



Azure Blockchain Service ハンズオン ~Logic App 統合編~

日本マイクロソフト株式会社
Azure アプリケーション開発技術営業本部
南澤 拓法 (Takunori MINAMISAWA)



本資料について&各種情報

本資料のコンテンツを通して Starbucks の Blockchain 事例アーキテクチャの一部を作成することができます。 Azure Blockchain Service にはじめて触る方に特におススメのコンテンツです。

◎サンプルコード・クエリ

<https://github.com/Azure-Samples/blockchain-devkit/tree/master/integrate/data/sql>

◎事前準備

- ・有効なマイクロソフトアカウント
- ・Azure サブスクリプション

Starbucks | Bean to Barista

Challenge

スターバックスはコーヒーを取り巻く広大なサプライチェーンでコーヒー豆の動きの可視化を向上させる必要がありました

Strategy

スターバックスはBlockchainとスマートコントラクトを用いて、各サプライチェーン参加者の役割と責任を定義し、農場から飲料を注ぐまで可視化しました

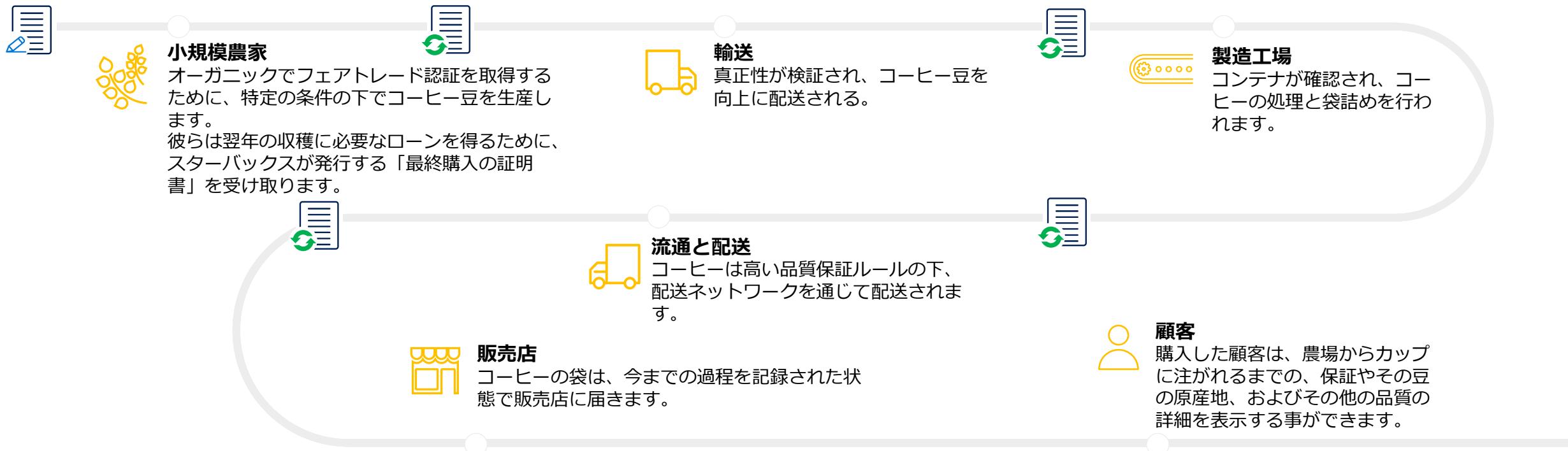
Results

- ・ 小規模農家の信用力確立向上の支援、より高い与信によるローン利用を実現、これによって生産と運用を拡大できます
- ・ 豆の流通と在庫の**コントロールを向上**
- ・ 顧客に対して、**各地域の農場より良いコーヒーを栽培する為の方策**の可視化



Over the next two years, we will look to demonstrate how technology and innovative data platforms can give coffee farmers even more financial empowerment”
– Kevin Johnson, Starbucks CEO

Using blockchain to reinforce trust from farmer to customer



Farmer
✓ Plot #839049
✓ 25 tons of cocoa beans
✓ Certifications
購入証明

Carrier
✓ Shipped 2/5/2018
✓ Organic
✓ Fair Trade

Manufacturer
✓ Delivered 2/12/2018
✓ Organic
✓ Fair Trade

Distributor
✓ Delivered 2/19/2018
✓ Organic
✓ Fair Trade

Retailer
✓ Delivered 2/26/2018
✓ Organic
✓ Fair Trade

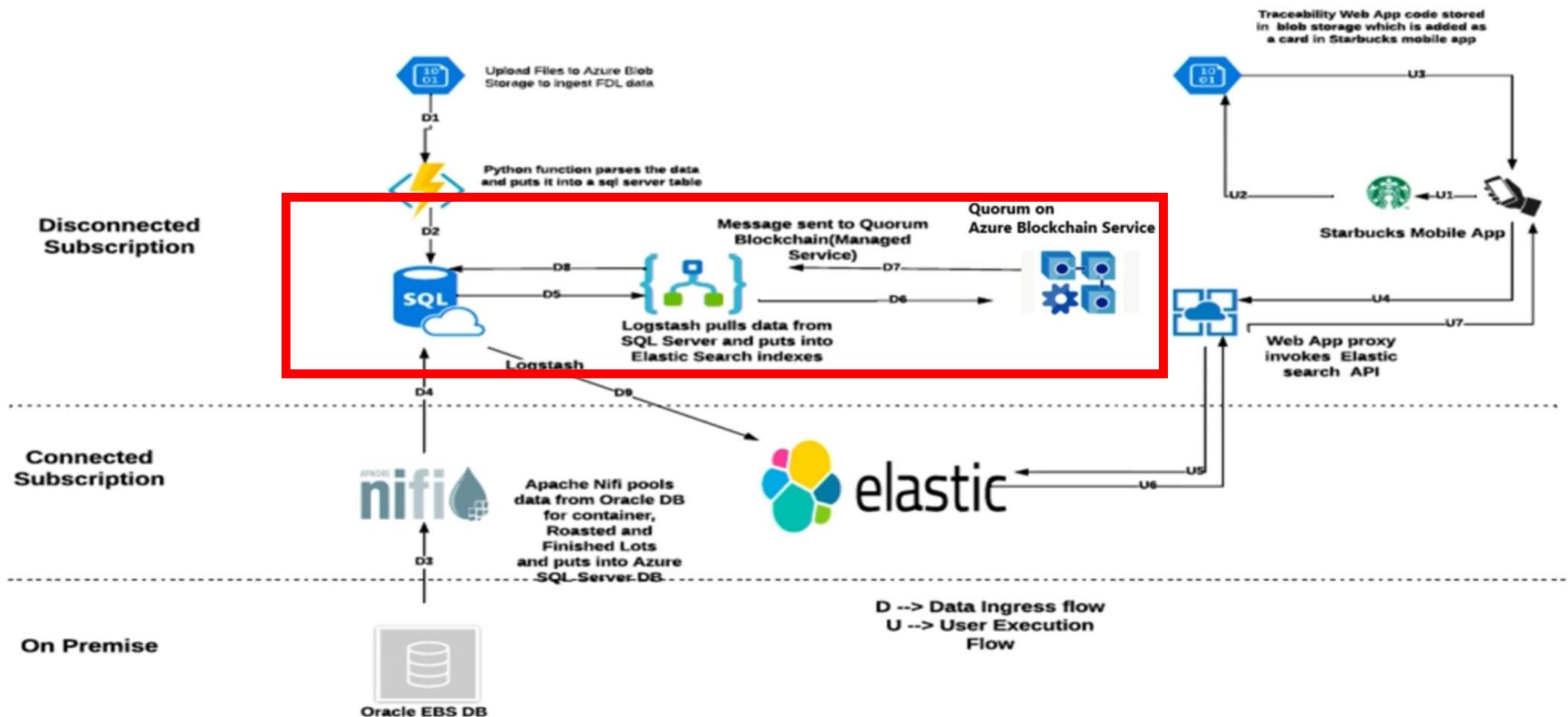
Customer
✓ Guaranteed fresh
✓ Certified organic
✓ Certified Fair Trade

品質認証

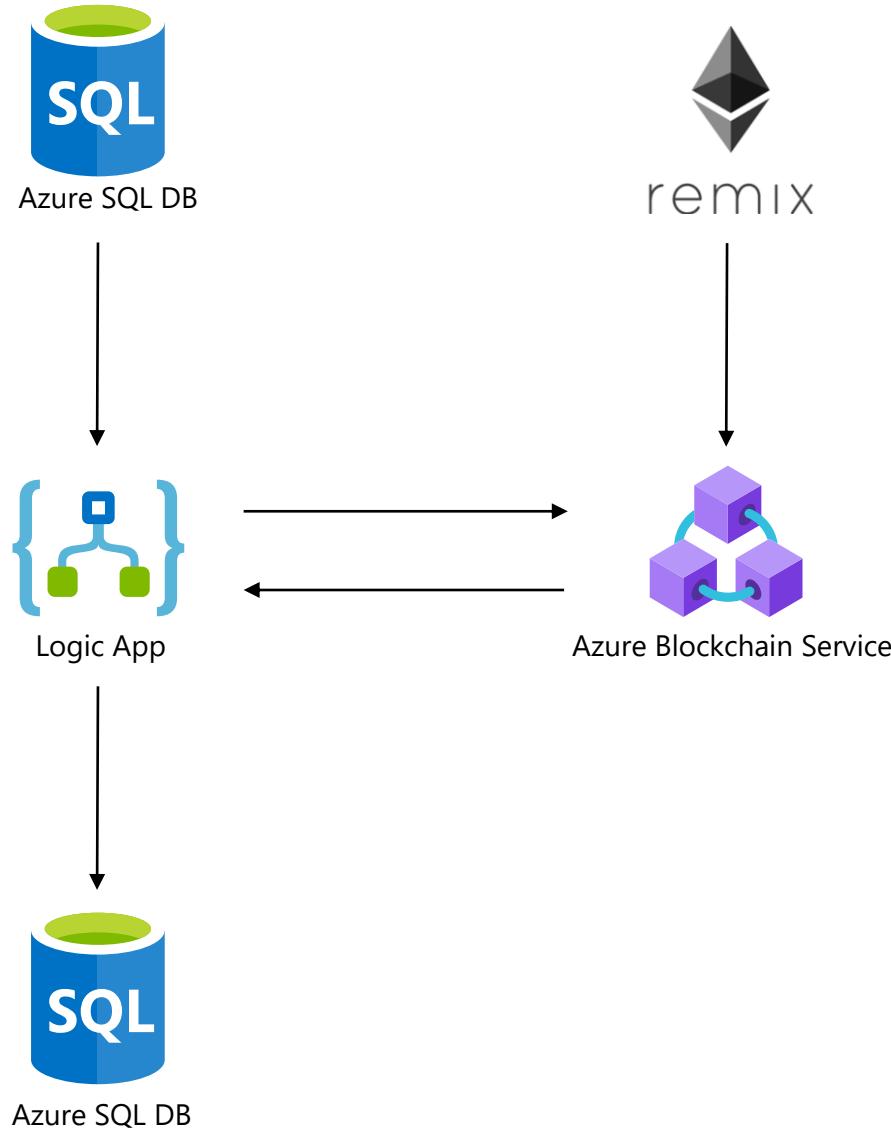
消費者からの可視化

これらの過程の記録は、様々な時点でのIoTデバイスによる製品スキャンによる状態参照、Blockchain上で状態更新記録が行われます。

TECHNICAL ARCHITECTURE | TRACEABILITY



本ハンズオンで作るもの全体像



- Remix 上で作成したスマートコントラクトを Azure Blockchain Service にデプロイします。
- Azure SQL DB にデータがインサートされたことをトリガーにスマートコントラクトを実行します。
- コントラクトアドレスを取得し、DB をアップデートします。

ステップ 1

Azure Blockchain Service の作成

Azure Portal にアクセスし、「リソースの作成」を選択してください

The screenshot shows the Microsoft Azure portal interface. At the top, there is a navigation bar with the Microsoft Azure logo, a search bar labeled 'リソース、サービス、ドキュメントの検索 (G+ /)', and various icons for account management and help. The main content area is titled 'Azure サービス' (Azure Services) and features a large 'Create resource' button with a plus sign, which is highlighted with a red box. Below this, there are sections for 'Recent resources', 'Move' (with links to Subscription, Resource Group, All resources, and Dashboard), 'Tools' (with links to Microsoft Learn, Azure Monitor, Security Center, and Cost Management), and 'Link' (with links to Technical documentation, Azure Service Expert search, Latest Azure update program, and Click Start center). On the right side, there are download links for the Azure mobile app from the App Store and Google Play.

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+ /)

☰ Microsoft Azure

リソースの作成

SQL データベース サブスクリプション リソース グループ Virtual Machines App Service ストレージ アカウント Azure Database for PostgreSQL Azure Cosmos DB その他のサービス →

最近のリソース

移動

サブスクリプション リソース グループ すべてのリソース ダッシュボード

ツール

Microsoft Learn Azure Monitor Security Center コスト管理

役に立つリンク

技術文書 Azure 移行ツール Azure サービス Azure Expert の検索 最新の Azure 更新プログラム クイック スタート センター

Download on the App Store GET IT ON Google Play

「ブロックチェーン」カテゴリから「Azure Blockchain Service」を選択してください

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+ /)

ホーム > 新規

新規

Marketplace を検索

Azure Marketplace すべて表示 おすすめ すべて表示

① ブロックチェーン

② Azure Blockchain Service (プレビュー)

作業の開始 最近作成 AI + Machine Learning 分析 ブロックチェーン Compute コンテナー データベース 開発者ツール DevOps ID 統合 モノのインターネット (IoT) メディア Mixed Reality 管理ツール ネットワーキング サービスとしてのソフトウェア (SaaS) セキュリティ ストレージ Web

Azure Blockchain Workbench (プレビュー)
Ethereum Proof-of-Authority Consortium (プレビュー)
Hyperledger Fabric Consortium (プレビュー)
Corda Single Ledger Network (プレビュー)
Quorum Single Member Blockchain Network (プレビュー)
Chain Core Developer Edition (プレビュー)
Syscoin Full Node (プレビュー)
Ethereum Studio (プレビュー)
Syscoin Price Peg Server (プレビュー)

taminami@microsoft.com MICROSOFT

必要項目を入力し、「確認および作成」をクリックしてください

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+/)

ホーム > 新規 > ブロックチェーン メンバーの作成

ブロックチェーン メンバーの作成
プレビュー

Blockchain Service の詳細を表示します

プロジェクトの詳細

デプロイされているリソースとコストを管理するサブスクリプションを選択します。フォルダーのようなリソース グループを使用して、すべてのリソースを整理し、管理します。

サブスクリプション * ① Microsoft Azure 社内従量課金プラン (4521f350-fcd3-4a43-9e23-33c294...)

リソース グループ * ① (新規) DX-Handson-RG

新規作成

領域 * ① 東日本

契約されている Azure サブスクリプション
任意の名前
「東日本」をお選びください

ブロックチェーンの詳細

参加を招待されているプロトコルとコンソーシアムを選択するか、独自のコンソーシアムを作成して開始します。後で、参加するように他のメンバーを招待することができます。

プロトコル * ① Quorum

コンソーシアム * ① (新規) myhandsonconso

新規作成

任意の名前

メンバーの詳細

名前 * ① taminahol

メンバー アカウントのパスワード * ①

価格 * ① 基本 - 1 個の仮想コア
1 個の検証ノード、1 個のトランザクション ノード
推定コスト 1,031.53 米国ドル/月
変更

任意の名前
任意のパスワード
「Basic Plan」をお選びください

トランザクション ノードの詳細

ノードのパスワード * ①

任意のパスワード

確認および作成 戻る 次へ: タグ

「作成」をクリックし、リソースを作成してください

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+/)

ホーム > 新規 > ブロックチェーン メンバーの作成

ブロックチェーン メンバーの作成

レビュー

✓ 検証に成功しました

基本 タグ 確認および作成

概要

Azure Blockchain Service (プレビュー)
Microsoft 提供
利用規約 | プライバシー ポリシー

価格 * ⓘ
基本 - 1 個の仮想コア
1 個の検証ノード、1 個のトランザクション ノード
推定コスト 1,031.53 米国ドル/月
[変更](#)

利用規約

[作成] をクリックすることで、お客様は (a) 上記の Marketplace のオファリングに関連する法律条項とプライバシーに関する声明に同意し、(b) Microsoft より、そのオファリングに関連する料金が、現在の支払い方法に対して Azure サブスクリプションと同じ請求頻度で請求されることを認め、かつ、(c) Microsoft がお客様の連絡先情報、使用量情報、取引に関する情報を、サポート、請求、その他の取引上のアクティビティを目的として、オファリングのプロバイダーと共有する可能性があることに同意するものとします。Microsoft は、サード パーティのオファリングに対する権利は提供しません。その他の詳細については、[Azure Marketplace 使用条件](#)をご覧ください。

基本

サブスクリプション	Microsoft Azure 社内従量課金プラン
リソース グループ	DX-Handson-RG
領域	東日本
プロトコル	Quorum
コンソーシアム	myhandsonconso
名前	taminahol

NETWORK

Validator nodes 1

[作成](#) 前の: タグ 次へ

※留意点1 (既知のバグ/報告済み)

The screenshot shows the Microsoft Azure portal interface. The top navigation bar includes the Microsoft Azure logo, a search bar, and a user profile for taminami@microsoft.com. The main content area displays a deployment status for a blockchain member payload. A red error message at the top right states: "Invalid blockchain member payload: invalid sku tier '基本' when sku name is 'B0' 詳細については、ここをクリックしてください →". Below this, a summary section shows the deployment name as Microsoft.Blockchain_80ce929288fd4ac5a4e4f02f9afed1fb, start time as 2020/1/20 12:02:58, and resource group as DX-Handson-RG. A detailed table below lists resources, their types, and deployment status, all showing '結果がありません' (No results). To the right, there are links to the Security Center and Microsoft Experts.

デプロイに失敗しました

デプロイ名: Microsoft.Blockchain_80ce929288fd4ac5a4e4f02f9afed1fb 開始時刻: 2020/1/20 12:02:58
サブスクリプション: Microsoft Azure 社内従量課金プラン (4521f350-fcd... 相関 ID: 3437c6b8-5972-4a3d-a697-5c9319e4c8d8
リソース グループ: DX-Handson-RG

▽ 展開の詳細 (ダウンロード)

リソース	種類	状態	操作の詳細
結果がありません。			

▽ 次の手順

● デプロイできませんでした 12:03
リソース グループ 'DX-Handson-RG' へのデプロイメントが失敗しました。
役立つ可能性がある基本 API から得られた追加の詳細情報: 少なくとも 1 つのリソースのデプロイ操作に失敗しました。詳細については、デプロイ操作の一覧を表示してください。使用方法の詳細については、<https://aka.ms/DeployOperations> を参照してください。

セキュリティセンター
アプリおよびインフラストラクチャをセキュリティで保護します
[Azure Security Center に移動 >](#)

無料の Microsoft チュートリアル
[今すぐ学習を開始する >](#)

エキスパートとの共同作業
Azure のエキスパートは、Azure での資産の管理を支援し、サポートの最前線となるサービス プロバイダーパートナーです。
[Azure Expert の検索 >](#)

作成時に SKU に関してのエラーが起こる場合があります。
この場合、Azure ポータルの規定の言語を「英語」に変更して再度作成してください。

「メンバー アカウント」をコピーし、メモ帳等に貼り付けてください

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+/)

taminahol taminami@microsoft.com MICROSOFT

ホーム > taminahol

taminahol Azure Blockchain Service - ブリューバー ディレクトリ: Microsoft

検索 (Ctrl+ /) 削除

リソース グループ : DX-Handson-RG
状態 : 使用可能
場所 : 東日本
サブスクリプション : Microsoft Azure 社内従量課金プラン
サブスクリプション ID : 4521f350-fcd3-4a43-9e23-33c294f092f6

メンバー名 : taminahol
プロトコル : Quorum
価格レベル : 基本 (1 個の仮想コア, 2 ノード)
コンソーシアム : myhandsonconso
RootContract アドレス : 0xb255f55e8d600f09ebc1035dd2118acec1018912
メンバー アカウント : **0x815e9bada4cabfc8c07c67ecce0d99b73fdf1262**

概要
アクティビティ ログ
アクセス制御 (IAM)
タグ
設定
プロパティ
ロック
テンプレート の エクスポート
価格レベル
ブロックチェーン
トランザクション ノード
Blockchain Data Manager

コンソーシアム
コンソーシアムからメンバーを招待または削除します。
起動 >

ご意見をお聞かせください
Azure Blockchain Service に関するフィードバックを提供します。
検索 ▾

Blockchain 開発キット
作業を開始するためのサンプル コードとアクセラレータ。
検索 ▾

次に指定する直近の期間のデータを表示する: 1 時間 6 時間 12 時間 1 日 7 日 30 日

ブロック

トランザクション

12
10
8
6
4
2
0
11:30 11:45 12時 12:15
validator-node-0 taminahol 16
transaction-node taminahol 14

9
8
7
6
5
4
3
2
1
0
11:30 11:45 12時 12:15
transaction-node taminahol 8
validator-node-0 taminahol 8

作成した Blockchain Service を開き、「トランザクションノード」からノードを選択してください

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+/)

ホーム > taminahol - トランザクションノード

taminahol - トランザクションノード Azure Blockchain Service - ブリューバー ディレクトリ: Microsoft

検索 (Ctrl+ /) 追加 最新の情報に更新 削除

概要 アクティビティ ログ アクセス制御 (IAM) タグ

設定 プロパティ ロック テンプレートのエクスポート 価格レベル

ブロックチェーン トランザクション ノード Blockchain Data Manager

監視 警告 メトリック 診断設定 ログ

サポート + トラブルシューティング 新しいサポート リクエスト

名前 DNS 公開キー 状態

taminahol (既定のノード) taminahol.blockchain.azure.com L8jVq5ekQLEvp18rg/KuipieP220vMS/B5oBQpRC... 健全

「接続文字列」タブから、HTTPS (アクセスキー 1) をコピーし、メモ帳等に貼り付けてください

The screenshot shows the Microsoft Azure portal interface. The user is in the 'taminahol - ブロックチェーンノード' blade, specifically the 'taminahol - 接続文字列' tab. On the left, a sidebar lists several connection string types: '概要', 'アクセス制御 (IAM)', '設定' (with '基本認証', 'アクセスキー', and 'ファイアウォール規則'), and '接続文字列'. The '接続文字列' item is highlighted with a red box. The main content area displays five connection string entries:

- HTTPS (基本認証)**: https://taminahol:<password>@taminahol.blockchain.azure.com:3200
- HTTPS (アクセスキー 1)**: https://taminahol.blockchain.azure.com:3200/CUaC7AjxReraxg1WkS8y5SHI (This entry is also highlighted with a red box)
- HTTPS (アクセスキー 2)**: https://taminahol.blockchain.azure.com:3200/3-A9iRI8Qk9rqKDjzYeGp5Sk
- WebSocket (アクセスキー 1)**: wss://taminahol.blockchain.azure.com:3300/CUaC7AjxReraxg1WkS8y5SHI
- WebSocket (アクセスキー 2)**: wss://taminahol.blockchain.azure.com:3300/3-A9iRI8Qk9rqKDjzYeGp5Sk

ステップ 2

Azure SQL DB の作成

「リソースの作成」を選択してください

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+/)

☰ Microsoft Azure taminami@microsoft.com MICROSOFT

Azure サービス

リソースの作成 (Red Box)

SQL データベース サブスクリプション リソース グループ Virtual Machines App Service ストレージ アカウント Azure Database for PostgreSQL Azure Cosmos DB その他のサービス →

最近のリソース

移動

サブスクリプション リソース グループ すべてのリソース ダッシュボード

ツール

Microsoft Learn Azure Monitor Security Center コスト管理

Microsoft Learn: Microsoft が提供する無料のオンライントレーニングで Azure について確認する

Azure Monitor: アプリとインフラストラクチャを監視します

Security Center: アプリおよびインフラストラクチャをセキュリティで保護します

コスト管理: 無料でクラウド支出を分析して最適化する

役に立つリンク

技術文書 Azure サービス 最新の Azure 更新プログラム クイック スタート センター

Azure 移行ツール Azure Expert の検索

Azure mobile app

Download on the App Store GET IT ON Google Play

「SQL Database」を選択してください

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+/)

ホーム > 新規

新規

Marketplace を検索

Azure Marketplace すべて表示 人気順

作業の開始 最近作成 AI + Machine Learning 分析 ブロックチェーン Compute コンテナー データベース 開発者ツール DevOps ID 統合 モノのインターネット (IoT) メディア Mixed Reality 管理ツール ネットワーキング サービスとしてのソフトウェア (SaaS) セキュリティ ストレージ Web

Windows Server 2016 Datacenter クイックスタート チュートリアル

Ubuntu Server 18.04 LTS 詳細情報

Web アプリ クイックスタート チュートリアル

SQL Database クイックスタート チュートリアル

関数アプリ クイックスタート チュートリアル

Azure Cosmos DB クイックスタート チュートリアル

Kubernetes Service クイックスタート チュートリアル

DevOps Project クイックスタート チュートリアル

ストレージ アカウント - Blob、File、Table、Queue クイックスタート チュートリアル

最近作成したアイテムを表示する

taminami@microsoft.com MICROSOFT

必要項目を入力し、「確認および作成」をクリックしてください

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+/-)

ホーム > 新規 > SQL データベースの作成

SQL データベースの作成 Microsoft

基本 ネットワーク 追加設定 タグ 確認および作成

希望する構成で SQL データベースを作成します。[基本] タブをすべて入力し、[確認と作成] に移動して、スマートな既定値でプロビジョニングするか、各タブに移動してカスタマイズします。[詳細情報](#)

プロジェクトの詳細

デプロイされているリソースとコストを管理するサブスクリプションを選択します。フォルダーのようなリソース グループを使用して、すべてのリソースを整理し、管理します。

サブスクリプション * ① Microsoft Azure 社内従量課金プラン (4521f350-fcd3-4a43-9e23-33c294...)

リソース グループ * ① DX-Handson-RG

新規作成

契約されている Azure サブスクリプション
先程作成したリソースグループ

データベースの詳細

論理サーバーの選択や、コンピューティング リソースとストレージ リソースの構成など、このデータベースに必要な設定を入力します

データベース名 * tamina-dxhol-db

サーバー * ① (新規) tamina-dxhol-server ((アジア太平洋) 東南アジア)

新規作成

任意の名前
任意の名前で新規作成

SQL エラスティック プールを使用しますか? * ① はい いいえ

コンピューティングとストレージ * ① Basic 2 GB ストレージ データベースの構成

「Basic」を選択してください

確認および作成 次: ネットワーク >

「作成」をクリックしてください

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+/)

ホーム > 新規 > SQL データベースの作成

SQL データベースの作成

基本 ネットワーク 追加設定 タグ 確認および作成

製品の詳細

SQL データベース Microsoft 提供
利用規約 | プライバシー ポリシー

月ごとの推定コスト
558.99 JPY
[価格の詳細を表示](#)

使用条件

[作成] をクリックすることで、お客様は (a) 上記の Marketplace のオファリングに関連する法律条項とプライバシーに関する声明に同意し、(b) Microsoft より、そのオファリングに関連する料金が、現在の支払い方法に対して Azure サブスクリプションと同じ請求頻度で請求されることを認め、かつ、(c) Microsoft がお客様の連絡先情報、使用量情報、取引に関する情報を、サポート、請求、その他の取引上のアクティビティを目的として、オファリングのプロバイダーと共有する可能性があることに同意するものとします。Microsoft は、サード パーティのオファリングに対する権利は提供しません。その他の詳細については、以下を参照してください: [Azure Marketplace の使用条件](#)。

基本

サブスクリプション	Microsoft Azure 社内従量課金プラン
リソース グループ	DX-Handson-RG
リージョン	(アジア太平洋) 東南アジア
データベース名	tamina-dxhol-db
サーバー	(新規) tamina-dxhol-server
コンピューティングとストレージ	Basic: 2 GB ストレージ

ネットワーク

Azure サービスおよびリソースにこのサーバーへのア いいえ
クセスを許可する

プライベート エンドポイント (プレビュー)
なし

追加設定

既存のデータを使用します	Blank
照合順序	SQL_Latin1_General_CI_AS
Advanced Data Security	後で

作成 < 前へ Automation のテンプレートをダウンロードする

DB名とサーバー名をコピーし、メモ帳等に貼り付けてください

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+ /)

ホーム > tamina-dxhol-db (tamina-dxhol-server/tamina-dxhol-db)

tamina-dxhol-db tamina-dxhol-server/tamina-dxhol-db Microsoft

検索 (Ctrl+ /) コピー 復元 エクスポート サーバーファイアウォールの設定 削除 次に接続する... フィードバック

概要 リソース グループ (変更) : DX-Handson-RG サーバー名 : **tamina-dxhol-server.database.windows.net**
状態 : オンライン エラスティック プール : エラスティック プールはありません
場所 : 東南アジア 接続文字列 : データベース接続文字列の表示
サブスクリプション (変更) : Microsoft Azure 社内従量課金プラン 価格レベル : Basic
サブスクリプション ID : 4521f350-fcd3-4a43-9e23-33c294f092f6 最初の復元ポイント : 2020-01-20 03:48 UTC
タグ (変更) : タグを追加するにはここをクリック

次に指定する直近の期間のデータを表示する: 1時間 24時間 7日 集約型: 最... ▾

使用率の計算

DTU percentage (最大値)
tamina-dxhol-server/tamina-dxhol-db
0 %

データベースのデータストレージ ①
使用領域: 20 MB
割り当て済みの領域: 32 MB

通知 (0) データベース機能 (6)
すべて セキュリティ (4) パフォーマンス (1) 回復 (1)

Transparent Data Encryption Advanced Data Security

検索 (Ctrl+ /) コピー 復元 エクスポート サーバーファイアウォールの設定 削除 次に接続する... フィードバック

概要 リソース グループ (変更) : DX-Handson-RG サーバー名 : **tamina-dxhol-server.database.windows.net**
状態 : オンライン エラスティック プール : エラスティック プールはありません
場所 : 東南アジア 接続文字列 : データベース接続文字列の表示
サブスクリプション (変更) : Microsoft Azure 社内従量課金プラン 価格レベル : Basic
サブスクリプション ID : 4521f350-fcd3-4a43-9e23-33c294f092f6 最初の復元ポイント : 2020-01-20 03:48 UTC
タグ (変更) : タグを追加するにはここをクリック

次に指定する直近の期間のデータを表示する: 1時間 24時間 7日 集約型: 最... ▾

使用率の計算

DTU percentage (最大値)
tamina-dxhol-server/tamina-dxhol-db
0 %

データベースのデータストレージ ①
使用領域: 20 MB
割り当て済みの領域: 32 MB

通知 (0) データベース機能 (6)
すべて セキュリティ (4) パフォーマンス (1) 回復 (1)

Transparent Data Encryption Advanced Data Security

作成した SQL DB を開き、「クエリ エディター」にログインしてください

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+)

ホーム > tamina-dxhol-db (tamina-dxhol-server/tamina-dxhol-db) - クエリ エディター (プレビュー)

tamina-dxhol-db (tamina-dxhol-server/tamina-dxhol-db) - クエリ エディター (プレビュー) ディレクトリ: Microsoft

検索 (Ctrl+ /) ログイン 新しいクエリ クエリを開く フィードバック

概要 アクティビティ ログ タグ 問題の診断と解決 クイック スタート クエリ エディター (プレビュー) (選択)

Power Platform Power BI (プレビュー) PowerApps (プレビュー) Flow (プレビュー)

設定 設定 geo レプリケーション 接続文字列 別のデータベースに同期 Azure Search の追加 プロパティ ロック テンプレートのエクスポート

統合 Stream Analytics (プレビュー)

セキュリティ Advanced Data Security 監査 動的データ マスキング

SQL Database クエリ エディターへようこそ

SQL Server 認証 Active Directory 認証

ログイン * taminami として続行
または
パスワード * OK

※留意点2

The screenshot shows the Microsoft Azure portal interface for managing a SQL database. The left sidebar contains navigation links for Overview, Activity Log, Tags, Problem Diagnosis, Quick Start, and the currently selected 'Query Editor (Preview)'. The main content area is titled 'tamina-dxhol-db (tamina-dxhol-server/tamina-dxhol-db) - クエリ エディター (プレビュー)'. At the top, there's a search bar and a toolbar with icons for Login, New Query, Open Query, and Feedback. The central part of the screen displays the 'SQL Database クエリ エディターへようこそ' (Welcome to the SQL Database Query Editor) page. It features two login options: 'SQL Server 認証' (Authentication) and 'Active Directory 認証' (Authentication). Under 'SQL Server 認証', fields for 'Login' (taminami) and 'Password' (redacted) are filled in. A red error message at the bottom left states: 'Cannot open server 'tamina-dxhol-server' requested by the login. Client with IP address '167.220.233.110' is not allowed to access the server. To enable access, use the Windows Azure Management Portal or run sp_set_firewall_rule on the master database to create a firewall rule for this IP address or address range. It may take up to five minutes for this change to take effect.' A blue 'OK' button is located at the bottom right of the error message. The top right corner of the screen shows the user's email (taminami@microsoft.com) and Microsoft logo.

ログインできず、上記のようなエラーメッセージが出る場合には SQL Server の Firewall の設定で、表示されている IP アドレスを許可する必要があります。

ログイン後クエリを実行し、テーブルを作成してください (※要GitHub参照)

The screenshot shows the Microsoft Azure portal interface for managing a database named 'tamina-dxhol-db'. The left sidebar contains navigation links for various Azure services like Power BI, Flow, and Stream Analytics. The main content area is titled 'tamina-dxhol-db (tamina-dxhol-server/tamina-dxhol-db) - クエリ エディター (プレビュー)'. A search bar at the top right allows searching for resources, services, and documents. The central workspace displays a query editor window with the following details:

- クエリ**: クエリ エディター (プレビュー)
- タブ**: クエリ 1
- 操作**: 実行 (highlighted with a red box), クエリのキャンセル, クエリの保存, データを .json としてエクスポート, データを .csv としてエクスポート, データを .xml としてエクスポート
- クエリ**:

```
1 CREATE TABLE contractrecords (id int NOT NULL IDENTITY(1,1) PRIMARY KEY, contractID VARCHAR(50), supplyChainOwner VARCHAR(50), supplyC
```
- 結果**: クエリが成功しました: Affected rows: 0
- メッセージ**: クエリが成功しました | 0秒

ステップ 3

Logic Apps の作成

「リソースの作成」を選択してください

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+/)

☰ Microsoft Azure taminami@microsoft.com MICROSOFT

Azure サービス

リソースの作成 (Red Box)

SQL データベース サブスクリプション リソース グループ Virtual Machines App Service ストレージ アカウント Azure Database for PostgreSQL Azure Cosmos DB その他のサービス →

最近のリソース

移動

サブスクリプション リソース グループ すべてのリソース ダッシュボード

ツール

Microsoft Learn Azure Monitor Security Center コスト管理

Microsoft Learn: Microsoft が提供する無料のオンライントレーニングで Azure について確認する

Azure Monitor: アプリとインフラストラクチャを監視します

Security Center: アプリおよびインフラストラクチャをセキュリティで保護します

コスト管理: 無料でクラウド支出を分析して最適化する

役に立つリンク

技術文書 Azure サービス 最新の Azure 更新プログラム クイック スタート センター

Azure 移行ツール Azure Expert の検索

Azure mobile app

Download on the App Store GET IT ON Google Play

「統合」カテゴリから「Logic App」を選択してください

The screenshot shows the Microsoft Azure 'New' blade. On the left, there's a sidebar with various service categories. The 'Integration' category (marked with a red circle and number 1) is currently selected and highlighted in red. In the main content area, under the 'Azure Marketplace' heading, the 'Logic App' service is listed (marked with a red circle and number 2). It has a blue icon with a person symbol and the text 'Logic App クイックスタート チュートリアル'. Below it, other services are listed: API Management, Service Bus, 統合アカウント, 統合サービス環境, Logic Apps カスタム コネクタ, Data Factory, Data Catalog, Azure Stack Edge / Data Box Gateway, Dell Boomi Atom (Windows) (Preview), and Web.

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+/)

ホーム > 新規

新規

Marketplace を検索

Azure Marketplace すべて表示 おすすめ すべて表示

作業の開始

最近作成

AI + Machine Learning

分析

ブロックチェーン

Compute

コンテナー

データベース

開発者ツール

DevOps

ID

統合

モノのインターネット (IoT)

メディア

Mixed Reality

管理ツール

ネットワーキング

サービスとしてのソフトウェア (SaaS)

セキュリティ

ストレージ

Web

①

②

Logic App
クイックスタート チュートリアル

API 管理
クイックスタート チュートリアル

Service Bus
クイックスタート チュートリアル

統合アカウント
クイックスタート チュートリアル

統合サービス環境
詳細情報

Logic Apps カスタム コネクタ
詳細情報

Data Factory
クイックスタート チュートリアル

Data Catalog
詳細情報

Azure Stack Edge / Data Box Gateway
詳細情報

Dell Boomi Atom (Windows) (Preview)
→ 詳細情報

「作成」をクリックしてください

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+) プレビュー バグの報告 ダッシュボード > 新規 > Logic App taminami@microsoft.... MICROSOFT

リソースの作成 ホーム ダッシュボード すべてのサービス お気に入り サブスクリプション リソース グループ すべてのリソース App Service SQL データベース Azure Cosmos DB Virtual Machines ロード バランサー モニター ストレージ アカウント Azure Active Directory 仮想ネットワーク Advisor セキュリティ センター コストの管理と請求 ヘルプとサポート

Logic App Microsoft ダッシュボード > 新規 > Logic App Logic App Microsoft 後で使用するために保存 作成

Logic Apps を使用すると、開発者はトリガーと一緒に手順によって意図を明確にするワークフローを設計できます。各手順では、認証とベスト プラクティス (永続的実行など) に安全に対処しながら、App Service API App を呼び出します。

使いやすい設計ツール - Logic Apps は、ブラウザー内でエンド ツー エンドで設計できます。まず、トリガーから始めます。単純なスケジュールから、自分の会社についてのツイートがあった場合というような条件までさまざまです。次に、コネクタの豊富なギャラリーを使用して、いくつでもアクションを調整できます。

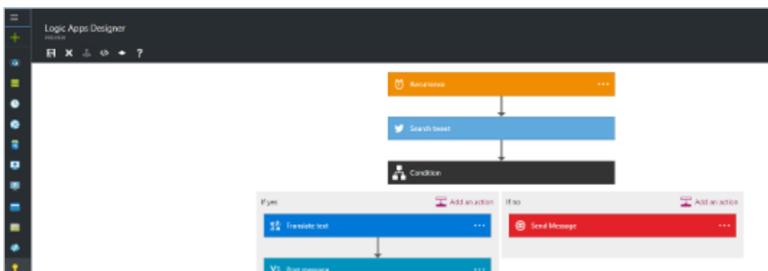
簡単に SaaS を作成 - 簡単に記述できる複合タスクでも、コードに実装するのは難しいものです。Logic Apps を使用すると、異種のシステムを簡単に接続できます。たとえば、Facebook または Twitter アカウントでの活動に基づいて CRM でタスクを作成する場合や、クラウド マーケティング ソリューションをオンプレミスの課金システムに接続する場合があります。ロジック アプリは、このような問題に対する解決策を提供するのに、最も手軽で信頼性の高い方法です。

組み込みの拡張性 - 必要なコネクタが見つかりませんか? Logic Apps は、App Service スイートに含まれており、API アプリと連動するように設計されています。そのため、コネクタとして使用する独自の API アプリを簡単に作成できます。自分専用に新しいアプリを作成することも、Marketplace で公開し、収益を得ることもできます。

実際の統合の力 - 簡単に始めて、必要に応じて拡張することができます。Logic Apps では、統合の専門家が自分たちの必要とするソリューションを構築できるようにする Microsoft の業界トップの統合ソリューションである BizTalk の力を簡単に活用できます。

役に立つリンク
ドキュメント
ソリューションの概要
価格の詳細

Logic Apps Designer



必要項目を入力し、「作成」をクリックしてください

The screenshot shows the Microsoft Azure Logic App creation interface. The top navigation bar includes the Microsoft Azure logo, a search bar, and user information (taminami@microsoft.com). The main title is "Logic App" with a subtitle "作成". The form fields are as follows:

- 名前 ***: taminahol-la (highlighted with a red arrow)
- サブスクリプション ***: Microsoft Azure 社内従量課金プラン (452) (highlighted with a red arrow)
- リソース グループ ***: DX-Handson-RG (highlighted with a red arrow)
- 場所 ***: 東日本 (highlighted with a red arrow)
- Log Analytics**: A toggle switch set to "Off" (highlighted with a red arrow)

A note at the bottom left says: "作成後に、ロジック アプリにトリガーとアクションを追加できます。" (After creation, you can add triggers and actions to the logic app.)

At the bottom right, there is a blue "作成" (Create) button, which is highlighted with a red border.

ステップ 4

スマートコントラクトのデプロイ

「<https://remix.ethereum.org>」にアクセスし、sol ファイルを作成してください

The screenshot shows the Remix Ethereum IDE interface. On the left, there's a sidebar with icons for File Explorers, a browser tab containing 'ballot.sol' and 'ballot_test.sol', and other development tools. The main area has tabs for 'Solidity', 'Vyper', and 'Workshops'. A central modal window titled 'Create new file' is open, with the 'File Name (e.g Untitled.sol)' input field containing 'DXHOL-contract.sol' and the 'OK' button highlighted with a red box. Below the modal, the 'File' menu is visible with options like 'New File', 'Open Files', and 'Import From:' with links to Gist, GitHub, Swarm, Ipfs, https, and Resolver-engine. To the right, there's a 'Featured Plugins' section with 'Pipeline' and 'Debugger' listed, and a 'See all Plugins' button. Another section labeled 'Resources' includes links to Documentation, Gitter channel, Medium Posts, and Tutorials. At the bottom, there's a footer with network status (0 transactions), a search bar, and a note about script usage.

FILE EXPLORERS

browser

ballot.sol
ballot_test.sol

Home

Create new file

File Name (e.g Untitled.sol)

DXHOL-contract.sol

OK Cancel

Environments

Solidity Vyper Workshops

File

New File
Open Files
Connect to Localhost
Import From:
Gist GitHub Swarm Ipfs https Resolver-engine

Featured Plugins

Pipeline
Debugger
See all Plugins

Resources

Documentation
Gitter channel
Medium Posts
Tutorials

script.
• Use exports/.register(key, obj)/.remove(key)/.clear() to register and reuse object across script executions.

「Plugin manager」をクリックし、「Deploy & run transactions」を Active にしてください

The screenshot shows the Remix IDE's Plugin Manager interface. On the left, there is a sidebar titled "PLUGIN MANAGER" with a gear icon. The main area displays a list of inactive modules. One module, "Deploy & run transactions", is highlighted with a red box and a red border around its "Activate" button. This module is described as "Execute And Save Transactions". Other visible modules include "3Box Spaces" (BETA), "Debug Tools for Remix plugins" (ALPHA), "Debugger", "Ethdoc - Documentation Generator", "Etherscan - Contract verification", and "ethPM". Each module has an "Activate" button next to it. The top right of the screen shows tabs for "Home" and "DXHOL-contract.sol". The bottom right contains a search bar and some network-related settings.

- 3Box Spaces BETA Activate
- Debug Tools for Remix plugins ALPHA Activate
- Debugger Activate
- Deploy & run transactions Activate
 - Execute And Save Transactions
- Ethdoc - Documentation Generator Activate
- Etherscan - Contract verification Activate
- ethPM Activate

Search with transaction hash or address

script.
• Use exports/.register(key, obj)/.remove(key)/.clear() to register and reuse object across script executions.

同様に「Solidity compiler」を Active にしてください

The screenshot shows the Remix IDE's Plugin Manager interface. On the left, there is a sidebar with various icons and a search bar. The main area displays a list of available plugins:

- Deploy And Interact With Private Contracts On A Quorum Network (Activate)
- Remix Workshops (BETA) (Activate)
- remixd (Activate)
- Solidity compiler (Activate) - This item is highlighted with a red rectangular box.
- Solidity static analysis (Activate)
- Solidity unit testing (Activate)
- Vyper (Activate)
- ZoKrates (ALPHA) (Activate)

The Solidity compiler plugin is described as "Compile Solidity Contracts". The entire sidebar area is enclosed in a light gray border. At the bottom of the screen, there is a footer bar with a settings gear icon and a "Settings" button.

「Web3 Provider」を選択し、先ほどコピーしたアクセスキーを入力してください

The screenshot shows the Remix IDE interface with the following elements:

- Left Sidebar:** Contains icons for deploying contracts, running transactions, and other developer tools.
- Top Bar:** Shows tabs for "Home" and "DXHOL-contract.sol".
- Deploy & Run Transactions Panel:** Includes fields for "Environment" (set to "Web3 Provider"), "Account" (set to "0xCA3...a733c (100 ETH)", with a red box around it), "Gas limit" (3000000), "Value" (0 wei), and a dropdown menu.
- Central Area:** Displays the contract code "DXHOL-contract.sol".
- External node request Dialog:** A modal window titled "External node request" with the following content:
 - A note: "Note: If you are using Geth, please configure it to allow requests from Remix: geth --rpc --rpccorsdomain https://remix.ethereum.org"
 - A "Web3 Provider Endpoint" input field containing the URL "https://taminahol.blockchain.azure.com:3200/CUaC7AjxReraxg1W", which is also highlighted with a red box.
 - Buttons: "OK" (highlighted with a red box) and "Cancel".
- Bottom Bar:** Includes a "listen on network" checkbox, a search bar, and a message about script usage.

スマートコントラクトを記述してください (※要GitHub参照)

The screenshot shows the Remix Solidity Compiler interface. On the left, there's a sidebar with compiler configuration options like Auto compile, Enable optimization, and Hide warnings. A prominent orange button at the bottom left says "No Contract Compiled Yet". The main area is a code editor titled "DXHOL-contract.sol" with the following Solidity code:

```
pragma solidity ^0.4.25;

contract BasicProvenance
{
    //Set of states
    enum StateType { Created, InTransit, Completed }

    //List of properties
    StateType public state;
    address public InitiatingCounterparty;
    address public Counterparty;
    address public PreviousCounterparty;
    address public SupplyChainOwner;
    address public SupplyChainObserver;
    int public ContractReference;

    event TransferResp(address who);
    event ContractComplete(int reference);

    constructor (address supplyChainOwner, address supplyChainObserver, int contractNumber) public
    {
        InitiatingCounterparty = msg.sender;
        Counterparty = InitiatingCounterparty;
        SupplyChainOwner = supplyChainOwner;
        SupplyChainObserver = supplyChainObserver;
        State = StateType.Created;
        ContractReference = contractNumber;
    }

    function TransferResponsibility(address newCounterparty) public
    {
        if (Counterparty != msg.sender || State == StateType.Completed)
        {
            revert();
        }

        if (State == StateType.Created)
        {
            State = StateType.InTransit;
        }

        PreviousCounterparty = Counterparty;
        Counterparty = newCounterparty;

        emit TransferResp(msg.sender);
    }

    function Complete() public
    {
        if (SupplyChainOwner != msg.sender || State == StateType.Completed)
        {
            revert();
        }
    }
}
```

At the bottom, there are network status indicators (0 listeners, 0 pending transactions), a search bar ("Search with transaction hash or address"), and a note about script usage.

コンパイラの設定を確認後、コンパイルを実行してください

※バージョンに注意

The screenshot shows the Remix Solidity Compiler interface. On the left, there's a sidebar with various icons and settings. A red arrow points from the text "※バージョンに注意" (Please pay attention to the version) to the "Compiler" dropdown menu, which is set to "0.4.25+commit.59dbf8f". A red box highlights the "Compile DXHOL-contract.sol" button. Another red box highlights the "Compiler Configuration" section, which includes checkboxes for "Auto compile", "Enable optimization", and "Hide warnings". Below this, a yellow box displays the message "No Contract Compiled Yet". On the right, the code editor shows the Solidity contract "DXHOL-contract.sol" with its source code. The code defines a state enum, properties for counterparty addresses and supply chain owners, and three functions: constructor, TransferResponsibility, and Complete. At the bottom, there are network status indicators and a search bar.

```
pragma solidity ^0.4.25;

contract BasicProvenance
{
    //Set of states
    enum StateType { Created, InTransit, Completed }

    //List of properties
    StateType public state;
    address public InitiatingCounterparty;
    address public Counterparty;
    address public PreviousCounterparty;
    address public SupplyChainOwner;
    address public SupplyChainObserver;
    int public ContractReference;

    event TransferResp(address who);
    event ContractComplete(int reference);

    constructor (address supplyChainOwner, address supplyChainObserver, int contractNumber) public
    {
        InitiatingCounterparty = msg.sender;
        Counterparty = InitiatingCounterparty;
        SupplyChainOwner = supplyChainOwner;
        SupplyChainObserver = supplyChainObserver;
        State = StateType.Created;
        ContractReference = contractNumber;
    }

    function TransferResponsibility(address newCounterparty) public
    {
        if (Counterparty != msg.sender || State == StateType.Completed)
        {
            revert();
        }

        if (State == StateType.Created)
        {
            State = StateType.InTransit;
        }

        PreviousCounterparty = Counterparty;
        Counterparty = newCounterparty;

        emit TransferResp(msg.sender);
    }

    function Complete() public
    {
        if (SupplyChainOwner != msg.sender || State == StateType.Completed)
        {
            revert();
        }
    }
}
```

正常にコンパイルが実行されたことをご確認ください

The screenshot shows the Remix Solidity Compiler interface. On the left, the 'SOLIDITY COMPILER' sidebar includes settings for the compiler version (0.4.25+commit.59dbf8f1), language (Solidity), and EVM version (compiler default). It also features a 'Compile DXHOL-contract.sol' button and checkboxes for Auto compile, Enable optimization, and Hide warnings. Below these are buttons for Publishing on Swarm and IPFS, and a 'Compilation Details' section with links for ABI and Bytecode.

The main area displays the Solidity code for the `BasicProvenance` contract. The code defines an enum `StateType` with values `Created`, `InTransit`, and `Completed`. It includes properties for `State`, `InitiatingCounterparty`, `Counterparty`, `PreviousCounterparty`, `SupplyChainOwner`, `SupplyChainObserver`, and `ContractReference`. The `constructor` initializes these variables. The `TransferResponsibility` function checks if the current state is `Completed` or if the new counterparty is the same as the sender. If either condition is true, it reverts. Otherwise, it updates the state to `InTransit`, sets the previous counterparty to the current one, and sets the new counterparty. It then emits a `TransferResp` event. The `Complete` function checks if the `SupplyChainOwner` is the sender or if the state is `Completed`. If either condition is true, it reverts.

```
pragma solidity ^0.4.25;

contract BasicProvenance {
    //Set of States
    enum StateType { Created, InTransit, Completed}

    //List of properties
    StateType public State;
    address public InitiatingCounterparty;
    address public Counterparty;
    address public PreviousCounterparty;
    address public SupplyChainOwner;
    address public SupplyChainObserver;
    int public ContractReference;

    event TransferResp(address who);
    event ContractComplete(int reference);

    constructor (address supplyChainOwner, address supplyChainObserver, int contractNumber) public {
        InitiatingCounterparty = msg.sender;
        Counterparty = InitiatingCounterparty;
        SupplyChainOwner = supplyChainOwner;
        SupplyChainObserver = supplyChainObserver;
        State = StateType.Created;
        ContractReference = contractNumber;
    }

    function TransferResponsibility(address newCounterparty) public {
        if (Counterparty != msg.sender || State == StateType.Completed)
        {
            revert();
        }
        if (State == StateType.Created)
        {
            State = StateType.InTransit;
        }
        PreviousCounterparty = Counterparty;
        Counterparty = newCounterparty;
        emit TransferResp(msg.sender);
    }

    function Complete() public {
        if (SupplyChainOwner != msg.sender || State == StateType.Completed)
        {
            revert();
        }
    }
}
```

At the bottom, there is a search bar for transaction hashes and addresses, and a note about using `exports.register(key, obj).remove(key).clear()` to register and reuse objects across script executions.

以下のコマンドで、メンバーアカウントをアンロックしてください

◎ Azure CloudShell (Bash) を使うと簡単です

```
curl -X POST -H "Content-Type:application/json" --data
'{"jsonrpc":"2.0","method":"personal_unlockAccount",
"params":["メンバーアカウントのアドレス", "メンバーアカウントに設定したパスワード",
アンロックする秒数],"id":1}' "https://hogepiyo.blockchain.azure.com:3200/アクセスキー"
```

The screenshot shows the Microsoft Azure portal interface. At the top, there's a blue header bar with the Microsoft Azure logo, a search bar, and various navigation icons. On the right side of the header, there's a user profile with the email taminami@microsoft.com and the Microsoft logo.

The main content area is titled "Azure サービス". It features a row of service icons: "リソースの作成" (Resource creation), "SQL データベース" (SQL Database), "サブスクリプション" (Subscription), "リソース グループ" (Resource Group), "Virtual Machines", "App Service", "ストレージ アカウント" (Storage Account), "Azure Database for PostgreSQL", "Azure Cosmos DB", and "その他のサービス" (Other services). There's also a large blue arrow icon pointing to the right.

Below this is the "Cloud Shell" interface, which is a terminal window. The title bar of the terminal says "Bash". The terminal output shows:

```
Bash
Requesting a Cloud Shell.Succeeded.
Connecting terminal...
Welcome to Azure Cloud Shell
Type "az" to use Azure CLI
Type "help" to learn about Cloud Shell
takunori@Azure:~$ curl -X POST -H "Content-Type:application/json" --data '{"jsonrpc":"2.0","method":"personal_unlockAccount", "params":["0x815e9bada4cabfc8c07c67ecce0d99b73fdf1262", "Ta
', 120],"id":1}' "https://taminahol.blockchain.azure.com:3200/CluaC7AjxReraxg1WkS8y5SHI"
{"jsonrpc":"2.0","id":1,"result":true}
takunori@Azure:~$
```

アドレスと Contract Number (任意の値) を入力し、「transact」をクリックしてください

The screenshot shows the Remix IDE interface. On the left, there's a sidebar with various icons. The main area has tabs for 'Home' and 'DXHOL-contract.sol'. The code editor contains Solidity code for a 'BasicProvenance' contract. The deployment section on the left is highlighted with a red box, showing fields for 'supplyChainOwner' (set to '0x9a5773d9ec637f21a3007deb8c876'), 'supplyChainObserver' (set to '0x9a5773d9ec637f21a3007deb8c876'), and 'contractNumber' (set to '0001'). A large orange 'transact' button is at the bottom of this section. Below it, there's an 'At Address' button and a 'Transactions recorded: 0' indicator. At the bottom, there's a search bar and a note about script usage.

```
pragma solidity ^0.4.25;
contract BasicProvenance {
    //Set of States
    enum StateType { Created, InTransit, Completed }

    //List of properties
    StateType public State;
    address public InitiatingCounterparty;
    address public Counterparty;
    address public PreviousCounterparty;
    address public SupplyChainOwner;
    address public SupplyChainObserver;
    int public ContractReference;

    event TransferResp(address who);
    event ContractComplete(int reference);

    constructor (address supplyChainOwner, address supplyChainObserver, int contractNumber) public {
        InitiatingCounterparty = msg.sender;
        Counterparty = InitiatingCounterparty;
        SupplyChainOwner = supplyChainOwner;
        SupplyChainObserver = supplyChainObserver;
        State = StateType.Created;
        ContractReference = contractNumber;
    }

    function TransferResponsibility(address newCounterparty) public {
        if (Counterparty != msg.sender || State == StateType.Completed)
        {
            revert();
        }
        if (State == StateType.Created)
        {
            State = StateType.InTransit;
        }
        PreviousCounterparty = Counterparty;
        Counterparty = newCounterparty;
        emit TransferResp(msg.sender);
    }

    function Complete() public
    {
        if (SupplyChainOwner != msg.sender || State == StateType.Completed)
        {
    }
```

アドレスは任意のもので大丈夫です。
例えば GitHub に公開されているものをコピーして使うことができます。

スマートコントラクトが正常に展開されたことをご確認ください

The screenshot shows the Remix IDE interface with the following details:

- Deploy & Run Transactions** sidebar:
 - Environment: Web3 Provider (Custom (1979) network)
 - Account: 0x815...f1262 (0 eth)
 - Gas limit: 3000000
 - Value: 0 wei
- Contract Source Code:** DXHOL-contract.sol

```
pragma solidity ^0.4.25;
contract BasicProvenance {
    enum StateType { Created, InTransit, Completed }
    StateType public State;
    address public InitiatingCounterparty;
    address public Counterparty;
    address public PreviousCounterparty;
    address public SupplyChainOwner;
    address public SupplyChainObserver;
    int public ContractReference;
    event TransferResp(address who);
    event ContractComplete(int reference);
    constructor (address supplyChainOwner, address supplyChainObserver, int contractNumber) public {
        InitiatingCounterparty = msg.sender;
        Counterparty = InitiatingCounterparty;
        SupplyChainOwner = supplyChainOwner;
        SupplyChainObserver = supplyChainObserver;
        State = StateType.Created;
        ContractReference = contractNumber;
    }
    function TransferResponsibility(address newCounterparty) public {
        if (Counterparty != msg.sender || State == StateType.Completed)
        {
            revert();
        }
        if (State == StateType.Created)
        {
            State = StateType.InTransit;
        }
        PreviousCounterparty = Counterparty;
        Counterparty = newCounterparty;
        emit TransferResp(msg.sender);
    }
    function Complete() public
}
```

- Deploy Section:**
 - SupplyChainOwner: "0x9a5773d9ec637f21a3007deb8c876"
 - SupplyChainObserver: "0x9a5773d9ec637f21a3007deb8c876"
 - Contract Number: 0001
 - Buttons: Deploy (disabled), transact (disabled)
- Transactions Recorded:** 1 transaction recorded.
- Deployed Contracts:** BasicProvenance at 0x644...4C65C (block 761)
- Logs:** A log entry is highlighted with a red box:

```
[block:761 txIndex:0] from:0x815...f1262 to:BasicProvenance.(constructor) value:0 wei data:0x608...00001 logs:0 hash:0x166...b69c9
```

ステップ 5

フローの作成

作成した Logic Apps リソースを開き、「空のロジックアプリ」を選択してください

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+)

ホーム > taminahol-la > Logic Apps デザイナー

Logic Apps デザイナー

一般的なトリガーで開始する
最もよく使用するトリガーの1つを選択してから、コネクタの豊富なコレクションを使用して多くのアクションを調整します

- メッセージが Service Bus キューで受信されたとき
- HTTP 要求の受信時
- 新しいツイートが投稿されたら
- Event Grid のリソースイベントが発生するとき
- 繰り返し
- 新しい電子メールが Outlook.com で受信されたとき
- 新しいファイルが OneDrive に作成されたとき
- ファイルが FTP サーバーに追加されたとき

テンプレート
ロジック アプリを作成するには、下にあるテンプレートを選択します。

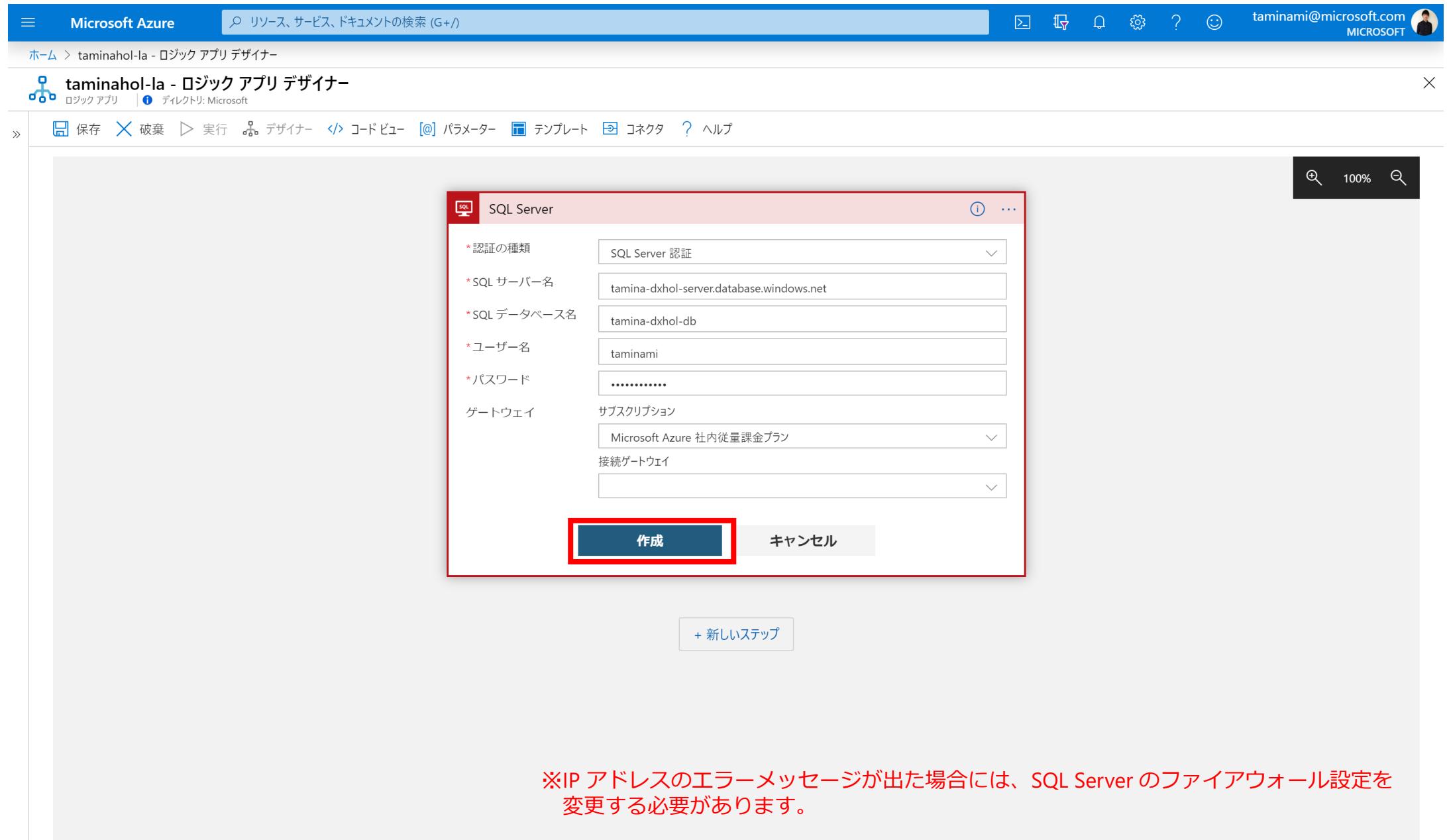
カテゴリ: すべて 並べ替え: 人気

- 空のロジック アプリ (選択可能)
- HTTP 要求と応答
- ピーコックで Service Bus メッセージを受信して完了する
- AS2 を介して X12 EDI ドキュメントを受信して XML に変換する
- Azure Monitor - Metrics Alert
- Service Bus のセッションを使用した、
- Sharepoint リストのアイテムが変更された
- AS2 ベイロードを受信し、非同期または

「SQL Server」で検索をし、「項目が作成されたとき」を選択してください

The screenshot shows the Microsoft Azure Logic Apps Designer interface. At the top, there is a navigation bar with the Microsoft Azure logo, a search bar labeled 'リソース、サービス、ドキュメントの検索 (G+ /)', and various icons for account management and help. Below the navigation bar, the URL 'ホーム > taminahol-la > Logic Apps デザイナー' is visible. The main title 'Logic Apps デザイナー' is centered above the toolbar. The toolbar includes icons for '保存' (Save), '破棄' (Delete), '実行' (Run), 'デザイナー' (Designer), 'コード ビュー' (Code View), 'パラメーター' (Parameters), 'テンプレート' (Template), 'コネクタ' (Connector), and 'ヘルプ' (Help). A search bar at the top of the content area contains the text 'SQL Server', which is highlighted with a red box. Below the search bar, there are several filter tabs: 'おすすめ' (Recommended), 'すべて' (All), 'ビルトイン' (Built-in), '標準' (Standard), 'エンタープライズ' (Enterprise), and 'カスタム' (Custom). The search results show two items under the 'トリガー' (Trigger) category: 'SQL Server' (with the '項目が作成されたとき (V2)' action selected) and 'SQL Server' (with the '項目が変更されたとき (V2)' action). Both triggers have a small '①' icon to their right. Below the triggers, a message reads '必要な情報が表示されませんか?' (Is the required information not displayed?) and '😊 次に追加するコネクタやトリガーについて、要望をお寄せください UserVoice' (For suggestions on connectors or triggers to add next, please leave a message on UserVoice).

「SQL Server 認証」を選択し、必須事項を入力後、「作成」をクリックしてください



The screenshot shows the Microsoft Azure Logic App Designer interface. A modal window titled "SQL Server" is open, prompting for connection details. The fields filled are:

- * 認証の種類: SQL Server 認証
- * SQL サーバー名: tamina-dxhol-server.database.windows.net
- * SQL データベース名: tamina-dxhol-db
- * ユーザー名: taminami
- * パスワード: (redacted)
- ゲートウェイ: サブスクリプション: Microsoft Azure 社内従量課金プラン
- 接続ゲートウェイ: (empty)

At the bottom of the modal, there are two buttons: "作成" (Create) and "キャンセル" (Cancel). The "作成" button is highlighted with a red border.

+ 新しいステップ

※IP アドレスのエラーメッセージが出た場合には、SQL Server のファイアウォール設定を変更する必要があります。

入力必須事項を選択し、「新しいステップ」をクリックしてください

The screenshot shows the Microsoft Azure Logic App Designer interface. The top navigation bar includes the Microsoft Azure logo, a search bar, and user account information (taminami@microsoft.com). The main title is "taminahol-la - ロジック アプリ デザイナー". Below the title, there are tabs for "ロジック アプリ" and "ディレクトリ: Microsoft". The toolbar contains buttons for "保存" (Save), "破棄" (Delete), "実行" (Run), "デザイナー" (Designer), "コードビューアー" (Code Viewer), "パラメーター" (Parameters), "テンプレート" (Template), "コネクタ" (Connectors), and "ヘルプ" (Help).

The main workspace displays a step configuration for "項目が作成されたとき (V2)". The configuration includes:

- * サーバー名: Use connection settings (tamina-dxhol-server.database.windows.net)
- * データベース名: Use connection settings (tamina-dxhol-db)
- * テーブル名: contractrecords
- 項目を確認する頻度:
 - * 間隔: 3
 - * 頻度: 分 (Minutes)
- 新しいパラメーターの追加

A message at the bottom states: "tamina-dxhol-db tamina-dxhol-server.database.windows.net に接続しました。接続を変更してください。"

In the bottom right corner of the workspace, a button labeled "+ 新しいステップ" (New Step) is highlighted with a red box.

「Ethereum」で検索をし、「Deploy smart contract」を選択してください

The screenshot shows the Microsoft Logic App Designer interface. At the top, there is a navigation bar with the Microsoft Azure logo, a search bar, and various icons. Below the navigation bar, the URL is [taminahol-la - ロジック アプリ デザイナー](#). The main workspace has tabs for '保存' (Save), '破棄' (Delete), '実行' (Run), 'デザイナー' (Designer), 'コード ビュー' (Code View), 'パラメーター' (Parameters), 'テンプレート' (Template), 'コネクタ' (Connector), and 'ヘルプ' (Help). A modal window titled 'アクションを選択してください' (Select an action) is open. In the search bar of this modal, the word 'Ethereum' is typed and highlighted with a red box. Below the search bar, there are filter tabs: 'おすすめ' (Recommended), 'すべて' (All), 'ビルトイン' (Built-in), '標準' (Standard), 'エンタープライズ' (Enterprise), and 'カスタム' (Custom). Under the 'ビルトイン' tab, there is a section for 'Ethereum Blockchain'. This section contains four actions: 'Deploy smart contract (プレビュー)' (Preview), 'Execute smart contract function (incurring gas cost) (プレビュー)' (Preview), 'Get smart contract state (all properties) (プレビュー)' (Preview), and 'Query smart contract function (incurring no gas cost) (プレビュー)' (Preview). The 'Deploy smart contract (プレビュー)' action is also highlighted with a red box. At the bottom of the modal, there is a message '必要な情報が表示されませんか?' (Is the required information not displayed?) and a link '(?) 次に追加するコネクタやトリガーについて、要望をお寄せください UserVoice'.

必須情報を入力し、「作成」をクリックしてください

The screenshot shows the Microsoft Azure Logic App Designer interface. A modal dialog box titled "Ethereum Blockchain" is open, prompting for connection details:

- *接続名: myblockchainconnection (任意の名前)
- *Ethereum RPC Endpoint: (コピーした Access Key)
- Private Key: Your private key (eg. 0x612a07ccb...)
- Account Address: 0x815e9bada4cabfc8c07c67ecce0d99b73fdf1262 (コピーしたアカウントアドレス)
- Account Password: (作成時に設定したパスワード)

A red box highlights the "作成" (Create) button at the bottom of the dialog.

At the bottom of the main designer area, there is a button labeled "+ 新しいステップ" (Add New Step).

ABI, Bytecode およびその他パラメータの値を設定してください (※要GitHub参照)

The screenshot shows the Microsoft Azure Logic App Designer interface. The top navigation bar includes the Microsoft Azure logo, a search bar, and user account information (taminami@microsoft.com, MICROSOFT). The left sidebar shows the project structure: Home > taminahol-la - ロジック アプリ デザイナー. The main workspace displays the logic app's configuration with several parameters listed:

- supplyChainOwner
- supplyChainObserver
- contractNumber

Below these parameters, there is a button labeled "動的なコンテンツの追加" (Add dynamic content) with a plus sign icon. At the bottom of the workspace, a message says "myblockchainconnection に接続しました。接続を変更してください。" (A connection was established with myblockchainconnection. Please change the connection.)

At the bottom center of the screen, there is a button labeled "+ 新しいステップ" (Add new step).

「新しいステップ」をクリックし、「SQL クエリを実行する」を選択してください

The screenshot shows the Microsoft Azure Logic App Designer interface. At the top, there's a navigation bar with 'Microsoft Azure', a search bar, and user information for 'taminami@microsoft.com MICROSOFT'. Below the navigation bar, the URL 'ホーム > taminahol-la - ロジック アプリ デザイナー' is visible. The main area shows a toolbar with '保存', '破棄', '実行', 'デザイナー', 'コード ビュー', 'パラメーター', 'テンプレート', 'コネクタ', and 'ヘルプ'. A large central window displays a 'アクションを選択してください' (Select an action) dialog. This dialog has a search bar containing 'SQL Server' with a red border, and tabs for 'おすすめ', 'すべて' (selected), 'ビルトイン', '標準', 'エンタープライズ', and 'カスタム'. Below the tabs, there's a section for 'トリガー' (Triggers) and 'アクション' (Actions). The 'Actions' section lists several items, with the second item, 'SQL クエリを実行する (V2)' (Run SQL query (V2)), highlighted with a red border. This action is associated with 'Power Query を使用してデータを変換します' (Transform data using Power Query) and 'SQL Server'. Other listed actions include 'ストアド プロシージャを実行する (V2)', 'テーブルを取得する (V2)', '行を削除する (V2)', and '行を取得する (V2)'. The background of the main designer area shows a partial view of the logic app steps.

必須事項を入力し、最後に「保存」をクリックしてください (※要GitHub参照)

The screenshot shows the Microsoft Azure Logic App Designer interface. At the top, there's a navigation bar with 'Microsoft Azure' and a search bar. Below it, the title bar says 'taminahol-la - ロジック アプリ デザイナー'.

The main area displays a workflow:

- A pink step labeled "項目が作成されたとき (V2)" (When item is created) with a SQL icon.
- An arrow points down to a grey step labeled "Deploy smart contract (プレビュー)" (Preview) with a diamond icon.
- From the "Deploy smart contract" step, an arrow points down to a red-highlighted step labeled "SQL クエリを実行する (V2)" (Run SQL query) with a SQL icon.

The "SQL クエリを実行する" step contains the following configuration:

- * サーバー名: Use connection settings (tamina-dxhol-server.database.windows.net)
- * データベース名: Use connection settings (tamina-dxhol-db)
- query:

```
UPDATE contractrecords SET contractAddress = '  
Smart Contract Address x ' WHERE contractID =  
contractID x |
```

To the right of the workflow, there's a sidebar with the following content:

- メッセージ: "このフローで使用されるアプリやコネクタから動的 非表示なコンテンツを追加します。"
- 動的なコンテンツ: "動的なコンテンツの検索", "Deploy smart contract", "Smart Contract Address", "Transaction Hash".
- 項目が作成されたとき (V2): "contractAddress", "contractID", "supplyChainObserver", "supplyChainOwner".

At the bottom left of the main area, there's a button labeled "+ 新しいステップ" (New step). At the top left of the main area, there's a red box around the "保存" (Save) button in the toolbar.



ステップ 6

実行 & 確認

SQL DB にデータをインサートして実験してみましょう (※要GitHub参照)

The screenshot shows the Microsoft Azure portal interface for managing a SQL database. The top navigation bar includes the Microsoft Azure logo, a search bar, and various navigation icons. The main title is "tamina-dxhol-db (tamina-dxhol-server/tamina-dxhol-db) - クエリ エディター (プレビュー)".

The left sidebar contains a navigation menu with the following sections:

- 概要
- アクティビティログ
- タグ
- 問題の診断と解決
- クイック スタート
- クエリ エディター (プレビュー)** (highlighted)
- Power Platform
 - Power BI (プレビュー)
 - PowerApps (プレビュー)
 - Flow (プレビュー)
- 設定
 - geo レプリケーション
 - 接続文字列
 - 別のデータベースに同期
 - Azure Search の追加
 - プロパティ
 - ロック
 - テンプレートのエクスポート
- 統合
- セキュリティ

The main content area displays the "クエリ エディター (プレビュー)" interface. It includes a search bar, a toolbar with "ログイン", "新しいクエリ", "クエリを開く", and "フィードバック" buttons, and a message center indicating "ここには、一部のオブジェクト エクスプローラーが表示されています。すべてを表示するには、SSDT を開いてください。".

The "クエリ 1" tab is active, showing the following SQL query:

```
1  INSERT INTO contractrecords (contractID, supplyChainOwner, supplyChainObserver) VALUES (1234, '0x9a5773d9ec637f23')
```

Below the query, there are options to "実行" (Run), cancel the query, save the query, export data as .json, .csv, or .xml. The "実行" button is highlighted.

The results section shows the message: "クエリが成功しました: Affected rows: 1". A green success message at the bottom states: "クエリが成功しました | 0秒".

Logic Apps で作成したワークフローが正常に動作したことをご確認ください

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+/-)

ホーム > taminahol-la - ロジック アプリ デザイナー

taminahol-la - ロジック アプリ デザイナー ロジック アプリ ディレクトリ: Microsoft

検索 (Ctrl+/) 保存 破棄 実行 デザイナー コード ビュー パラメーター テンプレート コネクタ ヘルプ

概要 アクティビティ ログ アクセス制御 (IAM) タグ 問題の診断と解決

開発ツール ロジック アプリ デザイナー ロジック アプリ コード ビュー バージョン API 接続 クイック スタート ガイド リリース ノート

設定 ワークフロー 設定 アクセス キー ID プロパティ ロック テンプレート の エクスポート

監視 警告 メトリック

The screenshot shows the Microsoft Azure Logic Apps Designer interface. On the left, there's a sidebar with navigation links like '概要', 'アクティビティ ログ', 'アクセス制御 (IAM)', etc. The main area displays a workflow with three steps:

- Step 1: '項目が作成されたとき (V2)' (When item is created) with a duration of '1秒' (1 second). It has a green checkmark icon.
- Step 2: 'Deploy smart contract' with a duration of '9秒' (9 seconds). It also has a green checkmark icon.
- Step 3: 'SQL クエリを実行する (V2)' (Run SQL query) with a duration of '0秒' (0 seconds). This step is also marked with a green checkmark.

Below the workflow, there are buttons for '保存' (Save), '破棄' (Delete), '実行' (Run), 'デザイナー' (Designer), 'コード ビュー' (Code View), 'パラメーター' (Parameters), 'テンプレート' (Template), 'コネクタ' (Connectors), and 'ヘルプ' (Help). There's also a search bar at the top right and a status bar showing '100%'.

Contract Address が格納されていることをご確認ください

Microsoft Azure リソース、サービス、ドキュメントの検索 (G+/)

ホーム > tamina-dxhol-db (tamina-dxhol-server/tamina-dxhol-db) - クエリ エディター (プレビュー)

tamina-dxhol-db (tamina-dxhol-server/tamina-dxhol-db) - クエリ エディター (プレビュー)

SQL データベース ディレクトリ: Microsoft

検索 (Ctrl+ /) ログイン 新しいクエリ クエリを開く フィードバック

概要 アクティビティ ログ タグ 問題の診断と解決 ケイック スタート クエリ エディター (プレビュー)

Power Platform Power BI (プレビュー) PowerApps (プレビュー) Flow (プレビュー)

設定 設定 geo レプリケーション 接続文字列 別のデータベースに同期 Azure Search の追加 プロパティ ロック テンプレートのエクスポート

統合 Stream Analytics (プレビュー)

セキュリティ

tamina-dxhol-db (taminami) クエリ 1 ×

ここには、一部のオブジェクトエクスプローラーが表示されています。すべてを表示するには、SSDT を開いてください。

実行 クエリのキャンセル クエリの保存 データを .json としてエクスポート データを .csv としてエクスポート データを .xml としてエクスポート

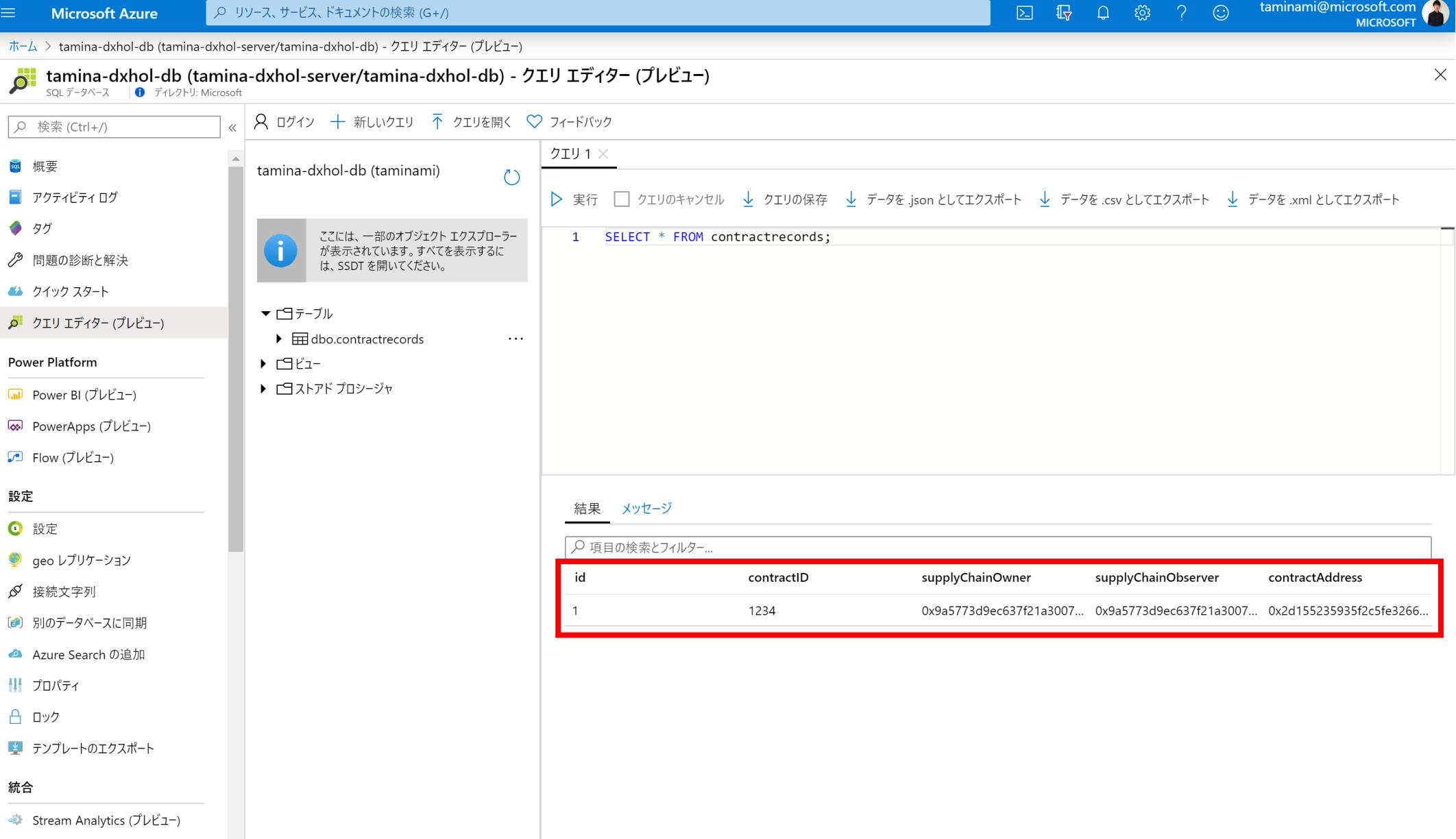
1 SELECT * FROM contractrecords;

結果 メッセージ

項目の検索とフィルター...

id	contractID	supplyChainOwner	supplyChainObserver	contractAddress
1	1234	0x9a5773d9ec637f21a3007...	0x9a5773d9ec637f21a3007...	0xd155235935f2c5fe3266...

クエリが成功しました | 0秒





© 2020 Microsoft Corporation. All rights reserved.

本情報の内容(添付文書、リンク先などを含む)は、2019年9月時点のものであり、予告なく変更される場合があります。

本コンテンツの著作権、および本コンテンツ中に出てくる商標権、団体名、ロゴ、製品、サービスなどはそれぞれ、各権利保有者に帰属します。