



InterSystems SQL の基礎： グローバルおよびマルチモデ ル・データ・アクセス

Version 2023.1

2024-01-02

InterSystems SQL の基礎 : グローバルおよびマルチモデル・データ・アクセス
InterSystems IRIS Data Platform Version 2023.1 2024-01-02
Copyright © 2024 InterSystems Corporation
All rights reserved.

InterSystems®, HealthShare Care Community®, HealthShare Unified Care Record®, IntegratedML®, InterSystems Caché®, InterSystems Ensemble®, InterSystems HealthShare®, InterSystems IRIS®, および TrakCare は、InterSystems Corporation の登録商標です。HealthShare® CMS Solution Pack™ HealthShare® Health Connect Cloud™, InterSystems IRIS for Health™, InterSystems Supply Chain Orchestrator™, および InterSystems TotalView™ For Asset Management は、InterSystems Corporation の商標です。TrakCare は、オーストラリアおよび EU における登録商標です。

ここで使われている他の全てのブランドまたは製品名は、各社および各組織の商標または登録商標です。

このドキュメントは、インターシステムズ社(住所: One Memorial Drive, Cambridge, MA 02142)あるいはその子会社が所有する企業秘密および秘密情報を含んでおり、インターシステムズ社の製品を稼働および維持するためにのみ提供される。この発行物のいかなる部分も他の目的のために使用してはならない。また、インターシステムズ社の書面による事前の同意がない限り、本発行物を、いかなる形式、いかなる手段で、その全てまたは一部を、再発行、複製、開示、送付、検索可能なシステムへの保存、あるいは人またはコンピュータ言語への翻訳はしてはならない。

かかるプログラムと関連ドキュメントについて書かれているインターシステムズ社の標準ライセンス契約に記載されている範囲を除き、ここに記載された本ドキュメントとソフトウェアプログラムの複製、使用、廃棄は禁じられている。インターシステムズ社は、ソフトウェアライセンス契約に記載されている事項以外にかかるソフトウェアプログラムに関する説明と保証をするものではない。さらに、かかるソフトウェアに関する、あるいはかかるソフトウェアの使用から起こるいかなる損失、損害に対するインターシステムズ社の責任は、ソフトウェアライセンス契約にある事項に制限される。

前述は、そのコンピュータソフトウェアの使用およびそれによって起こるインターシステムズ社の責任の範囲、制限に関する一般的な概略である。完全な参照情報は、インターシステムズ社の標準ライセンス契約に記載され、そのコピーは要望によって入手することができる。

インターシステムズ社は、本ドキュメントにある誤りに対する責任を放棄する。また、インターシステムズ社は、独自の裁量にて事前通知なしに、本ドキュメントに記載された製品および実行に対する代替と変更を行う権利を有する。

インターシステムズ社の製品に関するサポートやご質問は、以下にお問い合わせください:

InterSystems Worldwide Response Center (WRC)
Tel: +1-617-621-0700
Tel: +44 (0) 844 854 2917
Email: support@InterSystems.com

目次

InterSystems SQL の基礎 : グローバルおよびマルチモデル・データ・アクセス.....	1
1 グローバルとは	1
2 アプリケーション開発者がグローバルについて学ぶべき理由	2
3 InterSystems IRIS API を使用したグローバルへのアクセス	2
4 グローバルの詳細	2
4.1 グローバルおよびその構造	2
4.2 InterSystems Native SDK	2

InterSystems SQL の基礎：グローバルおよびマルチモデル・データ・アクセス

ここでは、InterSystems IRIS® データ・プラットフォームの基盤となっているストレージ構造であるグローバルの概念について紹介します。データの格納やデータへのアクセスをどのように行うことにしたかには関係なく、実行していることはグローバルの使用です。

グローバルは、リレーショナル・モデルを使用してアクセスするか、オブジェクト・モデルを使用してアクセスするか、または直接アクセスすることができます。このマルチモデル・アクセスの利点を説明するビデオ、および3つの代替方法を自分で試すことのできる実践演習については、“[Exploring Multiple Data Models with Globals](#)”を参照してください。

1 グローバルとは

InterSystems IRIS データ・プラットフォームの特徴の1つが、一度データを保存すると、複数のパラダイムを使用してそのデータにアクセスできるようになる機能です。例えば、InterSystems SQLを使用して、データを行と列で視覚化したり、ObjectScript を使用して、プロパティとメソッドを持つオブジェクトの観点からデータを考えることができます。アプリケーションでは、両方のデータ・モデルを組み合わせ、指定されたタスクに対して、より簡単で効果的な方のモデルを使用することも可能です。しかしながら、データをどのように書き込むか、またはデータにどのようにアクセスするかに関係なく、InterSystems IRIS では、グローバルとして知られる基盤となるデータ構造にデータが格納されます。

グローバルは、永続多次元スパース配列と考えることができます。

- ・ 永続グローバルは、データベースに格納され、そのデータベースにアクセスできる任意のプロセスにより、いつでも取り出すことができます。
- ・ 多次元グローバルのノードには、任意の数の添え字を指定できます。こういった添え字には、整数、10進数、または文字列を使用できます。
- ・ スパースノードの添え字は、連続する必要はありません。つまり、格納された値のない添え字はストレージを使用しません。

グローバルのノードには、以下のようなさまざまな種類のデータを格納できます。

- ・ 文字列
- ・ 数値データ
- ・ 文字ストリームまたはバイナリ・データ
- ・ リストや配列など、データのコレクション
- ・ 他のストレージの場所への参照

記述したサーバ側コードさえも最終的にはグローバルに格納されます。

2 アプリケーション開発者がグローバルについて学ぶべき理由

グローバルに関する知識をほとんど、あるいはまったく持たずに InterSystems IRIS プラットフォームでアプリケーションを作成することはできますが、グローバルに対する理解を深めることをお勧めする理由がいくつかあります。

- ・ 操作によっては、グローバルに直接アクセスすることで、より簡単に、あるいは効率的になるものがあります。
- ・ リレーショナル・データ・モデルまたはオブジェクト・データ・モデルに適合しないデータのカスタム・データ構造を作成できます。
- ・ システム管理タスクによっては、グローバル・レベルで実行されるものがあり、グローバルを理解することで、こういったタスクがより有意義なものになります。

3 InterSystems IRIS API を使用したグローバルへのアクセス

Python、Java、.NET 言語、Node.js、または C++ でアプリケーションを作成している場合、InterSystems IRIS には、ここで説明している 3 つのモデルを使用してデータを操作できるようにする API が用意されています。

- ・ [JDBC](#)、[ADO.NET](#)、DB-API、または ODBC を使用したリレーショナル・アクセス
- ・ InterSystems XEP API for Java および InterSystems XEP API for .Net を使用したオブジェクト・アクセス
- ・ InterSystems Native SDK を使用したグローバルへの直接アクセス

注釈 あらゆるアクセス形式がすべての言語でサポートされているわけではありません。

4 グローバルの詳細

以下にリストしたリソースを使用して、グローバルおよびグローバルへのアクセス方法について知識を深めてください。

4.1 グローバルおよびその構造

- ・ [Exploring Multiple Data Models with Globals](#) – ビデオと実践演習
- ・ [グローバルの使用方法](#) – グローバルの構造、グローバルを管理する方法、および SQL または ObjectScript を使用してグローバルにアクセスする方法について説明します。
- ・ [グローバル – “クラスの定義と使用”](#) のこのセクションでは、%Persistent クラスを拡張するクラスに作成したグローバルの命名規則について説明します。

4.2 InterSystems Native SDK

- ・ [Using the Native SDK for Java](#) (インタラクティブ・コース) – Native SDK for Java を使用して、グローバルにアクセスし、クラス・メソッドとクラス・ルーチン呼び出す方法を紹介しします。
- ・ Native SDK for Java の使用方法 (ドキュメント) – Native SDK for Java を使用して、グローバルにアクセスし、クラス・メソッドとクラス・ルーチン呼び出す方法を紹介しします。

- ・ [Using the Native SDK for .NET](#) (インタラクティブ・コース) – Native SDK for .NET を使用して、グローバルにアクセスし、クラス・メソッドとクラス・ルーチンを呼び出す方法を紹介します。
- ・ [InterSystems Native SDK for .NET の使用法 \(ドキュメント\)](#) – Native SDK for .NET を使用して、グローバルにアクセスし、クラス・メソッドとクラス・ルーチンを呼び出す方法を紹介します。
- ・ [Native SDK for Node.js の使用法 – \(ドキュメント\)](#) Native SDK for Node.js を使用して、グローバルにアクセスし、クラス・メソッドとクラス・ルーチンを呼び出す方法を紹介します。
- ・ [Using the Native SDK for Python](#) (インタラクティブ・コース) – Native SDK for Python を使用して、グローバルにアクセスし、クラス・メソッドとクラス・ルーチンを呼び出す方法を紹介します。
- ・ [Native SDK for Python の使用法 \(ドキュメント\)](#) – Native SDK for Python を使用して、グローバルにアクセスし、クラス・メソッドとクラス・ルーチンを呼び出す方法を紹介します。

