記事

Mihoko lijima · 2020年10月22日 9m read

VSCode を使ってみよう!

皆さんこんにちは!<u>Virtual Summit</u> ご覧いただけていますでしょうか。

<u>Virtual Summit</u> で VSCode の <u>ObjectScript 用エクステンションバージョン1.0</u> のリリースが 正式発表 されましたので、さっそく利用方法を投稿してみました。

来週の技術セッションではさらに詳しい説明があると思いますので、ご登録がまだの方は、ぜひご登録ください! (ご登録いただくとオンデマンド配信でいつでもセッションのビデオを視聴できます。機械翻訳ではありますがビ デオには日本語字幕が付いています。)

この記事では、以下の操作方法をご紹介します。

- 1. ObjectScript用エクステンションのインストール方法
- 2. <u>サーバへ接続する</u>
- 3. <u>クラス定義を作ってみる</u>
- 4. <u>ルーチンを作ってみる</u>
- 5. <u>デバッグを実行してみる</u>
- 《2024/1/17:更新》最新のObjectScriptエクステンションに合わせて内容を更新しました。

 6. Webサーバ使用時:デバッグの注意点
 《2024/1/17追加》プライベートWebサーバではない通常のWeb サーバを使用する場合の注意点を追加しました。

操作前の準備

<u>VSCodeのインストール</u>を行います。

インストール後、VSCode で作成するクラス定義(*.cls)やルーチン(*.mac や *.inc)用ファイルを配置するためのワークスペース(作業環境)用フォルダを作成します。作成が終わったら VSCode を開き、[File > Open Folder...]から作成したフォルダを選択します。

File Edit Selection View	Go Run 🔀 File Edit Selection V	ew Go Run ··· Welcome - V	/STest - Visual Studi — 🗆 🗙
New File	Ctrl+N Ctrl+N	• 🔀 Welcome 🛛 🗙	
New Window Ctrl+: Open File Open Folder Open Workspace Open Workspace	-Shift+N Ctrl+O Ctrl+O ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓ ↓	Start New file Open folder Add workspace folder	Customize Tools and languages Install support for JavaScript, Pyt Settings and keybindings
	8	Recent FirstStep-Inteoperability C:\docker	install the settings and keyboard
	.≉ ⊕	OUTPUT ···· Tasks	× ^ ① A ≖ ∨
	> NPM SCRIPTS		
	> DOCKER CONTAINERS		
	(8) > DOCKER IMAGES		
	> AZURE CONTAINER REG	S	
	203 DOCKER HUB		
	> SUGGESTED DOCKER HU	B	
			& Ų

《2024/1/18**追記》**

г

InterSystems製品のバージョン2023.1以下を利 用されている環境で**プライベート**Web**サーバ**()**ではなく** Webサーバ(IISやApache)を利用してVSCodeから接続する場合、Webサーバにパスを追加する必要があります。

詳細は

<u>Webサーバ(IISやApache)を利用してVSCodeからInterSystemsサーバに接続するときの注意点(バージョン202</u> 3.1以下)」をご参照ください。

()

InterSystems製品インストール時、簡易的にインストールされたApacheで52773番ポートでアクセスするWebサーバ

1、<u>ObjectScript用エクステンションのインストール</u>

エクステンションのアイコン



をクリックし「InterSystems」で検索し

をインストールします。







のアイコンが左端に表示されます。

2、<u>サーバへ接続する</u>

《2024/2/1現在のバージョンに合わせて修正しています》

メニューから File > Preferences > Settings (macOS では Code > Preferences > Settings) を選択して、VS Code Settings Editor を開きます。

検索欄に、objectscript を記入し、検索欄すぐ下にある [Workspace] を選択します。

Extensions > InterSystems ObjectScript を選択し、画面右に表示される設定項目の中から「ObjectScript: Conn」にある「Edit in settings.json」をクリックします。

X (File Edit Selection View Go Run …	$\leftarrow \rightarrow$	P TryIRIS [Administrator]	
Ŋ	Settings objectscriptで検索			
ρ	objectscript			
ço So	User Workspace			
~	Text Editor (2)	cuk		
\$ ^{>}	 Workbench (1) Editor Management (1) 	Objectscript: Compile On Say		
₽₽	✓ Extensions (37)	\checkmark Automatically compile an	InterSystems file when saved in the editor.	
ш	InterSystems Language Server (7)			
Q	intersystems Objectscript (50)	Objectscript: Conn		
		Server Access		
		Edit in settings.json		
		Objectscript > Debug: Copy T	o Clinboard	
		Show inline Copy Invocat	ion CodeLens action for ClassMethods and	Routine Labels.
		Obiostessint - Dobusy Dobus	This Mathad	

開いたワークスペースのフォルダ以下に [.vscode] フォルダが用意され、settings.json が作成されます(このファイルは、ワークスペースに対するサーバ接続情報を記述できる設定ファイルです)。

settings.json を開くと、"intersystems.servers":{} と、"objectscript.conn {} の記述があり、"intersystems.servers":{} に接続したいサーバ情報を名前を付けて記述し "objectscript.conn {} には "intersystems.servers":{} で名付けたサーバ名を使用して接続指定を行います。

接続先を簡単に切り替えることができるように、"intersystems.servers":{} には、複数の接続先サーバ情報を記述できます。

以下、2つの接続先情報(test1、test2)を指定した設定例です。



画面例では、test1の接続情報を使用して接続します。

{

}

```
接続したい IRIS のホスト名 (host)、Webサーバポート番号 (port)、httpかhttpsか (schema)、接続時
 に使用するユーザ名 (username)、接続サーバ情報 (server)、ネームスペース (ns)、"active": true
 を設定し保存します(Ctrl+sで保存できます)。
 コ
 ミュ
 ニティエ
 ディションを除くプライベートWebサーバを使用しない2023.2以降のバージョンでは、"pathPrefix":
 "/インスタンス名"
 の指定も必要となります。詳しく
 はこちらの記事をご参照ください。
 プライベートWebサーバを使用しないバージョンの新規インストール / アップグレード:概要]
"intersystems.servers": {
   "test1":{
       "webServer": {
          "host": "127.0.0.1",
          "port": 52773,
          "scheme": "http"
       },
       "username": "SuperUser"
   },
   "test2":{
       "webServer": {
          "host": "127.0.0.1",
          "pathPrefix": "/iris",
          "port": 80,
          "scheme": "http"
       },
       "username": "SuperUser"
   }
},
"objectscript.conn": {
   "active": true,
   "server": "test1",
   "ns": "USER"
}
```

なお、以下のように、"objectscript.conn {}の設定だけでも接続できますが、接続ユーザ名に対するパスワードを settings.jsonに記述しておく方法となります。

```
{
    "intersystems.servers": {},
    "objectscript.conn": {
        "active": true,
        "host": "localhost",
        "port": 52773,
        "ns": "USER",
        "username": "_SYSTEM",
        "password": "SYS"
```

接続できると、VSCode左下に以下のように接続情報が表示されます。



接続までの手順は以下の通りです。

}

}



クリック後、画面左下に表示される[ObjectScript](下図)をクリックし、画面上部中央に表示されるメニューから「Toggle Coonection」をクリックし、接続できるかご確認ください。



∢	File Edit Selection	
Ŋ	INTERSYSTEMS	Toggle Connection Enable current connection
Q	✓ ☆ Current > 鳥 test1	2 Intersystems.servers : 1 3 ···· "test1":{

接続が完了すると、下図のように各種メニューを表示できます。

≺	File Edit Selection	Select action for server: 127.0.0.1:52773[USER]
ζŊ	INTERSYSTEMS	Toggle Connection
	✓ SERVERS	Disable current connection
Q	〜 放 Current	Refresh Connection
	> 且 test1	Force attempt to connect to the server
የօ	> 🕲 Recent	Switch Namespace
۶.	> 📙 All Servers	Switch to a different namespace in the current server
		Open Management Portal
₿ [∕]		http://127.0.0.1:52773/csp/sys/UtilHome.csp?\$NAMESPACE=USER
		Open Class Reference
Ш	✓ , TryIRIS (127.0)	http://127.0.0.1:52773/csp/documatic/%25CSP.Documatic.cls?LIBRARY=USER
_	> 😫 Classes	on Studio Add in
Ä	> 🖃 Routines	11 "test2":{ 12 "webServer": {
	> 🔊 Includes	13 "host": "127.0.0.1".
n	> CSP Files	14 "pathPrefix": "/iris",
`	> 🔂 Other	15 "port": 80,
		16 "scheme": "http"
		17 },
	◆ 画 test1.05EK	18 "username": "SuperUser"
(8)		技術 小 悠 ど 唯 認 じ ざ ま 9 。 クリック す る と 画 面 上 部 に メニュー く
-00		が表示されます。
503		23 "server": "test1",
× (\otimes 0 \wedge 0 (W) 0 test 10	USERI In 24 Col 21 Spaces: 4 LITE-8 LE {} ISON with Comments
/		

メニューの [Refresh Connection] をクリックすると最新情報を取得できます。



をクリックすると、サーバ内のクラス / ルーチン / インクルードファイルなどの情報を確認できます。(サーバ 側編集モードを有効にしない限り、サーバ側コードは read-only で開きます)

×	File Edit Selection View	\leftrightarrow \leftrightarrow \rightarrow \mathcal{P} TryIRIS [Administrator]	- 🗆 ×		
Ch	INTERSYSTEMS ····	settings.json InboundAdapter.cli (read-only)			
	> SERVERS <pre></pre>	 /* Copyright (c) 2023 by InterSystems Corporation. Cambridge, Massachusetts, U.S.A. All rights reserved. Confidential property of InterSystems Corporation. */ 	A State Address of the Address of Addre		
م م	Operation.cls SAMLGenericS Service.cls	 4 5 /// SQL database polling client adapter. Repeatedly executes a query against 6 Class EnsLib.SQL.InboundAdapter Extends (Ens.InboundAdapter, EnsLib.SQL.Comm 			
¦. ⊳⊓	> SAP > Siebel	8 9 Parameter DOMAIN = "ENSEMBLE";	VECUCOUNTS		
щ	> SOAP > SQL	<pre>10 11 Parameter SERVICEINPUTCLASS = "EnsLib.SQL.Snapshot"; 12</pre>	Construction of the second sec		
A	 Service Common.cls 	<pre>13 /// The Base query string that will be repeatedly executed to look for new r 14 Property Query As %String(MAXLEN = "") [Required]; 15</pre>	*OW		
ų	CommonJ.cls GatewayResult	16 /// A comma-separated list of parameter value specifiers, if any. These corr 17 /// % <varname> refers to Adapter properties such as %LastKey;</varname>	*es !!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!!		
	InboundAdapt InboundProcA OutboundAda	 /// \$<varname> refers to Service class properties;</varname> /// &<varname> refers to persistent values saved between invocations.</varname> Property Parameters As %String(MAXLEN = 1000); 	Control of the second s		
0	Snapshot.cls TCP Talact	21 22 /// A comma-separated list of the SQL data types of each parameter used in t 23 /// This may be required for the query to execute successfully if the dataty 24 December 20 Types 40 % Chains (MAYLEN)	:he /pe		
0 503	Testing TN3270 PROJECTS	<pre>24 Property Paramoglypes As Astring(MAXLEN =); 25 26 /// If specified, used to delete rows after processing. Must have exactly or 27 Property DeleteOuery As %String(MAXLEN = "");</pre>	ıe		
×	× ⊗ 0 ∆ 0 № 0 test1[USER] Ln 1, Col 1 Tab Size: 4 ObjectScript Class Q				

サーバ情報は読み取り専用の為、VSCode 上で編集したい場合は、ソースコードを右クリックし「Export」を選択します。

Export選択後、ワークスペース > src 以下にソースコードがコピーされます。

システムコードは上書きしないようにご注意ください。

Export先ディレクトリは "objectscript.export": {} の設定で変更できます。デフォルトは src が設定されています。

以下のように "folder" の設定を使用して別ディレクトリに変更することができます。

```
"objectscript.export": {
    "folder": "mydir",
    "addCategory": false,
    "map": {},
    "atelier": true,
    "generated": false,
    "filter": "",
    "exactFilter": "",
    "category": "*",
    "noStorage": false,
    "dontExportIfNoChanges": false,
    "maxConcurrentConnections": 0,
    "mapped": true
}
```

3、<u>クラス定義を作ってみる</u>

早速、クラス定義を作成し、HelloWorld!の文字列を出力するメソッドを作ります。



をクリックし、ワークスペースに戻ります。

開いたワークスペースに、パッケージ名と同名のフォルダを用意します。例では、Test としています。



定義例)

```
Class Test.Class1 Extends %RegisteredObject
{
ClassMethod Hello()
{
    write "Hello world! ??????"
}
```

[Ctrl + スペース]を押すと以下のような入力候補が出ます。



コードの作成が完了したら、Ctrl + s で保存 + コンパイルを行います。(Outputウィンドウを表示しておくとコンパイル結果が確認できます。)

≮	File Edit Selection View Go Run Terminal He	p Class1.cls - VSTest - Visual Studio Code	- 1	٥	\times
ſħ	EXPLORER	Class1.cls × {} settings.json			
S	> OPEN EDITORS	vscode > Test > Class1.ds > Class1			
	> OPEN EDITORS ✓ VSTEST ✓ .vscode ✓ Test ● Class1.cls {} settings.json	<pre>vscode > Test > ● Class1.ds > ♥\$ TestClass1 1 Class Test.Class1 2 { 3 Debug this method 4 ClassMethod Hello() 5 { 6 write "Hello world! こんにちは!" 7 } 8 9 } 10</pre>			
8	NPM SCRIPTS DOCKER CONTAINERS DOCKER IMAGES AZURE CONTAINER REGISTRY	PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL ObjectScript マ 10/21/2020 23:42:04 に修飾子 'cuk' でコンパイルを開始しました。, 個のワーカー 使用 クラスのコンパイル中 Test.Class1 ルーチンのコンパイル中 Test.Class1.1	6 1 ・・ジョ	^ ヨブ?	× を
2 23	DOCKER HUB SUGGESTED DOCKER HUB IMAGES	コンハイルか止常に終了しました(所要時間: 0.027秒)。			

ターミナルを起動し、クラスメソッドの実行を試します。

USER>do ##class(Test.Class1).Hello()
Hello world! ??????
USER>

メモ:コンテナやLinux上のIRISに対しては、iris session インスタンス名 でIRISにログインできます。

例)

irisowner@0c6e3bc48ff6:~\$ iris session IRIS
???: 0c6e3bc48ff6 ?????: IRIS

USER>

4、<u>ルーチンを作ってみる</u>

ルーチンの作成についての注意事項は、<u>橋本さんに投稿いただいたこちらの記事</u> がとても参考になります。ぜひご参照ください。

ワークスペース以下にルーチン用ファイルを作成します。拡張子は mac とします。

ルーチンでも [Ctrl + スペース] を押すと入力候補が表示されます。

ルーチン実行例は以下の通りです。

```
USER>do hello<sup>*</sup>First()
??????????
USER>
```


<u>《2024/1/17時点の操作方法に合わせ図解など更新しています》</u>

クラス定義の場合、以下のようにメソッド定義の直前に[Debug]と表示されます。[Debug] をクリックするとデバッグが開始するので、ブレークポイント(F9)を事前に設定しておきます。

4	///
	Debug Copy Invocation
5	ClassMethod Hello()
6	{
• 7	write "Hello world! こんにちは!"
8	set sum=1+2
9	write sum,!
10	}

8 ID

[Debug]をクリックすると、VSCodeの画面上部にデバッグ用アイコン

が表示されます。

ステップ実行した時の状態は以下の通りです。(画面左側に実行中の変数などの一覧が表示されます)

ルーチンの場合もラベル名の上に表示される[Debug] をクリックすることで同様の手順でデバッグを開始できます。 VSCode **を使ってみよう!**

Published on InterSystems Developer Community (https://community.intersystems.com)

6、Webサーバ使用時:デバッグの注意点

プライベートWebサーバ(=InterSystems製品インストール時、簡易的にインストールされたApacheで52773 番ポートでアクセスするWebサーバ)使用時は以下の変更は不要です。

Webサーバ(IISやApache)を利用している環境

でVSCodeからObjectScriptのデバッグを実行する際、Webソケットモジュールがデフォルトでは有効になっていないため以下のようなエラーが表示されます。

Visual S	tudio Code	Х
8	Failed to start the debug session. Check that the InterSystems server's web server supports WebSockets.	
	Open 'launch.json' Cancel	

(WebサーバのWebソケットモジュールが有効になっていないため、デバッグセッションの開始が失敗した旨の エラーが表示されます)

以下、IISとApacheでのWebソケットモジュールの有効化方法をご紹介します。

<u>||Sの場合</u>

Webサーバ(IIS)のサービスを選択した状態で、「アプリケーション開発」以下にある「WebSocketプロトコル」をチェックします。

また、「HTTP共通機能」以下にある「HTTPリダイレクト」もチェックします。

Windows10での追加の場合は下図の通りです。

💽 Windows の機能	_		×
Windows の機能の有効化または無効化			?
機能を有効にするには、チェック ボックスをオンにしてください。機能を無効にするには、チェック ボックスをオフにしてくださ スは、機能の一部が有効になっていることを表します。	もい。 塗りつぶされ	たチェックオ	ボック
 □ 1/2 - ネット インフォメーション サービス ■ FTP サーバー ■ Web 管理ツール ■ HTTP 共通機能 □ HTTP 共通機能 □ HTTP リダイレクト □ WebDAV 発行 □ ディレクトリの参照 ○ ディレクトリの参照 ○ ディレクトリの参照 ○ アブリケーション開発機能 □ NET Extensibility 4.8 □ NET 拡張機能 3.5 □ Application Initialization □ ASP □ ASP.NET 3.5 □ ASP.NET 4.8 □ GGI □ ISAPI 27/1/2- 			
レー ISAPI 拡張 WebSocket プロトコル サーバー側イングルード			
			*
	ОК	キャンセ	211

<u>Apacheの場合</u>

WebSocketモジュールを有効化するため、以下実行します。(設定変更後、Apacheを再起動してください。)

\$ sudo a2enmod proxy_wstunnel Considering dependency proxy for proxy_wstunnel: Enabling module proxy. Enabling module proxy_wstunnel. To activate the new configuration, you need to run: systemctl restart apache2 \$ sudo systemctl restart apache2 \$

<u>#初心者</u> #InterSystems IRIS #InterSystems IRIS for Health #VSCode

ソースURL:

https://jp.community.intersystems.com/post/vscode-%E3%82%92%E4%BD%BF%E3%81%A3%E3%81%A6%E3%81%A6%E3%81%BF%E3%82%88%E3%81%86%EF%BC%81