

記事

[Mihoko Iijima](#) · 2023年2月14日 4m read

irissqlcli (IRIS SQLの高度なターミナル) : InterSystemsデベロッパーツールコンテスト2023入賞作品のご紹介

開発者の皆さん、こんにちは！

InterSystems デベロッパーツールコンテスト2023が開催され、21の応募作品の中から勝者が発表されました

この記事では、世界のIRIS開発者の皆さんから注目を集めた作品をご紹介します。

最初は、Experts Nomination 第1位に輝いた [@Dmitry Maslennikov](#) さんの作品をご紹介します！

[@Dmitry Maslennikov](#) さんが解説されている [記事 \(Welcome irissqlcli - advanced terminal for IRIS SQL\)](#) もあります。ぜひこちらもご覧ください。

[@Dmitry Maslennikov](#) さんは、IRIS SQL用の高度なターミナル [irissqlcli](#) を開発されました。

[irissqlcli](#)

を使用すると、SQL記述時にSQL構文、関数、型、IRIS内テーブル名、カラム名に対する候補が表示されるため、SQL文がとても書きやすくなります。

ヘルプも充実しています。(hでヘルプが表示されます)

```
[SQL]_SYSTEM@host.docker.internal:USER> \?  
+-----+  
| Command | Shortcut | Description |  
+-----+  
| .exit   | \q      | Exit. |  
| .mode   | \T      | Change the table format used to output results. |  
| .once   | \o [-o] filename | Append next result to an output file (overwrite using -o). |  
| .schemas | \ds [schema] | List schemas. |  
| .tables | \dt [schema] | List tables. |  
| \e      | \e      | Edit command with editor (uses $EDITOR). |  
| help    | \?      | Show this help. |  
| nopager | \n      | Disable pager, print to stdout. |  
| notee   | notee   | Stop writing results to an output file. |  
| pager   | \P [command] | Set PAGER. Print the query results via PAGER. |  
| prompt  | \R      | Change prompt format. |  
| quit    | \q      | Quit. |  
| tcommit | \tc     | Commit a Database Transaction. |  
| tee     | tee [-o] filename | Append all results to an output file (overwrite using -o). |  
| trollback | \tr    | Rollback a Database Transaction. |  
| tstart  | \ts     | Start a Database Transaction. |  
+-----+  
Time: 0.005s  
[SQL]_SYSTEM@host.docker.internal:USER> .tables  
+-----+  
| TABLE_NAME |  
+-----+  
| MyHealth.Steps |  
+-----+  
Time: 0.028s  
[SQL]_SYSTEM@host.docker.internal:USER> |
```

接続先のテーブル一覧を取得する場合は「.tables」で取得できました。

また、以下のように記入時に入力候補が表示されます。Pygments を利用されているようで、シンタックスがハイライトされてきれいです。

```
[SQL]SuperUser@localhost:USER> select * From "MyHealth"."Steps"  
"Steps"
```

```
[SQL]_SYSTEM@host.docker.internal:USER> select TO_DATE('YYYY-MM-DD  
TO_CHAR  
TO_DATE  
TO_NUMBER  
[F3] Multiline: OFF
```

[irissqlcli](#) の利用方法は3種類ありますが、この記事では、2種類をご紹介します。

その1 : pipでirissqlcliをインストールして使う方法

Pythonがインストールされている環境で、以下実行します。

```
pip install -U irissqlcli
```

後は、irissqlcliコマンドを利用して、お手元のIRISに接続するだけです。

接続情報の指定は以下の通りです。

irissqlcli iris://IRISのユーザ名:パスワード@ホスト名:スーパーサーバポート/ネームスペース名

IRISのユーザ名がSuperUserで、パスワードがSYS、localhostの1972のスーパーサーバポートのIRISのUSERネームスペースに接続する方法は以下の通りです。

```
irissqlcli iris://SuperUser:SYS@localhost:1972
```

実際の画面は以下の通りです。(exitを入力すると元のプロンプトに戻りました)

```
● (dev01) PS C:\WorkSpace\Contest> irissqlcli iris://SuperUser:SYS@localhost:1972
Server: InterSystems IRIS Version 2023.1.0.185 xDBC Protocol Version 65
Version: 0.4.0
[SQL]SuperUser@localhost:USER> exit
Goodbye!
○ (dev01) PS C:\WorkSpace\Contest> █
```

その2 : irissqlcli-web用コンテナを利用する方法

Webターミナルで実行できるコンテナが提供されているので、dockerをインストールした環境で以下実行します。

```
docker run -d --name irissqlcli --restart always -p 7681:7681 caretdev/irissqlcli-web irissqlcli iris://_SYSTEM:SYS@host.docker.internal:1972/USER
```

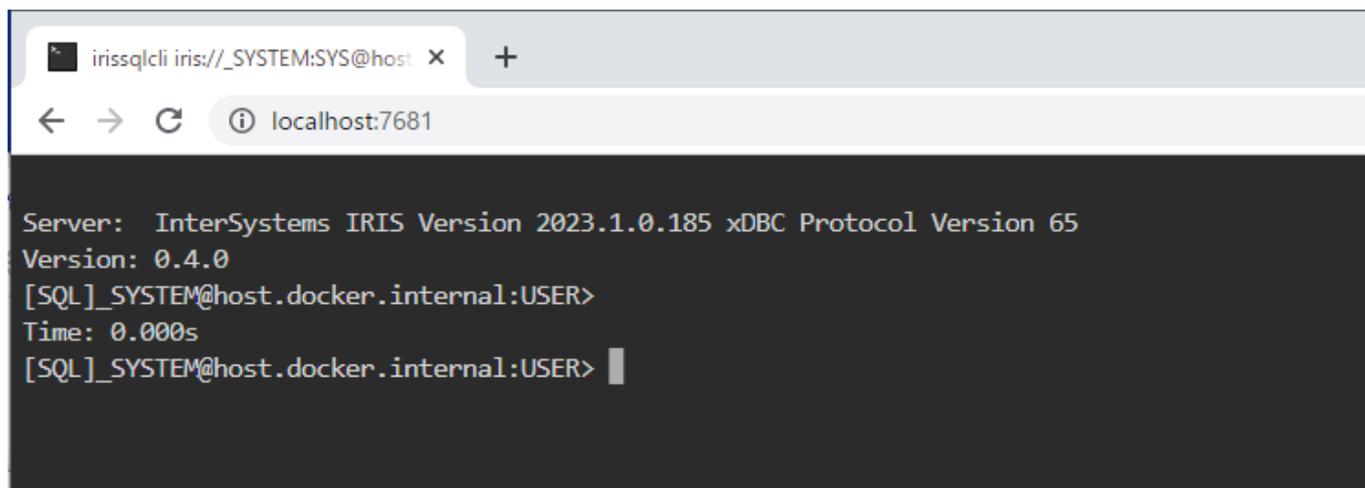
上記実行で、caretdev/irissqlcli-web:latestのイメージがPullされます。(イメージのサイズは157MBでした。とても小さいです!)

イメージのPullが完了するとコンテナが開始され、コンテナのホストにインストールされているIRISのスーパーサーバポート1972番、USERネームスペースにアクセスします。

コンテナが正常に開始されたかについては、docker ps で確認できます。

```
>docker ps
CONTAINER ID   IMAGE                                COMMAND                  CREATED
STATUS        PORTS                               NAMES
40f556b8fdd8  caretdev/irissqlcli-web            "/usr/bin/tini -- tt..." 3 seconds ago   Up 1 second    0.0.0.0:7681->7681/tcp      irissqlcli
```

localhost:7681 にアクセスすると、以下のように irissqlcliのWebターミナル版が開きます。



```
irissqlcli iris://_SYSTEM:SYS@host x +
localhost:7681
Server: InterSystems IRIS Version 2023.1.0.185 xDBC Protocol Version 65
Version: 0.4.0
[SQL]_SYSTEM@host.docker.internal:USER>
Time: 0.000s
[SQL]_SYSTEM@host.docker.internal:USER>
```

コンテナを停止する場合は、以下の実行で停止できます。

```
docker stop irissqlcli
```

コンテナを削除したい場合は、以下の実行で削除できます。

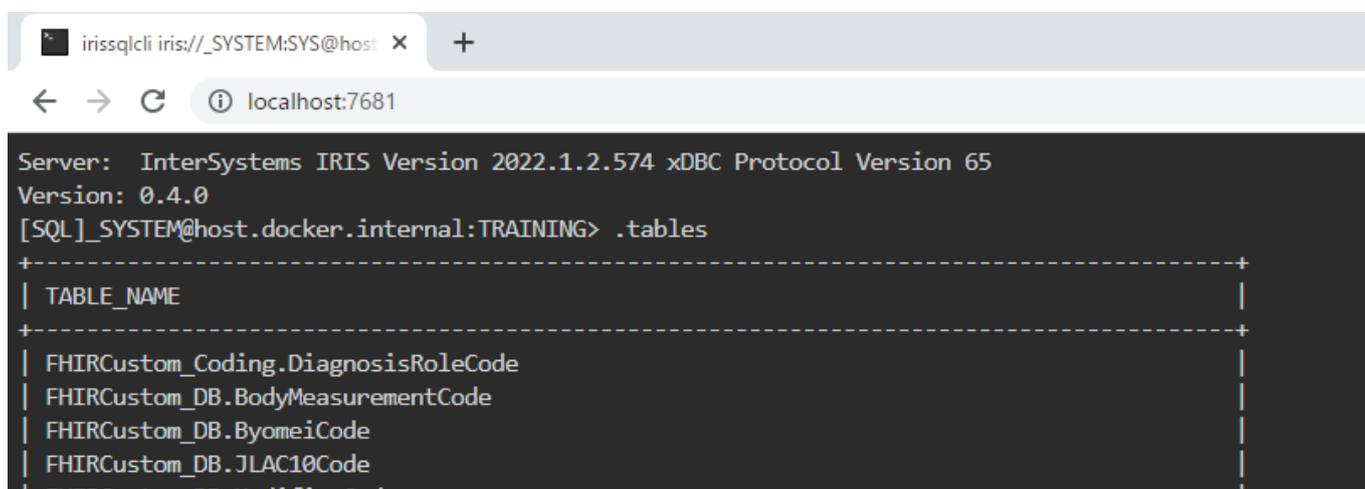
```
docker rm irissqlcli
```

接続するIRISのスーパーサーバーポートや名前空間を変更したい場合は、docker runで指定している末尾の引数 (irissqlcli以降) の接続情報を変更するだけで切り替えができます。

以下の実行例では、スーパーサーバーポート : 51778、名前空間 : TRAININGに接続しています。

```
>docker run -d --name irissqlcli --restart always -p 7681:7681 caretdev/irissqlcli-
web irissqlcli iris://_SYSTEM:SYS@host.docker.internal:51778/training
fd5523695e10f75b0b1157cf001741fb5ddfae5f74b4f58a5a960f96f88f6006
```

プロンプトの右端に指定した名前空間が表示されます。



```
irissqlcli iris://_SYSTEM:SYS@host x +
localhost:7681
Server: InterSystems IRIS Version 2022.1.2.574 xDBC Protocol Version 65
Version: 0.4.0
[SQL]_SYSTEM@host.docker.internal:TRAINING> .tables
+-----+
| TABLE_NAME |
+-----+
| FHIRCcustom_Coding.DiagnosisRoleCode |
| FHIRCcustom_DB.BodyMeasurementCode |
| FHIRCcustom_DB.ByomeiCode |
| FHIRCcustom_DB.JLAC10Code |
| FHIRCcustom_DB.ModifierCode |
```

どちらの方法も使用中IRISに追加のインストールが不要ですので、お手軽にご利用いただけると思います



ぜひ皆さんのIRISでも試してみてください。また、返信欄に使った感想など、ぜひお寄せください！

[#SQL](#) [#ツール](#) [#InterSystems IRIS](#) [#InterSystems IRIS for Health](#) [#IRIS contest](#)

ソースURL:<https://jp.community.intersystems.com/post/irissqlcli%EF%BC%88iris-sql%E3%81%AE%E9%AB%98%E5%BA%A6%E3%81%AA%E3%82%BF%E3%83%BC%E3%83%9F%E3%83%8A%E3%83%AB%EF%BC%89%EF%BC%9Aintersystems%E3%83%87%E3%83%99%E3%83%AD%E3%83%83%E3%83%91%E3%83%BC%E3%83%84%E3%83%BC%E3%83%AB%E3%82%B3%E3%83%B3%E3%83%86%E3%82%B9%E3%83%882023%E5%85%A5%E8%B3%9E%E4%BD%9C%E5%93%81%E3%81%AE%E3%81%94%E7%B4%B9%E4%BB%8B>