

演習 1 - シンプルなAPIの作成

この演習では、既存のAPIを呼び出すAPIをAPI Connectに登録してテストを行います。

演習 2 - 目的

この演習では、以下の内容を理解できます。

- API Managerへのログイン方法
- 既存APIを呼び出すAPIの定義方法
- API ManagerでのAPIの開発、単体テストの基本的な操作方法

1.1 - 事前作業としてのゲートウェイの構成

1. API Managerにアクセスしてログインします。

ユーザー・レジストリーの選択画面が表示された場合には、API Manager User Registry を選択し、ユーザー名、パスワードを入力して サインイン をクリックしてAPI Managerにログインします。

項目	入力値
ユーザー名	apic.userxxx
パスワード	Passw0rd!

```
> ![] [info]
>
> ユーザー名の*xxx*の部分には、受講生ごとに割り当てられた数字に置き換えてください。今後、API Manager
にログインする場合には、この手順でログインしてください。
```

1. カタログのゲートウェイ設定を確認して設定します。API Managerのトップページで カタログの管理 をクリックします。

2. Sandbox を選択します。

1. 左のメニューから 設定 をクリックします。

2. ゲートウェイ・サービス メニューを開き、編集 をクリックします。

- 利用可能なゲートウェイ・サービスが表示されています。利用するゲートウェイ・サービスにチェックが入っていない場合には、チェックを入れて **保存** をクリックします。

```
> ![][info]
>
> API Connect v2018では2種類のゲートウェイ・タイプが利用できます。API Connect v5ベースのランタイム・アーキテクチャーの`DataPower Gateway (v5 互換)`と、新しい`DataPower API Gateway`です。新しい`DataPower API Gateway`では、API処理に特化しており、ランタイムのパフォーマンスが向上しています。それぞれのゲートウェイ・タイプで一部利用できないポリシーがあるため注意してください。詳細は以下のリンクで確認できます。
> + [マニュアル : API Connect のゲートウェイ・タイプ]
(https://www.ibm.com/support/knowledgecenter/ja/SSMNED_2018/com.ibm.apic.overview.doc/ra
:
https://www.ibm.com/support/knowledgecenter/ja/SSMNED_2018/com.ibm.apic.overview.doc/ra

> + [APIGW Porting Notes] (https://github.com/ibm-apiconnect/apigw/wiki/APIGW-Porting-
Notes) : https://github.com/ibm-apiconnect/apigw/wiki/APIGW-Porting-Notes

> ![][info]
>
> この作業は、新しいカタログを利用する場合に、最初に必要なになります。Sandboxカタログはデフォルトのカタログです。カタログとは、APIのデプロイ先となるゲートウェイ上の環境の単位です。APIを公開する際に、公開先のカタログを指定します。APIを公開するとAPIのエンドポイントにカタログ名が含まれます。

以上でAPIを作成する準備作業が終了しました。
```

1.2 - APIの作成

- API Managerの左のメニューから **開発** に移動します。
- 開発** 画面で、**追加** メニューから **API** を選択します。
- ターゲット・サービス**から **を選択し、次へ** をクリックします。
- 作成するAPIの情報を以下のように入力し、**次へ** をクリックします。

項目	入力値	備考
タイトル	branch	APIの名前
バージョン	branch	デフォルトで 1.0.0 が入力されます

基本パス	branch	デフォルトでAPIのタイトルと同じ名前が入力されます
ターゲット・サービスURL	https://apictutorials.mybluemix.net/branches	このAPIが呼び出すターゲットサービスを入力します

```
> ![] [info]
>
> `ターゲット・サービスURL` に入力するURLは、店舗情報をGET要求で取得するAPIです。ブラウザーの別のタブを開き、`ターゲット・サービスURL` に入力したURLをコピー&ペーストして、APIのレスポンスが返ることを確認してください。
```

1. まずは、単純にプロキシするAPIを作成するため、`クライアント IDを使用した保護` のチェックをはずし、「次へ」をクリックします。
2. APIの雛形が完成します。`APIの編集` をクリックして、APIの設定の確認・編集を行います。
3. API設計のメニューから `プロパティ` を開くと、`target-url` というプロパティが自動定義されています。
`target-url` をクリックすると、デフォルト値に、API作成時のウィザードで入力したターゲット・サービスURLがセットされています。

プロパティは、デプロイしたカタログに応じてAPIのパラメーターを変えたい場合に便利です。プロパティを利用すれば、デプロイするカタログを変える際に、API定義を修正する必要がありません。ここでは、API Gatewayから呼び出すURLをカタログごとに変えることができるように、プロパティが設定されています。

ここまででシンプルなAPIのインターフェース定義が完了しました。

1.3 - アセンブルの定義

1. アセンブルでは、受け付けたAPI要求に対する処理内容を定義します。API開発画面上部から `アセンブル` をクリックして、アセンブル画面に移動します。
2. アセンブル画面が表示されます。アセンブルでは、受け付けたAPI要求に対する処理内容を定義します。

左側の **パレット** には、アセンブルで利用することができるコンポーネントが表示されています。コンポーネントを左側のパレットから真ん中の **キャンバス** 上の **アセンブリー・フロー** 上にドラッグ&ドロップして、処理を組み立てることができます。

アセンブリー・フロー では、受け付けたAPI要求に対する処理内容を定義します。API Connectは、API呼び出しが行われると、まずセキュリティが適用され、次にレート制限が適用された上で、**アセンブリー・フロー** に定義された内容に基づいて処理が行い、API要求元に対して応答を行います。

アセンブリー・フロー 上に配置されている **ポリシー** をクリックすると、右側にその **ポリシー** の **プロパティ・シート** が表示されます。 **プロパティ・シート** にポリシー内での処理を設定します。 **アセンブリー・フロー** には、 **invoke** ポリシーがすでに配置されています。 **invoke** ポリシーは外部のサービス呼び出すポリシーです。 **invoke** ポリシーをクリックすると、右側に **プロパティ・シート** 開きます。 **invoke** ポリシーの URL には、プロパティで定義している `${target-url}` が入力されています。プロパティで定義した値は、 `${}` を利用して指定します。

ここまでで、既存のAPIをパススルーして公開するシンプルなAPIの定義が完了しました。

1.4 - APIのテスト

- 作成したAPIのテストを行います。アセンブル画面で利用できるAPIのテストツールを利用します。画面上のボタンをクリックしてテストツールを表示します。
- APIのアクティブ化** をクリックします。 **APIのアクティブ化** をクリックすると、テスト用にAPIがカタログ上にデプロイされ、起動されます。デプロイ先のカタログとして **sandbox** が指定されています。APIの公開には、製品やプランも必要ですが、テストツールにより自動的に作成されます。
- しばらくすると、セットアップ・メニュー上に、作成された製品名やプラン名が表示され、右上のAPIのステータスが **実行中** に変更されます。これで準備が整ったのでAPIのテストを実行できます。
- 操作・メニューで呼び出す操作として **get /** を選択し、 **呼び出し** をクリックしてAPIを呼び出します。
- API応答が表示されます。状況コードが **200 OK** と表示されており、応答本文にJSON形式の応答が表示されていることを確認します。
- ここまでで、既存のAPIをパススルーで呼び出すシンプルなAPIの定義とテストが完了しました。右上の **保存** ボタンをクリックして、API定義を保存し、ホームメニューをクリックして、ホームに戻ります。

以上で、演習1は終了です。

続いて、[演習 2 - APIへの定義の追加](#)に進んでください。