



Caché Basic に関するよく ある質問

Version 5.1

2006-03-14

Caché Basic に関するよくある質問

Caché Version 5.1 2006-03-14

Copyright © 2006 InterSystems Corporation.

All rights reserved.

このドキュメントは、Sun Microsystems、RenderX Inc.、アドビ システムズ および ワールドワイド・ウェブ・コンソーシアム (www.w3c.org) のツールと情報を使用して、Adobe Portable Document Format (PDF) で作成およびフォーマットされました。主要ドキュメント開発ツールは、InterSystemsが構築したCaché と Javaを使用した特別目的のXML処理アプリケーションです。



Caché 製品とロゴは InterSystems Corporation の登録商標です。



Ensemble 製品とロゴは InterSystems Corporation の登録商標です。



InterSystems という名前とロゴは InterSystems Corporation の登録商標です

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動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。

Caché および InterSystems Caché、Caché SQL、Caché ObjectScript および Caché Object は、インターシステムズ社の商標です。

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。

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 ワールドワイド カスタマサポート

Tel: +1 617 621-0700

Fax: +1 617 374-9391

Email: support@InterSystems.com

目次

Caché Basic に関するよくある質問.....	1
一般	1
プログラミング	2
サポート	5
テーブル一覧	
Caché Basic 構文要素	2
Caché Basic とオブジェクトのクイック・リファレンス	3

Caché Basic に関するよくある質問

一般

Caché Basic は Windows プラットフォームでのみ作動するのですか？

いいえ。Caché Basic は完全にプラットフォーム独立で、Caché が実行するすべてのプラットフォームで作動します。Caché Basic は、Caché ObjectScript と同じ方法で Caché カーネルに組み込まれており、Microsoft Scripting Host などのすべての外部スクリプト・エンジンには依存していません。

Caché Basic は Caché ObjectScript よりも遅いのでしょうか？

いいえ。Caché Basic ソースは Caché ObjectScript と同じオブジェクト・コードにコンパイルされているので、Caché Basic の性能は Caché ObjectScript の性能とまったく同じです。

Caché Basic は、Basic のどの種類を基にしているのですか？

Caché Basic の構文は、Microsoft VBScript を基にしています。しかし、Caché Basic はサーバ・サイドのスクリプト言語で、データベースに高度に統合されているので、Microsoft VBScript の実装と Caché Basic の実装には多くの相違点があります。例えば、Caché Basic には MsgBox や InputBox 機能はありませんが、グローバル、リストなどの Caché 特有のデータタイプと直接動作する能力など、多くの拡張機能があります。

Caché Basic を導入したということは、インターシステムズが Caché ObjectScript のサポートを中止する計画があるという意味ですか？

いいえ。Caché Basic が目指しているのは、Caché ObjectScript に取って代わるのではなく、Microsoft Visual Basic などの Basic 実装にすでに精通している開発者が、より容易に Caché を利用できるようにすることです。Caché ObjectScript と Caché Basic は両方ともサポートされ、永久に共存します。

“学習が容易になる”という点以外に、目標としていることはありますか？

はい。Caché Basic の導入により、Caché を使用している企業は、以下のような多くの利点を得ることができるようでしょう。

- ・ 新規開発者の雇用や教育の簡素化

- ・ Caché で開発されたアプリケーションの販売促進。エンド・ユーザは、皆様のアプリケーションを使用し、拡張するために新しい言語を学習する必要はありません。
- ・ Caché での VB や ASP アプリケーションのビジネス・ロジックの再集結。VB クライアント・サイド、または中間層から、高性能と高度なスケーラビリティを実現しながら、既存のアプリケーションのビジネス・ロジックを Caché に移行することができます。CSP と Caché Basic の組み合わせにより、ASP (Active Server Pages) アプリケーションの移行が便利になります。
- ・ 興味深いリソースへのアクセス。VB のソース・コード・アーカイブを参照することで、Caché 開発者は、価値のある興味深い多くのコード・サンプル (ゲームなど) を見ることができます。
<http://www.planet-source-code.com> や <http://www.vbcode.com> などから始めてみるとよいでしょう。

プログラミング

“Caché Basic シェル” はありますか？

いいえ。しかし、独自のものを自由に実装することはできます。Caché Basic シェルの例は、インターシステムズにお問い合わせください。

Caché Basic では、どのようにグローバルを扱うのですか？

以下の構文を使用して、グローバルに直接参照することができます。

Caché Basic 構文要素

構文要素	説明
<code>^Global(index)="Value" Println ^Global(index)</code>	グローバルへの直接アクセス
<code>Exists(^a(1))</code>	値が定義されている場合
<code>Traverse(^a(" "))</code>	次の、または前の添え字に移動
<code>EraseValue</code> 、 <code>EraseArray</code> 、 <code>Merge</code> 、 <code>Lock</code> 、 <code>Unlock</code> 、 <code>Increment()</code>	その他の関数

List や Piece などの Caché 特有の構造は、どのように扱うのですか？

Caché Basic には、List や Piece のサポートに対する、多くの拡張機能があります。以下はその例です。

```
l = ListBuild("blue","red")
Println List(l,l)

p = "blue^red"
noOfItems = Len(p,"^")
Println Piece(p,"^",1)
```

Caché Basic には、\$Order 関数と類似した関数がありますか？

Traverse() 関数が、Caché ObjectScript の \$Order と同じ機能を提供します。

```
i = Traverse(^MyData(""))
While (i <> "")
    Println ^MyData(i)
    i = Traverse(^MyData(i))
Wend
```

Caché Basic では、どのようにオブジェクトを扱うのですか？

Caché Basic とオブジェクトのクイック・リファレンス

Caché Basic 構文	オブジェクト参照
"Basic.Human".ClassMethod()	クラス・メソッドの呼び出し
obj.Report()	インスタンス・メソッド/プロパティ
Me.Name="Anton"	現在のオブジェクト・プロパティ/メソッド
obj=New Basic.Human()	新規オブジェクトの生成
obj=OpenId Basic.Human(1)	オブジェクト・インスタンスのオープン

Caché Basic での、##this.Method() や ..Method() に類似した関数は何でしょうか？

```
Me.Method()
```

Caché Basic では、どのように SQL を扱うのですか？

以下のように、ダイナミック・クエリ・オブジェクトを使用します。

```
result = New %ResultSet()
result.Prepare("SELECT Name,Age FROM Basic.Human WHERE Age<?")
result.Execute(Arg1)

While (result.Next())
    Println result.Data("Name") & ", " & result.Data("Age")
Wend
```

Caché Basic では、どのようにファイルを扱うのですか？

以下のように、%File オブジェクトを使用します。

```
file = New %File("c:\test.txt")
file.Open("WSN")
file.WriteLine("This is a test")
file.Close()
```

Caché Basic では、どのようにエラーをトラップしますか？

以下のように、On Error Goto 文を使用します。

```
Function ErrorTest(Arg1)
  On Error Goto errorhandler
  return 1/Arg1
errorhandler:
  PrintLn "Error ", Err.Number, " ", Err.Description, " ", Err.Source
  Err.Clear
  return 0
End Function
```

Caché Basic では、なぜ <UNDEFINED> エラーが発生しないのでしょうか？

Basic では、各変数は既定で空文字列です。したがって未定義の変数、または関数を参照すると、<UNDEFINED> エラーの代わりに “ ” が返されます。

未定義の変数、または関数を不注意で参照することを防ぐため、Option Explicit 文を使用します。

Caché Basic から Caché ObjectScript プログラムを呼び出したり、またその逆を行うことはできますか？

はい。記述されている言語にかかわらず、メソッドや関数/プロシージャを呼び出すことができます。

Caché ObjectScript では、以下のとおりです。

```
do Procedure^BasicRoutine()
do ##class(MyClass).BasicClassMethod()
```

Caché Basic では、以下のとおりです。

```
Procedure@ObjectScriptRoutine()
"MyClass".ObjectScriptMethod()
```

Caché Basic で SQL トリガを作成することはできますか？

いいえ。Caché 5 ではできません。しかしこの機能は、将来のリリースで提供できるように計画されています。

Caché Basic から、\$ZU 関数を呼び出すことはできますか？

直接、\$ZU 関数を呼び出すことはできません。Caché ObjectScript で記述された複数のクラス・メソッドでラッパ・クラスを生成し、次に Caché Basic で、これらのクラス・メソッドを呼び出します。

なぜ、プロシージャの範囲外でローカルの変数を見ることができないのでしょうか？

これは適切な動作です。Basic 言語が、この機能を定義しています。すべての変数の範囲は、それが定義されたプロシージャ、または関数に制限されています。%myvar など、% で開始する変数名は、例外です。

\$H 形式から、または \$H 形式に日付値と時刻値を変換するには、どのようにすればいいのでしょうか？

利用できる変換関数は、DateConvert()、TimeConvert()、および DateTimeConvert() です。以下はその例です。

```
myDate = "07/15/2002"  
println DateConvert(myDate, vbToInternal)    ' returns $H format  
myHDate = 59000  
println DateConvert(myHDate, vbToExternal)
```

Caché Basic で間接演算を使用することはできますか？

できません。

サポート

Caché Basic でバグを発見しました。どうすればよいのでしょうか？

バグの詳細を、インターシステムズのサポート窓口 までご連絡ください。

Caché Basic に組み込んでほしい機能があります。インターシステムズは、そのような要求に応えていただけるのでしょうか？

Caché Basic は準備が整っていますが、機能の拡張に関しては改善の余地があります。あなたのアイデアをインターシステムズのサポート窓口 までお送りください。送っていただいたアイデアが価値あるものなら、インターシステムズはその機能を将来のバージョンに組み込むように検討します。

このドキュメントに対して意見と提案があります。どうすればよいでしょうか？

ご意見、ご要望は、インターシステムズのサポート窓口 までお送りください。