



外部サービス連携
OAuth 関連ライブラリ
シナリオ作成マニュアル

NTTアドバンステクノロジー株式会社

目次

1. はじめに	1
1.1. 動作対象	1
1.2. 制限事項	1
2. シナリオ作成手順	3
2.1. 認証情報の取得	3
2.2. シナリオへのライブラリの配置とプロパティ設定	4
2.2.1. 「アクセストークンの取得」ライブラリの配置とプロパティ設定	4
2.2.2. 「RESTful API 実行」ライブラリの配置とプロパティ設定	6
2.2.3. 「アクセストークンの取り消し」ライブラリの配置とプロパティ設定	7
3. ライブラリ・プロパティの一覧	8
3.1. アクセストークンの取得	9
3.2. RESTful API 実行	10
3.3. アクセストークンの取り消し	12
4. 特記事項	13
4.1. 認証情報の保護	13
4.2. シナリオ実行中エラーの解析	14
4.3. プロキシの利用	18

WinActor 外部サービス連携 OAuth関連シナリオ作成マニュアル

1. はじめに

本マニュアルは、WinActor を使用して外部サービスの RESTful API 実行を行うシナリオを作成する手順について説明します。外部サービスの RESTful API 実行を行うシナリオの作成には OAuth 関連ライブラリを使用します。

OAuth 関連ライブラリをご利用の際は、OAuth2.0 の認証方式に関する知識と RESTful API に関する知識が必要になります。

OAuth 関連ライブラリは、OAuth2.0 による認可の制御や、認可情報を使用した API へのアクセスを提供する各ライブラリの総称です。

本マニュアルが想定する外部サービスについては、「1.1 動作対象」をご参照ください。

1.1. 動作対象

OAuth 関連ライブラリを使用して外部サービスの RESTful API 実行を行う場合、対象となる外部サービスは OAuth2.0 によるアクセスの認可制御(RFC6749、RFC7009)と RESTful API を実装していることを前提とします。

動作実績のある外部サービスを表 1-1 に示します。(2019年1月25日時点)

表 1-1 OAuth 関連ライブラリ 動作実績のある外部サービス一覧

No.	外部サービス名	API バージョン
1	Salesforce.com (※1)	44.0

※1 Salesforce は Salesforce.com,inc の商標登録です。

1.2. 制限事項

OAuth 関連ライブラリを使用して外部サービスの RESTful API 実行を行う場合、次の制限事項が発生します。

- OAuth2.0 による認可グラントとして使用可能なのは「リソースオーナーパスワードクレデンシャル」のみです。
- 外部サービスから OAuth 関連ライブラリを呼び出すことはできません。
- セキュリティ上の理由により、HTTPS を使用したアクセスに限定しています。
- HTTP メソッドとして使用可能なのは、GET、POST、PATCH、DELETE のみです。
- OAuth2.0 の認可情報(取得したアクセストークンなど)は、WinActor 変数として取り出すことはできません。
- RESTful API 実行の HTTP レスポンスにおけるステータスコード以外のヘッダは、参

WinActor 外部サービス連携 OAuth関連シナリオ作成マニュアル

照できません。

例として、ヘッダにて情報を返却する API や cookie などは利用できません。

2. シナリオ作成手順

OAuth 関連ライブラリを使用して、外部サービスの RESTful API 実行を行います。

OAuth 関連ライブラリの「アクセストークンの取得」「RESTful API 実行」「アクセストークンの取り消し」ライブラリを例にとって、基本的なシナリオの作成手順を説明します。

[手順]

2.1 認証情報の取得

2.2 シナリオへのライブラリの配置とプロパティ設定

2.2.1 「アクセストークンの取得」ライブラリの配置とプロパティ設定

2.2.2 「RESTful API 実行」ライブラリの配置とプロパティ設定

2.2.3 「アクセストークンの取り消し」ライブラリの配置とプロパティ設定

2.1. 認証情報の取得

OAuth2.0 による認可を行う際には、外部サービスより事前に認証用の各種情報を取得する必要があります。

OAuth2.0 による認可に必要な情報を表 2-1 に示します。

なお、各情報の取得方法については外部サービスが提供するドキュメントを確認するか、外部サービスの提供者にお問い合わせください。

表 2-1 OAuth2.0 認可に使用する認証情報一覧

No.	名称	説明
1	ユーザ名	アクセストークン取得時に使用するアカウントのユーザ名です。
2	パスワード	アクセストークン取得時に使用するアカウントのパスワードです。
3	クライアント ID	外部サービス登録時に得られる文字列で RFC6749 の client_id に該当します。
4	クライアントシークレット	クライアント ID と同様に外部サービス登録時に得られる文字列で RFC6749 の client_secret に該当します。
5	セキュリティトークン	外部サービスによっては本プロパティを使用します。不要な場合は空文字列を設定してください。本プロパティが設定された場合、OAuth 関連ライブラリは認可の際にパスワードと連結して使用します。(自動で連結します)

WinActor 外部サービス連携 OAuth関連シナリオ作成マニュアル

2.2. シナリオへのライブラリの配置とプロパティ設定

2.2.1. 「アクセストークンの取得」ライブラリの配置とプロパティ設定

フローチャート画面のライブラリタブを選択し、「99_外部サービス連携」とその配下に存在する「01_OAuth 関連」を順にダブルクリックして展開します。

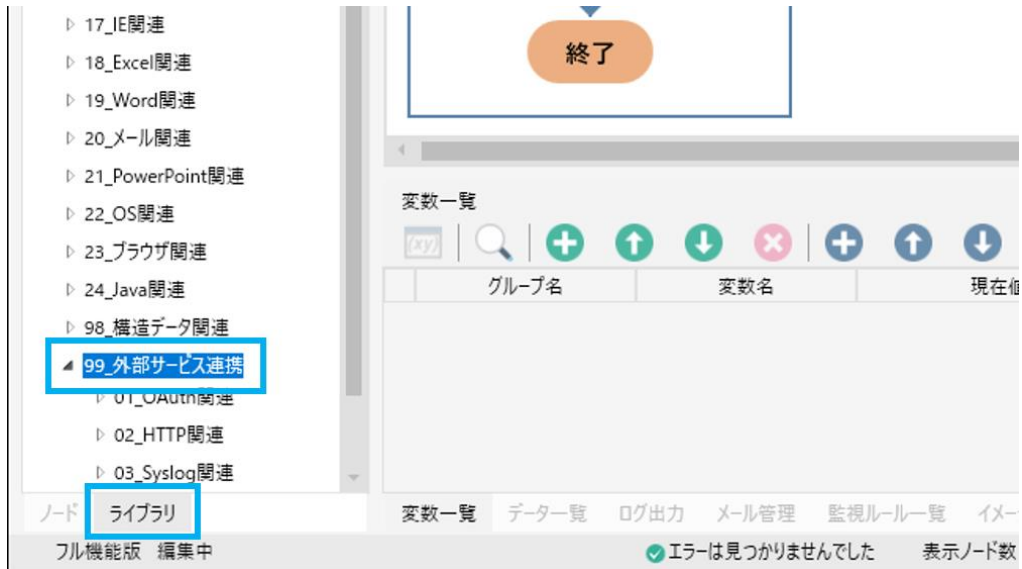


図 2-1 ライブラリタブから「99_外部サービス連携」を展開

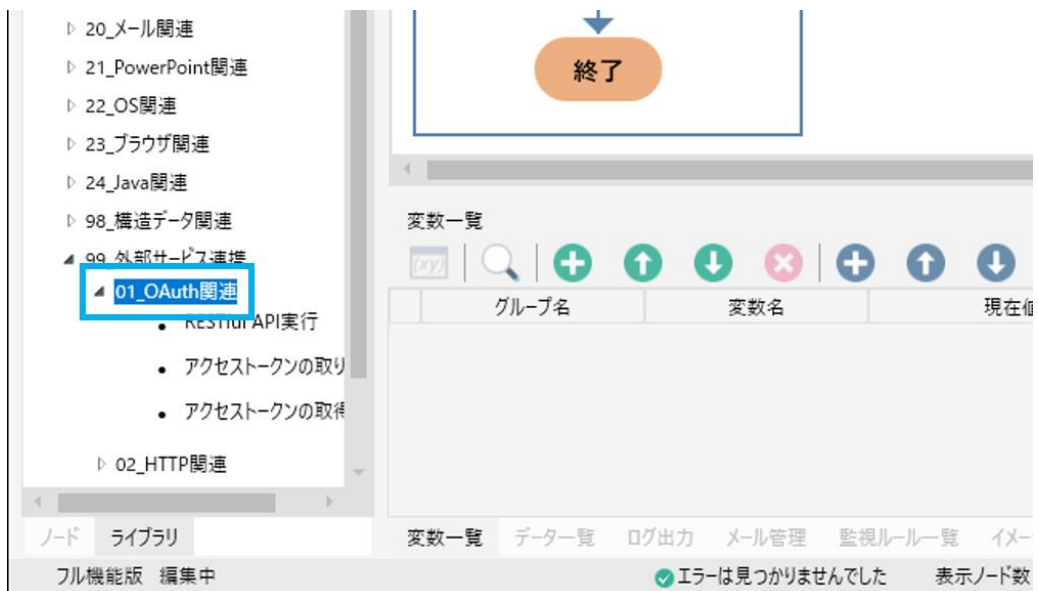


図 2-2 ライブラリタブから「01_OAuth 関連」を展開

WinActor 外部サービス連携 OAuth関連シナリオ作成マニュアル

ライブラリタブから「アクセストークンの取得」ライブラリ (図中①) を選択し、シナリオに配置します (図中②)。次にシナリオに配置した「アクセストークンの取得」ライブラリをダブルクリックしてプロパティを表示します (図中③)。

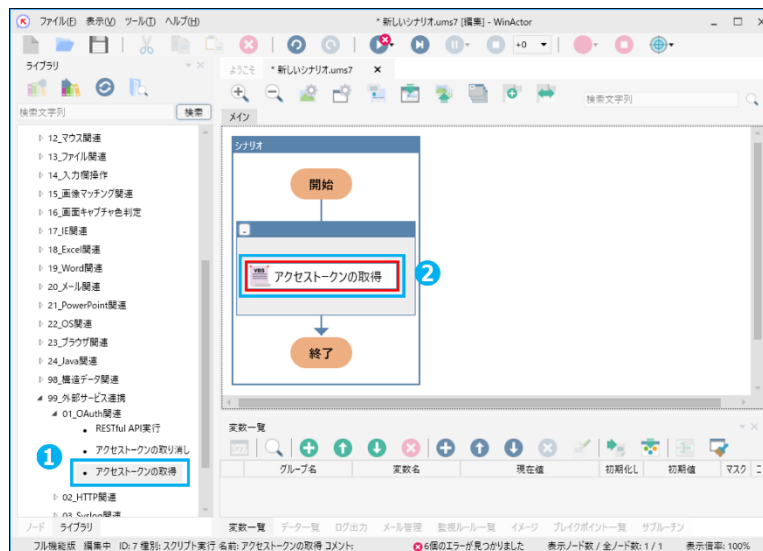


図 2-3 「アクセストークンの取得」ライブラリをシナリオに配置

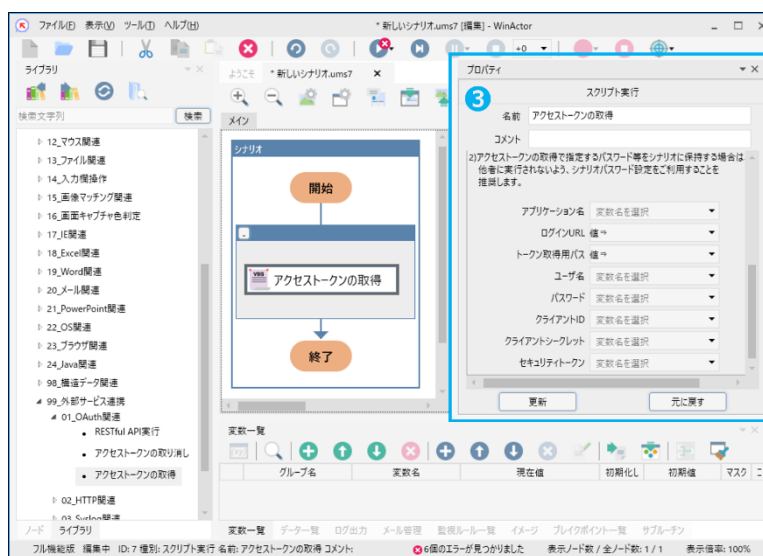


図 2-4 「アクセストークンの取得」ライブラリのプロパティエリア

各プロパティ項目を設定し、更新ボタンをクリックします。各プロパティ項目の説明については、「3.1 アクセストークンの取得」をご参照ください。なお、アプリケーション名については「3 ライブラリ・プロパティの一覧」で解説していますのでご参照ください。

WinActor 外部サービス連携 OAuth関連シナリオ作成マニュアル

2.2.2. 「RESTful API 実行」ライブラリの配置とプロパティ設定

2.2.1 と同様にライブラリタブから「RESTful API 実行」ライブラリ (図中①) を選択し、シナリオに配置します (図中②)。次にシナリオに配置した「RESTful API 実行」ライブラリをダブルクリックしてプロパティを表示します (図中③)。

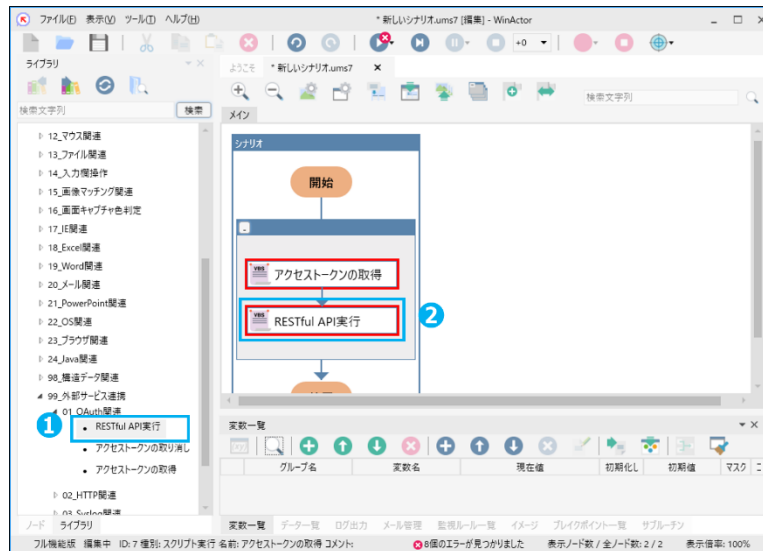


図 2-5 「RESTful API 実行」ライブラリをシナリオに配置

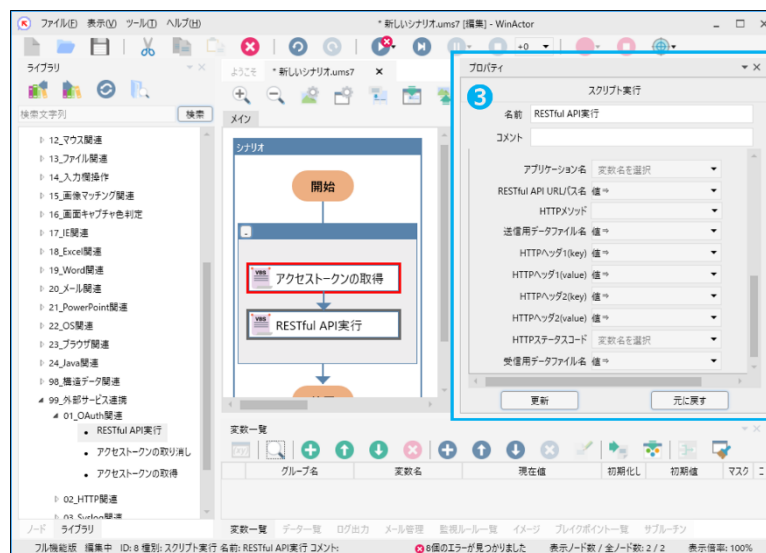


図 2-6 「RESTful API 実行」ライブラリのプロパティエリア

各プロパティ項目を設定し、更新ボタンをクリックします。各プロパティ項目の説明については、「3.2 RESTful API 実行」をご参照ください。なお、アプリケーション名については「3 ライブラリ・プロパティの一覧」で解説していますのでご参照ください。

WinActor 外部サービス連携 OAuth関連シナリオ作成マニュアル

2.2.3. 「アクセストークンの取り消し」ライブラリの配置とプロパティ設定

2.2.1、2.2.2 と同様にライブラリタブから「アクセストークンの取り消し」ライブラリ (図中①) を選択し、シナリオに配置します (図中②)。次にシナリオに配置した「アクセストークンの取り消し」ライブラリをダブルクリックしてプロパティを表示します (図中③)。

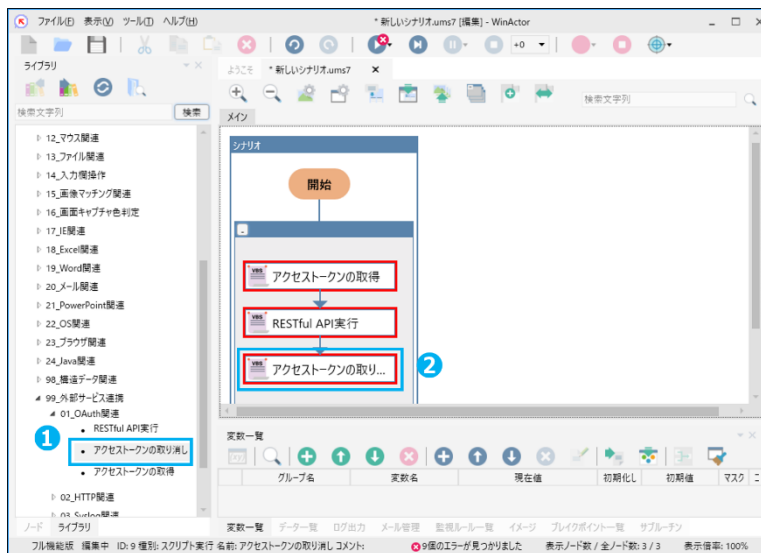


図 2-7 「アクセストークンの取り消し」ライブラリをシナリオに配置

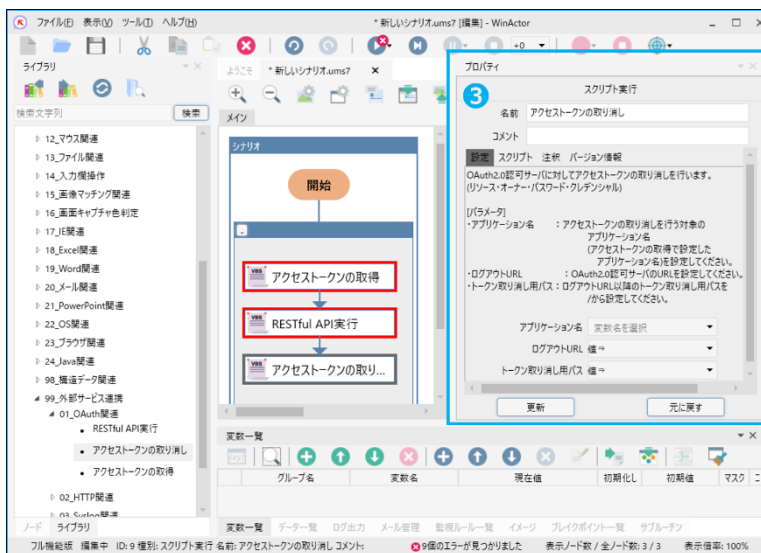


図 2-8 「アクセストークンの取り消し」ライブラリのプロパティエリア

各プロパティ項目を設定し、更新ボタンをクリックします。各プロパティ項目の説明については、「3.3 アクセストークンの取り消し」をご参照ください。なお、アプリケーション名については「3 ライブラリ・プロパティの一覧」で解説していますのでご参照ください。

3. ライブラリ・プロパティの一覧

本章では OAuth 関連ライブラリに属するライブラリと、各ライブラリのプロパティ項目を示します。

なお、OAuth 関連ライブラリにおいては共通的なプロパティ項目としてアプリケーション名が存在します。

アプリケーション名とは、接続先を識別するために使用し、ユニークな 1 文字以上の文字列を指定します。同じアプリケーション名を使用することで、同一の認証情報および認可情報を複数の処理で共有することができます。

次の例は 2 つのアカウント A,B を使用して RESTful API を実行する例です。

例：

- 1) アプリケーション名”a”として、アカウント A にてアクセストークンの取得を行う。
- 2) アプリケーション名”b”として、アカウント B にてアクセストークンの取得を行う。
- 3) アプリケーション名”a”として、RESTful API 実行を行う。
※ アカウント A の情報を元に、API へアクセスを行う。
- 4) アプリケーション名”b”として、RESTful API 実行を行う。
※ アカウント B の情報を元に、API へアクセスを行う。
- 5) アプリケーション名”a”として、アクセストークンの取り消しを行う。
※ アカウント A の情報が破棄される。
- 6) アプリケーション名”b”として、アクセストークンの取り消しを行う。
※ アカウント B の情報が破棄される。

上記のようにアカウントとアプリケーション名が 1 対 1 に対応するような使用方法を推奨します。

なお、アプリケーション名に紐づけられた認証情報(アカウント情報)および認可情報は、シナリオ内で「アクセストークンの取り消し」ライブラリを使用し、必ず明示的に破棄してください。

特に、「RESTful API 実行」ライブラリのように通信失敗など環境に起因する事象によってシナリオ実行中にエラーが発生しうるライブラリを使用する場合、例外処理の実行後に「アクセストークンの取り消し」ライブラリをシナリオに入れ、アプリケーション名に紐づけられた認証情報(アカウント情報)および認可情報の破棄を行うなどの工夫が必要となります。

WinActor 外部サービス連携 OAuth関連シナリオ作成マニュアル

3.1. アクセストークンの取得

「アクセストークンの取得」ライブラリは、OAuth2.0 認可サーバよりアクセストークンを取得します。

ユーザによって指定された認証情報を元に認可を行い、その結果である認可情報を取得します。取得に成功した場合、認証情報および認可情報(アクセストークン、リソースサーバの URL※)をアプリケーション名と紐づけます。

※アクセストークン、リソースサーバの URL はアプリケーション名に紐づいて管理されるため、ユーザが意識する必要はありません。

表 3-1 ライブラリ・プロパティ項目

ライブラリ名	プロパティ項目	説明
アクセストークンの取得	アプリケーション名	接続先を識別するための文字列を変数で指定します。
	ログイン URL	OAuth2.0 認可サーバのホスト名を値または変数で指定します。 (※1、※2)
	トークン取得用パス	OAuth2.0 認可サーバからアクセストークンを取得するためのパス名を値または変数で指定します。(※2、※3)
	ユーザ名	「2.1 認証情報の取得」で取得した認証情報を変数で指定します。
	パスワード	
	クライアント ID	
	クライアントシークレット	
セキュリティトークン		

※1 https プロトコルの URL を指定してください。

※2 以下にログイン URL とトークン取得用パスの指定例を示します。

例：

アクセストークン取得用の URL : “https://access.example.com/oauth2/token”

ログイン URL : “https://access.example.com”

トークン取得用パス : “/oauth2/token”

※3 “/”文字から始まるパス名を指定してください。

WinActor 外部サービス連携 OAuth関連シナリオ作成マニュアル

3.2. RESTful API 実行

「RESTful API 実行」ライブラリは、OAuth2.0 による認可が必要な RESTful API を実行します。

「アクセストークンの取得」ライブラリによって得られた認可情報(アクセストークン、リソースサーバの URL)を元に、RESTful API URL パス名に対してアクセスを行い、HTTP ステータスコードとレスポンスボディを返却します。また、アクセスの際にアクセストークンの有効期限が超過していた場合、アクセストークンの更新を併せて行います。

なお、「RESTful API 実行」ライブラリは、リソースサーバからのレスポンスを受信し、処理を終了します。RESTful API における処理の成否については、シナリオ内にてユーザが明示的に判定を行う必要があります。

表 3-2 ライブラリ・プロパティ項目

ライブラリ名	プロパティ項目	説明
RESTful API 実行	アプリケーション名	接続先を識別するための文字列を変数で指定します。(※1)
	RESTful API URL パス名	RESTful API のパス名を値または変数で指定します。(※2、※3)
	HTTP メソッド	リクエストに指定するメソッドを選択します。 PATCH、GET、POST、DELETE が選択可能です。
	送信用データファイル名	リクエストのボディとして送信するファイルパスを値または変数で指定します。(※4) 省略時は、ボディが空であるリクエストを送信します。
	HTTP ヘッダ 1(key) HTTP ヘッダ 1(value) HTTP ヘッダ 2(key) HTTP ヘッダ 2(value)	リクエストに設定するヘッダの名称と内容を値または変数で指定します。(※5、6)
	HTTP ステータスコード	受信したレスポンスのステータスコードを格納する変数を指定します。

WinActor 外部サービス連携 OAuth関連シナリオ作成マニュアル

	受信用データファイル名	受信したレスポンスのボディを出力するファイルパスを値または変数で指定します。(※4) 省略時またはレスポンスのボディが空の場合、ファイルへの出力を行いません。
--	-------------	--

- ※1 事前にシナリオに配置した「アクセストークンの取得」ライブラリで指定したアプリケーション名と同じアプリケーション名を指定します。
- ※2 “/”文字から始まるパス名を指定してください。
- ※3 以下に RESTful API URL パス名の指定例を示します。なお、リソースサーバの URL は「アクセストークンの取得」ライブラリで得られたものとなります。
例：
RESTful API 実行用の URL : “https://resource.example.com/services/data”
リソースサーバの URL : “https://resource.example.com”
RESTful API URL パス名 : “/services/data”
- ※4 相対パスを指定した場合は、WinActor のインストール先フォルダ (WinActor7.exe のあるフォルダ) からの相対パスになります。
- ※5 HTTP ヘッダは HTTP ヘッダ 1 と HTTP ヘッダ 2 の最大 2 組まで指定可能です。なお、HTTP ヘッダの設定が不要な場合は、空文字列を設定してください。
- ※6 OAuth2.0 の認可情報を指定するヘッダ(Authorization)は自動的に設定されます。

WinActor 外部サービス連携 OAuth関連シナリオ作成マニュアル

3.3. アクセストークンの取り消し

「アクセストークンの取り消し」ライブラリは、OAuth2.0 認可サーバにアクセストークンの取り消しを要求します。

なお、本ライブラリは取り消しの成功・失敗に関わらず、アクセストークンを取得していない状態に戻します。

表 3-3 ライブラリ・プロパティ項目

ライブラリ名	プロパティ項目	説明
アクセストークンの取り消し	アプリケーション名	接続先を識別するための文字列を変数で指定します。
	ログアウト URL	OAuth2.0 認可サーバのホスト名を値または変数で指定します。 (※1、※2)
	トークン取り消し用パス	OAuth2.0 認可サーバにアクセストークン取り消しを要求するためのパス名を値または変数で指定します。(※2、※3)

※1 空文字列または https プロトコルの URL を指定してください。

空文字列を設定した場合、リソースサーバの URL が自動的に使用されます。

※2 以下にログアウト URL とトークン取り消し用パスの指定例を示します。

例：

アクセストークンの

取り消し用の URL : “https://access.example.com/oauth2/revoke”

ログアウト URL : “https://access.example.com”

トークン取り消し用パス : “/oauth2/revoke”

※3 “/”文字から始まるパス名を指定してください。

4. 特記事項

OAuth 関連ライブラリを使用して外部サービスの RESTful API 実行を行うにあたり、特筆すべき事項を説明します。

4.1. 認証情報の保護

- ・「アクセストークンの取得」ライブラリのプロパティであるユーザ名、パスワード、クライアント ID、クライアントシークレット、セキュリティトークンについては他者に変数値を見られないように変数値のマスクをご利用されることを推奨します。変数一覧画面のマスク欄にチェックを付けることで、変数値の表示にマスクをかけることができます。

※変数値のマスクについての詳細は『WinActor 操作マニュアル』をご参照ください。

- ・「アクセストークンの取得」ライブラリで指定するパスワード等をシナリオに保持する場合は、他者にシナリオを実行されないように、シナリオパスワード設定をご利用することを推奨します。

※シナリオパスワード設定についての詳細は『WinActor 操作マニュアル』をご参照ください。

4.2. シナリオ実行中エラーの解析

本章ではシナリオ実行中エラーの解析を行う方法を説明します。OAuth 関連ライブラリはシナリオ実行中にエラーが発生した場合、ログ出力画面にメッセージ「ライブラリ実行 (spv_OAuth) に失敗しました。」を出力します。また、実行中に発生したエラーの詳細は「例外処理」ノードと「SPV エラー情報収集」ライブラリを組み合わせることにより取得できます。取得手順は次の通りです。

- ① 「例外処理」ノードをシナリオに配置
- ② 「例外処理」ノードの「正常系」に「アクセストークンの取得」ライブラリを配置
- ③ 「例外処理」ノードの「異常系」に以下の順番でライブラリ及びノードを配置
 1. 「SPV エラー情報収集」ライブラリ
 2. 「クリップボード」ノード
 3. 「テキストファイル書込」ライブラリ

※「SPV エラー情報収集」ライブラリはクリップボードに情報を出力するため、「クリップボード」ノードで情報を変数（変数名の例：error_log）にコピーし、「テキストファイル書込」ライブラリで指定のパス（ファイル）に書き出すようにします。

シナリオに配置した図を図 4-1 に示します。

※ここでは「アクセストークンの取得」ライブラリを例に挙げていますが、「RESTful API 実行」ライブラリ、「アクセストークンの取り消し」ライブラリも同様の方法でエラー情報を収集することができます。

※「例外処理」ノードの詳細については『WinActor 操作マニュアル』をご参照ください。
※「SPV エラー情報収集」ライブラリの詳細については『WinActor ユーザライブラリ サンプル説明書』をご参照ください。

WinActor 外部サービス連携 OAuth関連シナリオ作成マニュアル

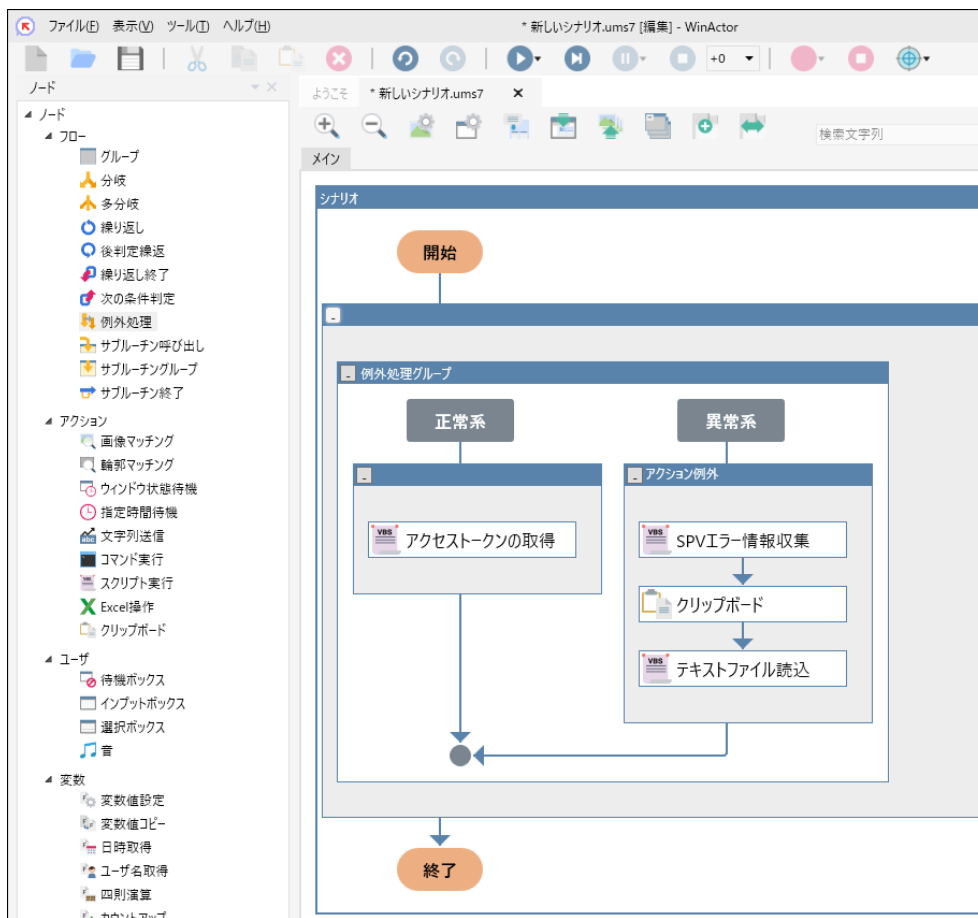


図 4-1 OAuth 関連ライブラリの実行中のエラー情報の収集方法

WinActor 外部サービス連携 OAuth関連シナリオ作成マニュアル

表 4-1は SPV エラー情報収集ライブラリによって得られるエラー情報の例を示しています。

表 4-1 SPV エラー情報の例

エラー情報	エラーの内容と確認ポイント
Unintended response. (code:xxx) <以下略> xxx は任意の数字	外部サービスが異常を検出したことを示しています。ユーザ名やパスワード、クライアント ID、クライアントシークレットが空でないこと (入力されていること) やパスワードに誤りがないことを確認してください。
Please set token acquisition path with starting / <以下略>	トークン取得用パスが/から始まっていないことを示しています。トークン取得用パスを確認してください。
Please set REST URL path with starting / <以下略>	RESTful API URL パス名が/から始まっていないことを示しています。RESTful API URL パス名を確認してください。
Please set token revocation path with starting / <以下略>	トークン取り消し用パスが/から始まっていないことを示しています。トークン取り消し用パスを確認してください。
Failed to read file. (送信用ファイル名) <以下略>	送信用ファイルを読み込めなかったことを示しています。送信用ファイル名のパスが正しいこと、ファイルが存在していることを確認してください。
Failed to write file. (受信用ファイル名) <以下略>	受信用ファイルに書き込みできなかったことを示しています。受信用ファイル名のパスが正しいことを確認してください。
unmatch argument designation between key and value. (key:, value:xxx) <以下略> xxx は任意の値	HTTP ヘッダの key 値が空であることを示しています。正しい key 値を設定してください。

WinActor 外部サービス連携 OAuth関連シナリオ作成マニュアル

unmatch argument designation between key and value. (key:xxx, value:) <以下略> xxx は任意の値	HTTP ヘッダの value 値が空であることを示しています。正しい value 値を設定してください。
failed to communicate. <以下略>	通信に失敗したことを示しています。指定した URL に誤りがないこと、およびネットワークへの接続設定を確認してください。またプロキシ使用時は、プロキシサーバの指定に誤りがないことを確認してください。
Application has not been login yet. <以下略>	アクセストークンを取得していない状態にあることを示しています。本エラーが発生する前までに「アクセストークンの取得」ライブラリを実行し、成功していることを確認してください。

4.3. プロキシの利用

OAuth 関連ライブラリが外部サービスにアクセスする際にプロキシを経由する必要がある場合は WinActor 本体のプロキシ設定が行われていることを確認してください。

WinActor 本体のプロキシの設定方法につきましては『WinActor インストールマニュアル』または『WinActor 操作マニュアル』をご参照ください。



外部サービス連携
OAuth 関連ライブラリ
シナリオ作成マニュアル

NTTアドバンステクノロジー株式会社

Copyright© 2020 NTT Advanced Technology Corp. All Rights Reserved.

本書は著作権法上の保護を受けています。本書の一部あるいは全部を無断で複写、複製することは禁じられています。

WA7-G-20200808
