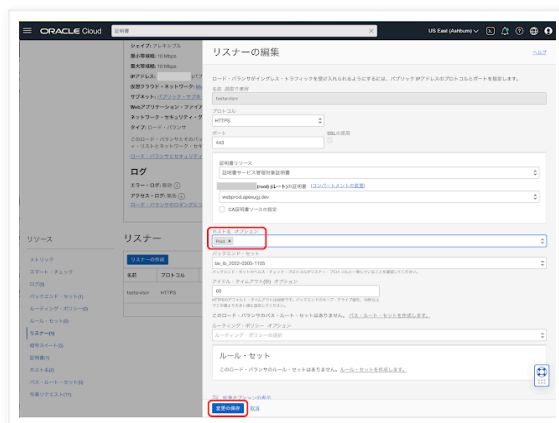
リスナーの構成

すでに作成済みのリスナーtestsrvlsnrにホスト名を割り当てます。ロード・バランサ詳細のリソースより、**リスナー**を選択します。

既存のリスナーtestsrvlsnrのメニューを開き、**編集**を実行します。



ドロワーが開きます。その中の**ホスト名**としてProd（つまりwebprod.apexugj.dev）を割り当てます。**変更の保存**をクリックし、ドロワーを閉じます。



リスナーのホスト名としてProdが表示されます。

ホスト名webtest.apexugj.devを割り当てるリスナーを作成します。

リスナーの作成をクリックします。

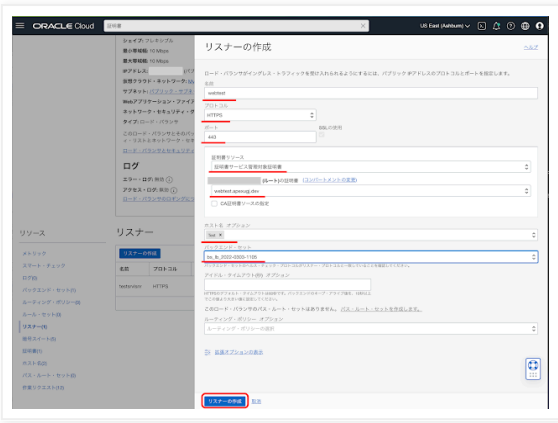


リスナーの作成のドロワーが、画面右に開きます。設定内容は、既に作成済みのリスナー `testsrvlsnr` とほぼ同様になります。

名前は任意ですが、ここでは `webtest` としています。プロトコルは `HTTPS`、ポートは `443` を選択します。証明書リソースとして証明書サービス管理対象証明書、証明書として `webtest.apexugj.dev` を選択します。

ホスト名として `Test` を選択します。バックエンド・セットとして、既に作成されているバックエンド・セット（ひとつのみのはず）を選択します。

リスナーの作成をクリックします。



リスナー `webtest` が新たに作成されます。



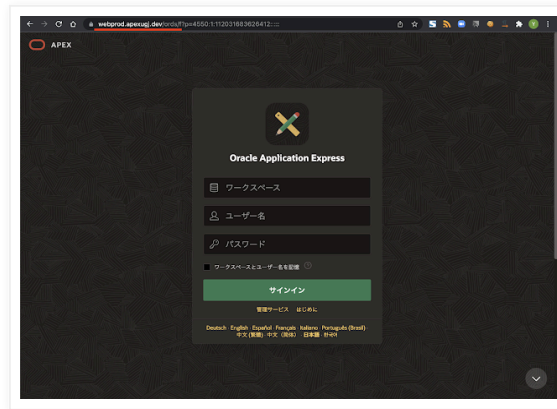
以上で登録したホスト名で、ロード・バランサが接続を受け付けるようになりました。

接続確認

最初に以下のURLでアクセスします。

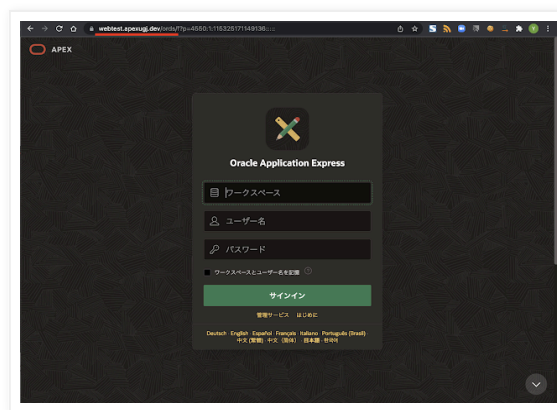
`https://webprod.apexugj.dev/ords/`

アプリケーション・ビルダーへのサインイン画面が開きます。



同様に以下のURLでもアクセスします。

<https://webtest.apexugj.dev/ords/>



同じ画面が表示されます。

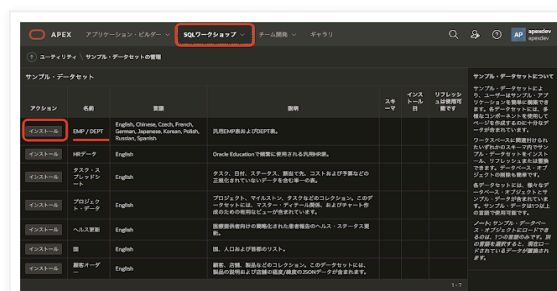
ホスト名が違っていても、両方とも同じAutonomous Databaseで実行されています。

アプリケーションの作成

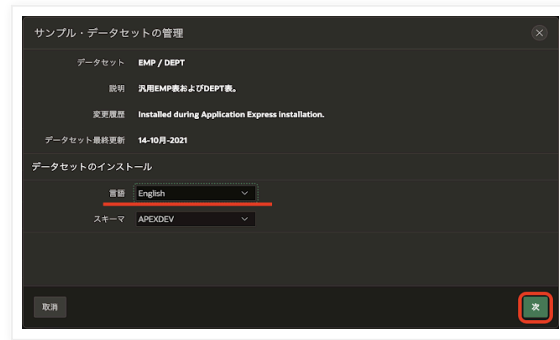
webprod.apexugj.devにサインインし、EMP / DEPTのサンプル・アプリケーションを作成します。

SQLワークショップのユーティリティより、**サンプル・データセット**を開きます。

EMP / DEPT のインストールを実行します。



(アプリケーションにアクセスするURLを英語にするため) 言語に**English**を選び、次へ進みます。



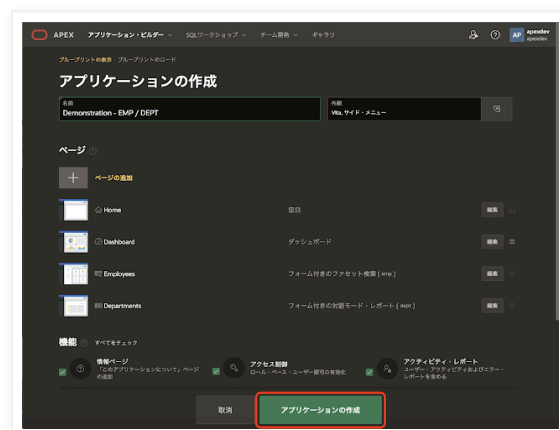
データ・セットのインストールを実行します。表EMP、DEPTおよびビューEMP_DEPT_Vが作成されます。



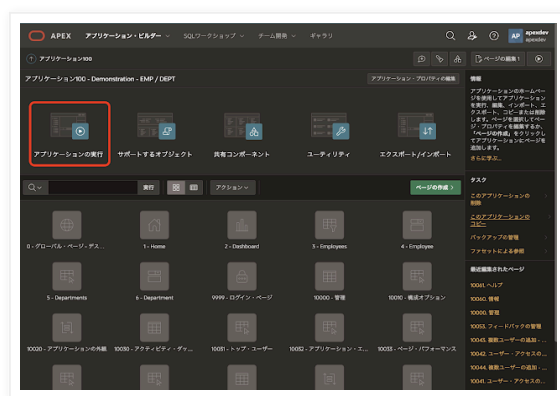
アプリケーションの作成を実行します。インストールしたデータ・セットを元にしたアプリケーションが作成されます。



アプリケーション作成ウィザードが起動します。特に設定の変更はせず、アプリケーションの作成を実行します。



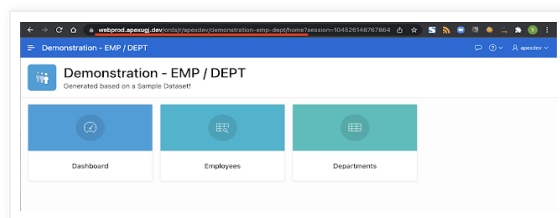
アプリケーションが作成されたことを確認し、**アプリケーションの実行**を行います。



実行したアプリケーションにサインインします。

画面のURLは以下で始まっています。

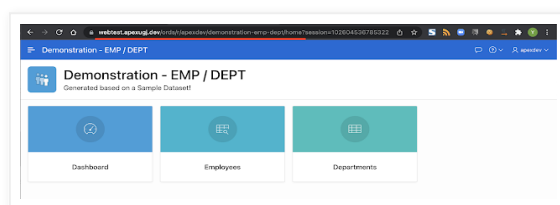
<https://webprod.apexugj.dev/ords/r/apexdev/demonstration-emp-dept/home>



webprodの部分をwebtestに変更してアクセスします。

<https://webtest.apexugj.dev/ords/r/apexdev/demonstration-emp-dept/home>

ユーザー名、パスワードを入力してサインインすると、同じアプリケーションが実行されます。



一般的な要件として、異なるURLで同じアプリケーションにアクセスできるようにしたい、というのはあまりありません。異なるURLまたはホスト名でのアクセスを認識して、アクセス可能なアプリケーションを限定したい、という場合が多いでしょう。

初期化PL/SQLコードによるホストの制限

英語の記事では、以下のPL/SQLコードをAPEXアプリケーションに設定することにより、アクセス可能なドメインを制限しています。

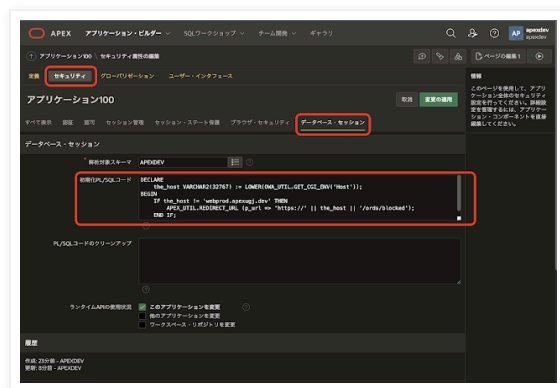
DECLARE

```
the_host VARCHAR2(32767) := LOWER(OWA_UTIL.GET_CGI_ENV('Host'));
```

```
BEGIN
```

```
IF the_host != 'webprod.apexugj.dev' THEN
  APEX_UTIL.REDIRECT_URL (p_url => 'https://' || the_host || '/ords/blocked');
END IF;
END;
```

アプリケーション定義のセキュリティのデータベース・セッションにある、初期化PL/SQLコードとして記述します。初期化PL/SQLコードは、APEXのページ・プロセスの先頭で必ず実行されます。



OWA_UTIL.GET_CGI_ENV('Host')の呼び出しにより、URLに含まれるホスト部分が取り出されます。PL/SQLのブロック内のIF文の条件で、取り出されたホスト名（この場合webprod.apexugj.dev）に一致しなければ、https://webtest.apexugj.dev/ords/blockedへリダイレクトします。

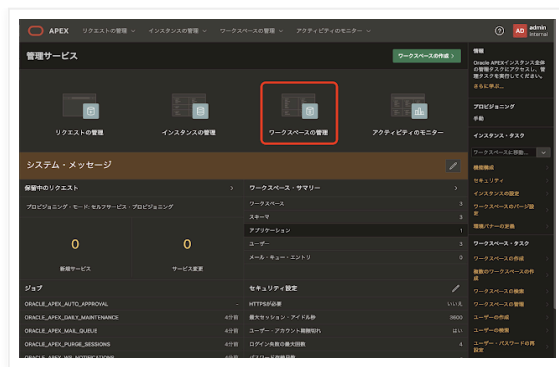
結果としてwebtest.apexugj.devを指定してアプリケーションにアクセスすると、以下のようなエラーが表示されます。



ワークスペースの分離

現時点ではAutonomous Databaseではサポートされていませんが、Oracle APEXにはワークスペースの分離という設定があります。

管理サービスにサインインし、ワークスペースの管理を開きます。



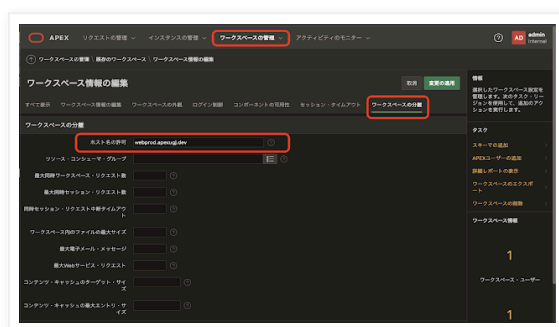
ワークスペース・レポートの既存のワークスペースを開きます。



編集するワークスペースを開きます。



ワークスペースの分離のタブを選択し、**ホスト名の許可**にホスト名を設定します。開発ツールを含むワークスペース内のすべてのアプリケーションに対して、ここで指定したホスト名が含まれたURLによるアクセスのみが許可されます。カンマ区切りで複数のホスト名が指定可能です。



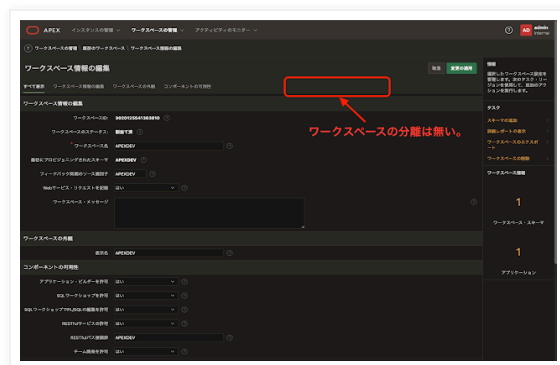
ワークスペースの分離として設定されていないホスト名が含まれるURLによって、APEXアプリケーションにアクセスすると**HTTP 404 Not Found**のエラーが発生します。



この仕組みを使うことにより、以下のような構成が可能になります。

- ワークスペースA = <https://www.domain-a.com>からのみアクセス可。
- ワークスペースB = <https://www.domain-b.com>からのみアクセス可。

ちなみにAutonomous Databaseで**既存のワークスペース**の編集画面を開くと、**ワークスペースの分離**が設定項目として無いことが確認できます。



オラクルの公式のドキュメントで制限について記載されている部分は以下になります。

<https://docs.oracle.com/en/cloud/paas/autonomous-database/adbsa/apex-restrictions.html#GUID-E13D5044-B9DD-4168-8A12-C99532940DA9>

Restrictions and Limitations for Oracle APEX with Autonomous Database

Disabled options in Administration ServicesとしてWorkspace Isolationが含まれています。

以上が、現時点でのAutonomous Database上のAPEXによる複数ドメインのサポート状況になるようです。

元の英語の記事にはロード・バランサの設定なども含まれていますが、あるAPEXアプリケーションにアクセスできるドメインを特定するために、ロード・バランサのルールやAPEX側のコードで制御しています。

ワークスペースの分離によって実現可能な要件であれば、Autonomous Databaseで利用可能になるのを待つのも選択肢になるでしょう。

完

Yuji N. 時刻: 16:45

共有

<

ホーム

>

[ウェブ バージョンを表示](#)

自己紹介

Yuji N.

日本オラクル株式会社に勤務していて、Oracle APEXのGroundbreaker Advocateを拝命しました。こちらの記事につきましては、免責事項の参照をお願いいたします。

[詳細プロフィールを表示](#)