

HL7 FHIR ハンズオンチュートリアル

会期:2019年9月14日(土)13:00~17:00 会場:兵庫県立大学神戸情報科学キャンパス (兵庫県神戸市)







I. チュートリアルの要点







オブジェクト機能のリレーショナル側からの見え方

<u>https://qiita.com/torisan_/items/af07a05e352e3c63f742</u> にて文法解説中!

Gunma University Hospital SIC, Confidential

3.1.変数記述の基本

宣言(#dim)、大文字小文字の区別(ないものとあるものがあるが、基本的にはあるように記述すべき)

ClassA.cls

```
Class User.Main.ClassA Extends %Persistent
{
   //%Persistentはデータベースへの永続的な保存ができるクラス
Property fielda As %Numeric; //テーブルにfielda列が生成される、という意味でもある
Property fieldb As %String;

ClassMethod Func1() As %Integer
{
    #dim i As %Integer //変数とデータ型の記載の仕方
    i = 3
    QUIT i //戻り値、Rerutn i;と同じ
}
```

Gunma University Hospital SIC, Confidential





ObjectScriptクラス内に、Trigger サブルーチンを記述することで、そのクラス(テーブル)に変更が生じた際の処理を追加できる

```
Trigger UpdateTrigger [ Event = UPDATE, Time = AFTER ]
{
//必要な処理を記述する
}
```

Event = の後ろのキーワード例

DELETE — SQL DELETE 操作の間に起動

INSERT — SQL INSERT 操作の間に起動

UPDATE — SQL UPDATE 操作の間に起動

INSERT/UPDATE — SQL INSERT 操作または SQL UPDATE 操作の間に起動

INSERT/DELETE - SQLの挿入操作または削除操作の間に起動

UPDATE/DELETE — SQLの更新操作または削除操作の間に起動

INSERT/UPDATE/DELETE — SQL INSERT 操作、SQL UPDATE 操作、または SQL DELETE 操

作の間に起動

「IRIS Production」の概要

EnsLib.File.InboundAdapterは、構成された場所でファイルを検索したり、入力を読み取ったり、入力をストリームとして関連するビジネス・サービスに送信したりします。 ユーザが作成および構成するビジネス・サービスは、このストリームを使用してプロダクションの他の部分と通信します。受信ファイル・アダプタが構成された場所に複数の ファイルを見つけた場合は、ファイルの最終変更時刻に基づいて、それが早いものから順番に処理します。ただし、アダプタは、時刻値の小数点以下の秒数は無視します。そ6 ため、2つ以上のファイルの変更日付と時刻が、時刻の小数点以下の秒数部分のみ異なる場合は、アダプタはこれらを任意の順序で処理します。

以下の図は、全体的なフローを示しています。



Gunma University Hospital SIC, Confidential





上中 進太郎 InterSystems Japan

チュートリアル Azure+IRIS開発環境セットアップ

このパートでは、以下の流れに沿ってチュートリアルで使用するサー バサイド環境を構築します。

- I. Azure上でIRIS for Healthを含む仮想マシンを構成
- IRIS for Healthを起動するためのファイル(docker起動ファイルや 評価ライセンス)を取得

3. クライアントで使用するInterSystems IRIS Studioをセットアップ

チュートリアル環境 概要



チュートリアル IRIS for Health開発

このパートでは、以下の流れに沿ってIRIS for Health上でテーブル データをFHIR化するための開発を行います。

- 1. 患者テーブル作成とデータのインポート
- 2. REST ビジネスサービスの開発
- 3. テーブルからデータを取り出すビジネスオペレーションの開発
- チーブルデータをFHIRのPatientリソースへマッピングするデータ 変化の開発

Π

IRIS Interoperability機能(簡易図)



IRISのREST機能の動作の仕組み



Π



IRIS for Health FHIR Resource Repositoryのご紹介

FHIR RESTAPIに準拠した形式で、FHIR Resourceを蓄積し、検索することができる FHIR Resource Repository機能も提供しています。



2. Microsoft Azure の設定手順





Microsoft Azure			P 44-7.	. サービス、ドキュメントの検索	
 ・ ・	すべてのサービス 🔎 はべての検索	99 27			
骨 小ーム	すべて	全般 (16)			
:≡ すべてのサービス	全般	# すべてのリソース	*	④ 最近使った項目	*
 ★ お気に入り 	コンビューティング	💡 サブスクリプション	*	😭 אין ד-געון ד	*
すべてのリソース	ネットワーキンクストレージ	③ 予約	*	Marketplace	*
(*) リソース グループ	Web	♥ サービス正常性	*		ブレビュー ★
ダ クイック スタート センター	モバイル	前若情報	*	🚀 クイック スタート センター	*
Function App	コンテナー	■ 毎料サービス	+		
SOL データベース	データベース		6		

1. Azure 管理画面から Marketplace を起動します

2. 「InterSystems」で検索し、「InterSystems IRIS for Health」を選択します。

現在「InterSystems IRIS for Health Community Edition」は利用できません。

右側の同じ名称の「InterSystems IRIS for Health」は選択しないでください。

$\pi - L > Marketplace$	π - L > Marketplace							
Marketplace								
保存リスト	⊘ InterSystems	×	料金:すべて) (Operating System :	すべて) (公開元:すべて)				
最近作成								
サービス プロバイダー	結果をすべて表示							
カテゴリ								
Get Started	unijum 👬	<u></u>	[#]	[#]				
Al + Machine Learning	InterSystems IRIS Data Platform	InterSystems IRIS for Health	InterSystems IRIS Evaluation Edition	InterSystems IRIS Community Edition	InterSystems IRIS Express Edition	InterSystems IRIS for Health	InterSystems IRIS for Health Community Edition	
分析	InterSystems Corporation	InterSystems Corporation	InterSystems Corporation	InterSystems Corporation	InterSystems Corporation	MerSystems Corporation	InterSystems Corporation	
ブロックチェーン	Complete, cloud-based data platform for SQL and NoSQL DBMS.	Complete, Cloud-based Data Platform for Healthcare	This special version of interSystems IRIS includes data, demos and	Complete, cloud-based data platform for SQL and NoSQL DBMS.	Complete, cloud-based data platform for SQL and NoSQL DBMS.	o te, Cloud-based Plath Healthc	Complete, Cloud-based Data Platform for Healthcare	
Compute	integration, and analytics		sample code for evaluating.	integration, and analytics	integration, and analytics			
37 77 -	\heartsuit	♡	♡	♡	0	♥	♡	
7-9/-7								

3. 「事前設定された構成で開始する」を選択します

ホーム > Marketplace > InterSystems IRIS for Health InterSystems IRIS for Health

InterSystems Corporation

L InterSystems	InterSystems Co	tems IRIS for Health	♡ 後で使用するために保存
	作成 プログラムによるデ	事前設定された構成で開始する プロイを実行しますか? 作来の開始	

InterSystems IRIS for Health provides all of the capabilities for building complex, mission-critical, data-intensive applications. It's a comprehensive platform spanning data management, interoperability, transaction processing and analytics, built to accelerate time to value.

Built for Big:

Efficiently enables applications to scale from scratch, both vertically and horizontally to cost-effectively handle any scale workload, data, and users.
Accelerating Connections:

· Native support for FHIR and every major global healthcare messaging standard, applications can rapidly ingest, normalize, and share information.

Deeper Intelligence:

Advanced data prep capabilities to create transformative healthcare models and optimize the effectiveness of learning solutions.

4. 最小構成にするため、「開発/テスト」「汎用 D シリーズ」を選択し、「VM の作成を続け る」をクリックします

ホーム > Marketplace > InterSystems IRIS for Health > ワークロードに適した推奨される既定値を選択する

● ワークロードに適した推奨される既定値を選択する

仮想マシンをすばやくカスタマイズするには、次のあらかじめ設定された構成のいずれかを選択します。これらの構成はいつでも変更することができま す。

ワークロード環境を選択する

	開発/テスト	全般 既定	運用環境/ミッション クリティカル
	✔ ブート診断	✔ ブート診断	✔ ブート診断
	Premium ディスク	✓ Premium ディスク	✓ Premium ディスク
	高可用性	高可用性	✔ 高可用性
	Azure Backup (使用可能な場合)	Azure Backup (使用可能な場合)	✓ Azure Backup (使用可能な場合)
7	ロークロードの種類を選択する		
	汎用 (D シリ−ズ) 既定	メモリ最適化 (E シリーズ)	コンピューティング最適化 (F シリーズ)
	Example sizes DS2_v2: 2 CPU, 7 GB DS3_v2: 4 CPU, 14 GB	Example sizes E2s_v3: 2 CPU, 16 GB E4s_v3: 4 CPU, 32 GB	Example sizes F2s_v2: 2 CPU, 4 GB F4s_v2: 4 CPU, 8 GB
	 	自荷の高いメモリ内アブリケーション用に最 適化された高いメモリ対コア比率	ロ コンピューティング集中型のワークロード用に最 適化された高い CPU 対メモリ比率
	Workload types エンタープライズ アプリケーション、リレーショナ ルデータベース、分析	 Workload types SAP HANA、SQL Hekaton、その他の大規 模なメモリ内ワークロード 	Ø Workload types パッチ処理、Web サーバー、ゲーム
V			

VM の作成を続行する

この手順をスキップする

5. 仮想マシンの作成に必要なパラメータを入力します

ホーム > Marketplace > InterSystems IRIS for Health > ワークロードに適した推奨される既定値を選択する > 仮想マシンの作成

仮想マシンの作成

基本 ディスク ネットワーク 管理 詳細 タグ 確認および作成

Linux または Windows を実行する仮想マシンを作成します。Azure Marketplace からイメージを選択するか、独自のカスタマイズされたイメージを 使用します。 [基本] タブに続いて [確認と作成] を完了させて既定のパラメーターで仮想マシンをプロビジョニングするか、それぞれのタブを確認してフル カスタマイ ズを行います。 クラシック VM をお探しですか? Azure Marketplace から VM を作成する

プロジェクトの詳細

デプロイされているリソースとコストを管理するサブスクリプションを選択します。フォルダーのようなリソース グループを使用して、すべてのリソースを整理し、管理します。



6. 検証が成功したら、「作成」をクリックして仮想マシンを作成します

ホーム > Marketplace > InterSystems IRIS for Health > ワークロードに適した推奨される既定値を選択する > 仮想マシンの作成 仮想マシンの作成 ✓ 検証に成功しました 基本 ディスク ネットワーク 管理 詳細 タグ 確認および作成 製品の詳細 InterSystems IRIS for Health クレジットの対象外です 🛛 0.0000 JPY/時間 発行者: InterSystems Corporation 利用規約 | プライバシーポリシー サブスクリプション クレジット適用可能 🛛 Standard D2 v3 発行者: Microsoft 14.4480 JPY/時間 利用規約|プライバシーポリシー 他の VM サイズの価格

利用規約

"作成"をクリックすることで、お客様は (a) 上記の Marketplace のオファリングに関連する法律条項とプライバシーに関する声明に同意し、(b) Microsoft より、そのオファリングに関連する料金が、現在の支払い方法に対して Azure サブスクリプションと同じ時求頻度で時求されることを認め、かつ、(c) Microsoft がお客様の連絡先得報、使用量情報、取引に関する情報を、サポート、請求、その他の取引上のアクティビティを目的として、オファリングのプ ロバイダーと共有する可能性があることに同意するものとします。Microsoft は、サード パーティのオファリングに対する権利は提供しません。その他の詳細に ついては、Azure Marketplace 使用条件を参照してください。

基本

サブスクリプション	無料試用版
リソース グループ	(新規) IRISResource
仮想マシン名	IRISforHealth
地域	(アジア太平洋) 東日本
可用性オプション	インフラストラクチャ冗長は必要ありません
認証の種類	パスワード
ユーザー名	irisadmin
ディスク	
OS ディスクの種類	Standard SSD
マネージド ディスクを使用	はい
エフェメラル OS ディスクを使用する	いいえ
ネットワーク	
仮想ネットワーク	(新規) IRISResource-vnet
サブネット	(新規) default (10.0.0.0/24)
パブリック IP	(新規) IRISforHealth-ip
NIC ネットワーク セキュリティ グループ	(新規) IRISforHealth-nsg
高速ネットワーク	77
この仮想マシンを既存の負荷分散ソリューションの後ろ に配置しますか?	เบเนี
管理	
ブート診断	オン
OS のゲスト診断	77
Azure Security Center	なし
診断ストレージ アカウント	(新規) irisresourcediag
システム割り当てマネージドID	オフ
自動シャットダウン	77
詳細	
拡張機能	なし
クラウドの初期化	เงเงิ
作成 < 前へ	次へ > Automation のテンプレートを

<前へ 次へ > Automation のテンプレートをダウンロードする

7. デプロイが完了するまで待機します。デプロイ完了後に「リソースに移動」をクリック します

前削	除 🛇 キャンセル 🟥	再デプロイ 🏾 🕗 最新の情報に到	更新				
	デプロイが進行 デブロイ名: CreateVm-in サブスクリブション: 無料詞 リソースグループ: IRISResc	中です tersystems.intersystems-iris-h 用版 (0a90f9d5-ee3a-488d-9o purce	ealth-in-2019 140-0cbd4053	開始時刻: 2019/9/3 15:14:38 相関 ID: 9083b956-1018-420f-ab1e-(D6f70e680105		
^ .	展開の詳細(タワンロート) リッース	種類	状態	操作の詳細			
Ø	irisforhealth301	Microsoft.Network/netw	Created	操作の詳細			
Ø	IRISResource-vnet	Microsoft.Network/virtu	OK	操作の詳細			
Ø	IRISforHealth-nsg	Microsoft.Network/netw	OK	操作の詳細			
Ø	IRISforHealth-ip	Microsoft.Network/publ	ОК	操作の詳細			
0	irisresourcediag	Microsoft.Storage/stora	Accepted	操作の詳細			
前川	除 🛇 キャンセル 🏦	再デプロイ 🛛 最新の情報に]	更新				
<	デプロイが完了	しました					
デプロイ名: CreateVm-intersystems.intersystems-iris-health-in-2019 開始時刻: 2019/9/3 15:14:38 サブスクリプション: 無料試用版 (0a90f9d5-ee3a-488d-9d40-0cbd4053 相関 ID: 9083b956-1018-420f-ab1e-06f70e680105 リソース グループ: IRISResource							
◇ 展開の詳細 (ダウンロード)							
~ ;	次の手順						
	リソースに移動						

8. パブリック IP アドレス等を確認する。Studio や管理ポータルでアクセスする際に使用

します

IRISforHealth 仮想マジン						
	«	🗣 接続 🕨 開始	🤁 再起動 🔳 停止 🕃 キャプチャ 🧰 削除 💍 最新の情報に更新			
() 自動シャットダウン	•	リソース グループ (変更) 状態	: IRISResource 実行中		コンピューター名 オペレーティング システム	: IRISforHealth : Linux (ubuntu 18.04)
🥌 バックアップ		場所	: Japan East		サイズ	: Standard D2 v3 (2 vcpu 数、8 GiB メモリ)
🥔 ディザスター リカバリー		サブスクリプション (変更)	無料試用版		エフェメラル・OS ディスク	<u>. 該当たし</u>
🕺 更新プログラムの管理		サブスクリプション ID	0a90f9d5-ee3a-488d-9d40-0cbd4053d4eb		パブリック IP アドレス	: 52.185.150.90
🝰 インベントリ				<u> </u>	ノライヘート IP アトレス	: 10.0.0.4
🍢 変更の追跡					仮想ネットワーク/サブネッ	IRISResource-vnet/default
🔝 構成管理 (ブレビュー)					DNS 名	: 確成
 ポリシー 		97 (変更)	: タクを追加するにはここをクリック	*		

9. シリアルコンソールを使い、仮想マシンにアクセスします。

IRISforHealth 仮想マシン										
_⊃	«	➡ 接続	▶ 開始	♥ 再起動	■ 停止	😥 キャプチ	ヤ 🛅	削除 🐧) 最新の情報	に更新
		リソース グル	レープ (変更)	: IRISResour	rce					
自動シャットダウン		状態		:実行中						
🥔 バックアップ		場所		: Japan East						
🍣 ディザスター リカバリー		サブスクリフ	プション (変更)):無料試用版	R					
🕺 更新プログラムの管理		サブスクリフ	プション ID	: 0a90f9d5-	ee3a-488d-	9d40-0cbd40)53d4eb			
🝰 インベントリ										
🐚 変更の追跡										
🧱 構成管理 (プレビュー)										
ⓒ ポリシ−		タク (変更))	: タクを追加す	るにはここをク	リック				
🛃 実行コマンド										
陸坦		次に指	定する直近の	D期間のデータを	表示する: 🤇	1 時間	6 時間	12 時間	1日間	7日
■の		CDU (포	+4n)							
		CFO (1	~1)							
		100%								
前面 メトリック		80%								
👻 診断設定		70%								
🌪 アドバイザーのレコメンデーション		60%								
1 0 ログ		50%								
		30%								
188 接続モニター		20%								
サポート + トラブルシューティング		10%								
💝 リソース正常性		0%			14:30					
🛂 ブート診断		Percen irisforh	itage CPU (平均) iealth							
■ Performance Diagnostics (プ										
📍 パスワードのリセット		ネットワー	ク(合計)							
🏓 再デプロイ										
🔳 อปราม ฉบบ-ม		45MB 40MB					٨			
接続のトラブルシューティング		35MB					_Λ			
 新日い世ポート日クエフト 		30MB					-1			
■ #ICUD9/0=ド 99±スド	T	25MB					-1			

作成した管理者アカウントでログインします(一度コンソールにフォーカスしてから Enter

を押してください)



 シリアルコンソール上で docker image の確認や docker 起動用ファイル、評価用ライ センス等の取得を行い、docker image を起動します。

10.1 docker images コマンドで IRIS for Health 2019.2 を確認

docker images

irisadmin@IRISforHealth:~\$	docker images			
REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
intersystems/irishealth	2019.2.0.107.0	68a7ce8f3e68	2 months ago	2.77GB
irisadmin@IRISforHealth:~\$				

10.2 wget コマンドで zip ファイルを取得

wget http://20.188.2.244/ISC_4Com.zip



10.3 ダウンロードした zip ファイルを /opt/以下にコピーして、/opt に移動

sudo cp ISC_4Com.zip /opt/

cd /opt



10.4 zip ファイルの解凍用 unzip をインストール

sudo apt-get install unzip

```
irisadmin@IRISforHealth:/opt$ sudo apt-get install unzip
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages were automatically installed and are no longer required: linux-azure-cloud-tools-4.18.0-1018 linux-azure-headers-4.18.0-1018
  linux-azure-tools-4.18.0-1018 linux-cloud-tools-4.18.0-1018-azure
  linux-headers-4.18.0-1018-azure linux-image-4.18.0-1018-azure
  linux-modules-4.18.0-1018-azure linux-tools-4.18.0-1018-azure
Use 'sudo apt autoremove' to remove them.
Suggested packages:
 zip
The following NEW packages will be installed:
  unzip
0 upgraded, 1 newly installed, 0 to remove and 67 not upgraded.
Need to get 167 kB of archives.
After this operation, 558 kB of additional disk space will be used.
Get:1 http://azure.archive.ubuntu.com/ubuntu bionic/main amd64 unzip amd64 6.0-21ubuntu1 [167 kB]
Fetched 167 kB in 0s (474 kB/s)
Selecting previously unselected package unzip.
(Reading database ... 105449 files and directories currently installed.)
Preparing to unpack .../unzip_6.0-21ubuntul_amd64.deb ...
Unpacking unzip (6.0-21ubuntu1) ...
Processing triggers for mime-support (3.60ubuntul) ...
Setting up unzip (6.0-21ubuntul) ...
Processing triggers for man-db (2.8.3-2ubuntu0.1) ...
```

IO.5 ISC_4Com.zip ファイルを解凍

sudo unzip ISC_4Com.zip

irisadmin@IR]	[SforHealth:/opt\$ sudo unzip ISC_4Com.zip
Archive: ISC	C_4Com.zip
inflating:	ISC/change_password.sh
inflating:	ISC/docker-compose.yml
creating:	ISC/dur/
inflating:	ISC/help.sh
inflating:	ISC/info.sh
inflating.	TSC/inia

10.5 ISC フォルダに移動し、docker-compose.yml ファイルを使用して IRIS for Health の docker image を起動

cd ISC

docker-compose up -d

irisadmin@IRISforHealth:/opt\$ cd ISC							
irisadmin@IRISforHealth:/opt/ISC\$ ls							
change_password.sh	dur	info.sh	key	reset.sh			
docker-compose.yml	help.sh	iris	load.sh	status.sh			
irisadmin@IRISforHea	alth:/opt/	ISC\$ dock	cer-compos	se up -d			
Creating network "is	sc_default	" with th	ne default	driver			
Creating irishealth	done						

11.外部から IRIS 管理ポータルにアクセスできるように、Azure 上の仮想マシンのネット ワーク設定を変更し、51773/52773の受信ポート規則を追加します

ホーム > IRISforHealth - ネットワーク								
IRISforHealth - ネットワーク Effitty								×
,○ 摸责(Ctrl+/) 《	🍲 ネットワーク インターフェイスのき	妾続 🐢 ネットワーク インターフェイスのデタ						
🧕 振要	📱 ネットワーク インターフェイス:	irisforhealth301 有効なセキュリティルール トポロジ						
 アクティビティ ログ 	仮想ネットワーク/サブネット: IRISP	Resource-vnet/default NIC パブリック IP: 52.185.150.90	NIC プライベート IP: 10.0.0.4	高速ネットワーク: 無効				
📫 アクセス制御 (IAM)	受信ボートの規則 送信ボ	ートの規則 アプリケーションのセキュリティ グループ 負荷分散					_	
• 97	🕏 ネットワーク セキュリティ グル	ーブ IRISforHealth-nsg (ネットワーク インターフェイスに接続: irisfo	orhealth301)				愛信が	一トの規則を追加する
★ 問題の診断と解決	影響 0 サフネット、1 ネットワー	-9 129-7212		-				
	级先度 	14.00	N-1-	7013/4	9-2	用光	77939	
※ ネットワーク	1010	A SSH	22	109	1日草 10.et.alblaturadi	注意 Maturalitational	o हरना • सम्बद्ध	
🖳 9100	65001	AllowAturel.oxdRalancerInBound	118. (F#	128	Azurel ox/Balancer	(千音	০ ফল	
0 セキュリティ	65500	DenyAllInBound	任意	任意	任意	任意	 拒否 	
☑ 抗張機能								
6% 絶絶的デリパリー (プレビュー)								
可用性セット								
💼 受信セキュリティ規	則の追加	>	●信わま	っリティ想則の追加	1	×		
IRISforHealth-nsg			IRISforHealth	-nsg				
📌 Basic			📌 Basic					
* ソース 🛛			* ソース 🛛					
Any		~	Any			~		
* ソース ポート範囲 ❶			* ソース ポート範囲	0				
*			*					
* <u></u>			* 宛先 6					
Any		~	Any			~		
			+ = + = + = =	<u> </u>				
* 宛先不一ト範囲 💿		4	52773	,		~		
5///5		•						
* プロトコル								
Any TCP UD	P ICMP			F ODF ICMIF				
* アクション			* アクション					
第可 拒否								
* 優先度 θ			★ 優先度 🛛					
1030			1020					
* 名前			* 名前					
Port_51773		✓	Port_52773			~		
			説明					
說明								

12.Web ブラウザを起動し、IRIS 管理ポータルにアクセスします。

IP アドレスは 8. で確認した IP アドレスを使用します。

http://<ipaddress>:52773/csp/sys/UtilHome.csp

R ログイン IRI	s >	< +				_	×
$\leftarrow \ \ni \ C$	① 保護されていなし	\通信 52.185.1	50.90:52773/cs	p/sys/UtilHome.csp)	☆	:
	IR						
	インスタンス: IRIS						
		コーザ名					
		- 74					
		パスワード					
				ログイン			

I3. _SYSTEM/SYS でログインします。パスワード変更画面に遷移するので任意のパスワードに変更します。(ここで設定したパスワードを忘れないようにしてください)
 管理ポータルへは、この後も何度もアクセスするので、ブックマーク等を追加しておくと便利です。

14.今回のチュートリアルで使用する新規ネームペース/データベース MTA を作成します。
管理ポータルから システム管理→構成→システム構成→ネームスペースを選択します

Ris Data Platform 10 253960fdf5e0 ネームスペース %S	YS ஜே ユーザ_sist	∞ ライセ	>ス先 ISC Learning Services -	Stud	lent Key インスタンス IRI	s
うこそ, _SYSTE	M क					
☆ ホーム	橫成	>	システム構成	>	VEIL-84655	c
	セキュリティ	•	101 011	>	ネームスペース	¢
Health	ライセンス	-	ミラー設定	>	0-カルデータベーズ ^{通10}	オプション
	暗号化	>	データベースバックアップ	>	リモートデータベース	c
Analytics			ウェブゲートウェイ管理	۵	シャード構成	c
			squ とオブジェクトの設定	>	ジャーナル設定	c
lnteroperability			デバイス設定	>		
			国際言語設定	>		
🔅 システムオペレーション			レポートサーバー	>		
			追加の設定	>		
🔍 システムエクスプローラ						
📅 システム管理						

新規ネームスペース作成 をクリックします

システム>構成>ネームスペース ネームスペース 新規ネームスペース作成							
現在のネーム	ムスペースお	よびそれら	のグローノ 行歌: 1000	バル/ルーチン 結果:6 ページ: k	ノ 小に対するデフ	オルトデータヘ	ベース::
Namespa	ce Globals	Routines	Temp Storage				
%ALL	%DEFAULTDB	%DEFAULTDB	IRISTEMP	グローバルマッピン	ッグ ルーチンマッピング	<u>ブ パッケージマッピング</u>	<u>10189:</u>
%SYS	IRISSYS	IRISSYS	IRISTEMP	グローバルマッピン	<u>ッグ ルーチンマッピンク</u>	<u>ブ パッケージマッピング</u>	-
HSCUSTO	HSCUSTOM	HSCUSTOM	IRISTEMP	グローバルマッピン	ッグ ルーチンマッピング	<u>ブ パッケージマッピング</u>	<u>10189:</u>
HSLIB	HSLIB	HSLIB	IRISTEMP	グローバルマッピン	<u>ッグ ルーチンマッピンク</u>	<u>ブ パッケージマッピング</u>	削除
HSSYS	HSSYS	HSSYS	IRISTEMP	グローバルマッピン	ッグ ルーチンマッピング	<u>ブ パッケージマッピング</u>	削除
USER	USER	USER	IRISTEMP	グローバルマッピン	22 ルーチンマッピンク	Ź <u>パッケージマッピング</u>	削除

ネームスペース名に「MTA」を入力し、「新規データベース作成」ボタンを押して新規データベースを作成します。

システム > 構成 > ネームスペース > 新規ネームスペース		
新規ネームスペース	保存	キャンセル

下記のフォームを使用して新規ネームスペースを作成してください。:

*-624-28	MTA 必須です
コピー元	T
このネームスペースでグローバルのデフォルト・データペースは	 ● ローカル・データベース ● リモート・データベース
グローバルのための既存のデータベースを選択	● 新規データペース作成
このネームスペースでルーチンのデフォルト・データペースは	 ● ローカル・データベース ● リモート・データベース
ルーチンのための既存のデータペースを選択	▲ 新規データベース作成
このネームスペースにデフォルトのウェブアプリケーションを作成	Ø
次からネームスペースマッピングをコピー	
相互運用プロダクション用にネームスペースを有効化	8

データベースウィザードでは データベースの名前、データベースディレクトリの2か所に「MTA」と入力します。 このページでは「次へ」、次ページではパラメータデフォルトのまま「完了」をクリックしてデータベースの作成を完了 します。

	, ,
データペースウィザード	ユ SYSTEM ザ ネ ー ム 気SYS ス ス SSYS ス ス
このウィザードは新規データベースを作成するお手伝い このテータベースの名前を入力してくださ MTA ジスです。 データベースティレクト J MTA ディレクトンが存在し	いをします。
	戻る 次へ 完了 キャンセル ヘルプ
4	

グローバルのための既存のデータベースを選択 欄が「MTA」になっていることを確認し、同様にルーチンのための既 存のデータベースを選択 の箇所も「MTA」を選択します。その後、「保存」ボタンを押してネームスペース/データベ ースの作成を完了します。

システム>欄成>ホームスペース>新州ホームスペース 新規ネームスペース	存 キャンセル	
下記のフォームを使用して新規ネームスペース	を作成してください。:	
<i>ት–</i> ኴスペース ዿ	MTA 必須です。	
コピー元	¥	
このネームスペースでグローバルのデフォルト・データペースは	● ローカル・データベース ○ リモート・データベース	
グローバルのための既存のデータベースを選択	MTA 必須です。	▼ 新規データベース作成
このネームスペースでルーチンのデフォルト・データペースは	● ローカル・データベース ● リモート・データベース	
ルーチンのための既存のデータベースを選択	MTA	▼ 新規データベース作成…
このネームスペースにデフォルトのウェブアプリケーションを作成	×.	
次からネームスペースマッピングをコピー	•	
相互運用プロダクション用にネームスペースを有効化		

しばらくすると、ネームスペース作成ログが表示されるのでエラーが発生していないことを確認して、「閉じる」をクリ

ックします

ネームスペース: MTA ユーザ: _SYSTEM 開始: 2019-09-04 01:37:45



15.Windows10にIRIS for Health アクセス用の Studio2019.2 をインストールします。 http://20.188.2.244/Studio-2019.2.0.107.0-win.msi

からキットをダウンロードし、インストールを実行してください。インストーラーをダブル クリックして起動し、すべてデフォルトのまま「Next」をクリックしてインストールを完 了してください。 16.Studio を起動し、作成した Azure 上の IRIS for Health インスタンスにアクセスします。

InterSystems IRIS Studio を起動し、接続先を登録します

	I Server Connection	×
D	Select a server from the list.	Add
2	A InterSystems IRIS Studio 新規	Edit
ŝ	IRISHealth Ý	2000
Φ	Java 🗸	ОК
	o H 🔯 📜 🌍	Cancel

↓IP アドレス部分は 8.で確認した IP アドレスを入力します。

Connection	×
Server Name: Azure_IRIS4H IP Address: 20.188.6.136	Port: Telnet Port: 51773 23
Web Server IP Address: 20.188.6.136 CSP Server Instance:	Web Server Port: 52773
 Password Authentication Kerberos Kerberos with Packet Security Kerberos with Encription 	Windows IRIS telnet Service Service Principal Name :
	0%
¥	UK Cancel

ユーザ名 _SYSTEM, パスワードには |3. で変更したパスワードを使用します ネームスペースは先ほど作成した「MTA」を選択します

log on to Azure_IRIS4H	X IRIS Connection Manager Select a server and namespace from the list. Server.
	Azure_IRIS4H (20.188.6.136[51773])
Username:	Namespace:
SYSTEM	%SYS HSCUSTOM
Password:	HSLIB HSSYS
John .	
Hemember Password	OK
OK Cancel	Cancel

17.Studioから演習で使用するソースコードをインポートします。

配布資料のソースコード¥MTA.xml をインポートします。

スタジオにて ツール→ローカルからインポート を選択し、上記ファイルを選択してインポートします。

(演習で作成するクラスが足りないために一部エラーが表示されますが、問題ありません。)



18.REST サービスのためのウェブ・アプリケーションの設定を行います。

管理ポータル→システム管理→セキュリティ→アプリケーション→ウェブ・アプリケーション を選択します。

InterSystems"	管理ポータル				ホーム
サーバ 18f0ac2f3377 ネームスペース MT	A <u>変更 ユーザ _system</u>	ライセン	ス先 ISC Learning Services -	Studen	t Key インスタンス IRIS
ようこそ, _SYSTE	M 表示: []] [Ξ			
向 ホーム	構成	>	ユーザ	0	ウェブ・アプリケーション 0
	セキュリティ	>	ロール	0	Doc DB アプリケーション 0
🕀 Health	ライセンス	>	リソース	0	特権ルーチンアプリケーション①
	暗号化	>	サービス	0	
Analytics			アプリケーション	>	
	-		SSL/TLS 構成	0	
lnteroperability			X.509証明書	0	
	-		OAuth 2.0	>	
😳 システムオペレーション			管理ファイル転送接続	0	
	-		システム・セキュリティ	>	
🔍 システムエクスプローラ			監査	>	
	_		セキュリティアドバイザ	0	
问 システム管理			携帯電話	0	
			公開鍵基盤	0	

「新しいウェブ・アプリケーションを作成」を選択

システム>セキュリティ管理>ウェブ・アプリケーション

ウェブ・アプリケーション 新しいウェブ・アプリケーションを作成

以下が現在定義されているウェブ・アプリケーションの一覧です:

フィル	/9: ページサイズ:	0 最大行数	(: 1000 結果: 25 ページ:	< << 1	. » > の1			
	名前	ネームスペース	ネームスペースのデフォルト	有効	タイプ	リソース	認証方法	
	/api/atelier	%SYS	いいえ	はい	CSP	%Development	パスワード	<u>削除</u>
	/api/deepsee	%SYS	いいえ	はい	CSP		パスワード	<u>削除</u>
	/api/docdb	%SYS	いいえ	(よい	CSP		パスワード	<u>削除</u>
	/api/iknow	%SYS	いいえ	はい	CSP		パスワード	削除

以下の情報を入力します。

名前:/csp/mta 説明:(任意) ネームスペース:MTA

REST ディスパッチクラス:MTA.PROD.RESTService

許可された認証方法:「認証なし」、「パスワード」にチェックします。

入力したら「保存」ボタンをクリックします。

システム>セキュリティ管 ウェブ・	
以下のフォームを	使用して新しいウェブ・アプリケーションを作成します:
名前	/ <mark>csp/mta</mark> 必須です。(例./csp/appname)
コピー元	τ.
説明	MTAチュートリアルで使用するRESTサービス用のウェブアプリケーション設定
ネームスペース	MTA MTAのデフォルト・アプリケーション: /csp/healthshare/mta ネームスペースのデフォルト・アプリケーション
アプリケーション有効	
有効	REST ディスパッチ・クラス MTA.PROD.RESTServices メ海です。 メーズ メーズ メーズ メーズ ・・ ・・ ・ ・
	 CSP/ZEN □ アナリティクス ■ 若信 Web サービス □ ログイン CSRF 攻撃を防ぐ
セキュリティの設定	
セッションの設定	ראיד אראיל אראיל אראיל געד אראיל געד
	セッションにクッキーを使用する 常時 ▼ セッションクッキーパス /csp/mta/ ▼

「アプリケーション・ロール」タブをクリックし、このアプリケーションに適用されるロールを設定します。データベースへの検索等を実行するために、MTA データベースへのアクセス権を持つロール「%DB_DEFAULT」を追加します。

利用可能 一覧から「%DB_%DEFAULT」を選択し、▶マークをクリックします。その後「付与する」ボタンを押します。



アプリケーションロールの箇所に以下のように表示されていたら OK です。



3. IRIS for Health の設定手順





1. 患者情報を格納する MTA.SQL.Patient クラスを作成します。

IRIS Studio にて ファイル→新規作成 を選択し、(アイコンをクリックしても OK です)「Cache クラス定義」をクリックします。



パッケージ名:「MTA.SQL」 クラス名「Patient」を入力し、「完了」をクリックします。(大

文字小文字に注意してください)

所規クラスウィザード					×
新規クラスウィザード このウィザードは、新し 以下の手順に従い、 いつでも "完了"を選	へようこそ。 しいCache 次のページ 択すること	。 クラスを作成する手』 ジに進むには "次へ": ができます。	頁を説明します。 を押してください。		
パッケージ名を入力	:				
MTA.SQL					照(w)
クラス名を入力:	~	5.26	-1-	1	
Patient			LPON	<u>v</u> : >	
新しいクラスの説明を	を入力して	ください(オプション):			-
R		June 1			
<]	戻る(B)	次へ(N) >	完了	キャンセル	ヘルプ

以降は、デフォルトのまま、クラスタイプ:「永続(データベースに格納可能)」を選択し、「完了」をクリ ックします。

9 3	MTA.SQL.	Patient.cls *				
	Class {	MTA.SQL	Patient	Extends	%Persistent	
	}					

次に、クラスのプロパティ「PID」を追加します。プロパティは SQL アクセス時にはテーブルにおけるフ ィールドとなります。スタジオで クラス→追加→プロパティを選択します。



プロパティ名「PID」を入力し、プロパティ属性では「ユニーク」にチェックをいれて、ユニークインデッ クスも同時に作成します。

新規プロパティウィザード ×	新規プロパティウィザード ×
ギヨゴロパティウィザ ドムトネマス	プロパティタイプ
新規ノロハブイリオワートへようこと。 このウィザードはクラス定義に新しいプロパティを追加するお手伝いをします。以下の指示に従って、"次へ" を押すと次のページに移動します。いつでむ"完了"を押すことができます。	このプロパティは:
この新しいプロパティの名前を入力して下さい。	
PID	 ● 単一値タイプ: 参照(w)
' この新しいフロパティの説明を入力して下さい(任意):	C コレクションタイプ: ▼
SCORE DESON	含まれる要素タイプ: 参照(w)
	ר וער-אפעיעע
< 戻る(E) 次へ(N) > 完了 キャンセル ヘルブ	< 戻る(B) 次へ(N) > 完了 キャンセル ヘルプ
新規プロパティウィザード ×	
プロパティ属性	
□ 必須 このプロパティは必須(ヌル以外)	
▼ インデックス このプロパティでインデックスを作成	
▶ ユニーク このプロパティでユニークインデックスを作成	
□ 計算 このプロパティはメモリ上の記憶域を割り当てられません。	ATA SOL Patient cls *
□ 計算。このプロパティは計算された値を持ちます。	Class MTA SQL Dationt Extends "Porsistent
SQLフィールド名(オブション):	(
	Property PID As %String;
	Index PIDIndex On PID [Unique];
< 戻る(B) 次へ(N) > 完了 キャンセル ヘルブ	}

プロパティ PID に SqlColumnNumber 指定を行います。Studio のエディタ画面において、%String の後 ろにフォーカスを合わせて、[(大かっこ)を入力するとインテリセンスが起動します。「Sql」と入力する と SqlColumnNumber が候補に表示されるので選択し、値として「2」を入力します。



残りのプロパティは「IRIS4H 演習補助資料」フォルダの MTA.SQL.Patient.txt の内容をコピーペースト してください。その後、ビルド→コンパイル を選択してこのクラスをコンパイルします。

MTA.SQL.Patient.cls *	
Class MTA.SQL.Patient Extends %Persistent {	_
<pre>Property PID As %String [SqlColumnNumber = 2];</pre>	
Index PIDIndex On PID [Unique];	
Property GivenName As %String [SqlColumnNumber = 3];	
Property FamilyName As %String [SqlColumnNumber = 4];	
Property GivenNameKana As %String [SqlColumnNumber = 5];	
Property FamilyNameKana As %String [SqlColumnNumber = 6];	
Property PostalCode As %String [SqlColumnNumber = 7];	
Property Address As %String [SqlColumnNumber = 8];	- スタジオ - [MTA.SQL.Patient.cls *]
Property DateOfBirth As %Date [SqlColumnNumber = 9];	ビルド(B) デバッグ(D) ツール(T) ユーティリティ(U) ウィンドウ
Property Gender As %String [SqlColumnNumber = 10];	ジ コンパイル(C) Ctrl+F7
Property Phone As %String [SqlColumnNumber = 11];	オプションを指定してコンパイル(O) Ctrl+Shift+F7
	ゴゴ すべて再構築(R) F7
l B	オプションを指定してすべて再構築(A)

出力メニューを見ると、クラスと同時にテーブルもコンパイルされ生成していることがわかります。

Compilation started on 09/09/2019 05:44:02 with qualifiers 'cuk /checkuptodate=expandedonly' Compiling class MTA.SQL.Patient Compiling table MTA_SQL.Patient Compiling routine MTA.SQL.Patient.1 Compilation finished successfully in 0.268s.

サンプルデータ

11

 PID, GivenName, FamilyName, GivenNameKana, FamilyName 1001, 太郎, 山田, タロウ, ヤマダ, 1600023, 東京都新宿区 1002, 茂, 小林, シゲル, コバヤシ, 2270062, 神奈川県横浜 1003, 花子, 鈴木, ハナコ, スズキ, 1690074, 東京都新宿区 1004, 誠, 高橋, マコト, タカハシ, 1820033, 東京都調布市 1005, 美香, 伊藤, ミカ, イトウ, 3300853, 埼玉県さいたま 	Kana, PostalCode, Address, DateOfBirth, Gender, Phone 西新宿6丁目, 1970-01-01, 男, 0312345678↔ 市青葉区, 1943-04-25, 男, 0452345678↔ 北新宿1丁目, 1958-12-08, 女, 0423456789↔ 言士見町, 1969-03-22, 男, 0345678901↔ 市大宮区, 1974-08-03, 女, 0485678901↔
次にこのテーブルにサンプルデータをインポートしま スが「MTA」であることを確認し、システムエクスフ ウィザード→データ・インポート を選択し、サンプ	ミす。管理ポータルでホーム画面から、ネームスペー プローラ→SQL を選択します。 °ルデータをインポートします。
 じ Inter Systems ** 管理ポータル サーバ 18f0ac2f3377 ネームスペース MTA 変更 ユーザ _SY3 システム> S01	ホーム STEM ライセンス先 ISC Learning Services - Student Key イン
フィルタ MTA* ◎適用先 すべて ・ C 。 システム □ スキーマ ・ > <u>テーブル</u> > <u>ビュー</u>	ウィザード >> アクション >> テーブルを開く >> ツール >> データ・インボート 転行 参照 SQLステートメント データ・エクスボーテキストファイルからこのシステムにデータをインボートします。 データ移行

1.「インポートファイルが存在する場所」で「マイローカルマシン」を選択します。

2.「ファイルを選択」をクリックし、サンプルデータ¥DemoData_Patient.csv 選択します。

3.このサンプルファイルのエンコードは UTF-8 になっているので、「文字セット」で「UTF8」を選択します。

4.インポートするネームスペースを選択してください」:「MTA」

インポートするスキーマ名を選択:「MTA_SQL」

インポートするテーブル名を選択:Patient をそれぞれ選択します。

データインポートウィザード(ネームスペース MTA)

インポートウィザードはASCIIファイルからSQLテーブルにデータをインポートする手順をご案内します。

インポートファイルが存在する場所 🔘 18f0ac2f3377 🖲	●マイローカルマシン	
インポートファイルのパスおよび名前を入力してくださ ファイルを選択 DemoData_Patient.csv	A:	文字セットUTF8 UTF8
インポートするネームスペースを選択してください:	MTA v	
インポートするスキーマ名を選択:	MTA_SQL	▼]
インポートするテーブル名を選択:	ひとつ選択してください	
	Patient PulseRate	

次ページはデフォルトのまま「次へ」をクリックします。(SqlColumnNumber 順にフィールドは並んでい ます)

次ページでは CSV 形式のデータをインポートするため、

「カラムの区切りにどの区切り文字を使用しますか?」の設定で キャラクタ にチェックをいれ、テキ ストボックスに「,」を入力します。(残りはデフォルトのままにします)



「完了」を推してインポートを実行します。



「このタスクはバックグラウンドで実行されます。<u>バックグラウンドタスクページを参照するにはここを</u>

<u>クリックしてください</u>」

というメッセージが表示されるので、リンクをクリックしてデータがインポートされていることを確認し

ます。Imported 5 rows という表示があれば成功です。

現在	荘のバックグラウンドタスク:										
~-	ジサイズ	: 0 最大行数:	1000	結果:6 べ-	ージ: (Ø 1					
	ID	開始時間	ネーム スペー ス	タスク	ステータス	音节 有限	エラ ーカ ウン ト				
	92673	2019-09-09 06:07:20	МТА	SQL Import	Imported: 5 rows	DemoData_Patient.csv -> MTA_SQL_Patient Delimiter=, StringQuote=double_DateFormat=MM/DD/(YY)YY TimeFormat=hh.mm:ss TimeStampFormat=ODBC Format NoCheck=0 DeferIndices=0 HasHeaders=1 Charset=UTF8 Completed at 09/09/2019 06:07:20					

「ウインドウを閉じる」 ボタンをクリックして、現在のバックグラウンドタスクのウィンドウを閉じま す。

「完了」ボタンを押すと再び SQL 画面に戻りますので、クエリ「SELECT * FROM MTA_SQL.Patient」 を実行してデータが正しくインポートされたか確認します。

《 ウィザード » アクション » テーブルを開く ツール » ドキュメント »



行数:5パフォーマンス:0.002秒 11 グローバル参照 2371 行実行しました 0 ディスク読み込みレイテンシ (ms) クエリ・キャッシュ: <u>%seqleg.MTA.cls1</u> 最終到

ID	PID	GivenName	FamilyName	GivenNameKana	FamilyNameKana	PostalCode	Address	DateOfBirth	Gender	Phone
1	1001	太郎	山田	タロウ	ヤマダ	1600023	東京都新宿区西新宿6丁目	01/01/1970	男	0312345678
2	1002	茂	小林	シゲル	コパヤシ	2270062	神奈川県横浜市青葉区	04/25/1943	男	0452345678
з	1003	花子	鈴木	ハナコ	スズキ	1690074	東京都新宿区北新宿1丁目	12/08/1958	女	0423456789
4	1004	誠	高橋	マコト	タカハシ	1820033	東京都調布市富士見町	03/22/1969	男	0345678901
5	1005	美香	伊藤	ミカ	イトウ	3300853	埼玉県さいたま市大宮区	08/03/1974	女	0485678901

5 行が影響を受けました

2. MTA.PROD.RESTServices クラスの実装を行います。

スタジオで MTA.PROD.RESTServices を開きます。

MTA.PROD.RESTServices クラスは、 REST 機能を実装するために、 % CSP.REST、 EnsLib.REST.GenericService の二つのクラスを継承しています。

クラス内容主要なコンポーネントは REST アクセスの URL と、処理を実装したメソッドを関連付ける UrlMap XData Block と、処理を実装した各クラスメソッドです。

最初に、UrlMap に今回実装する FHIR 用の URL と呼び出されるメソッドを定義する Route を追加しま す。



<Route Url="/fhir/Patient/:pid" Method="GET" Call="getPatient"/> を追加します。

(IRIS4H 演習補助資料の MTA.PROD.RESTServices.UrlMap.txt を参照してください。)

```
■XData UrlMap [ XMLNamespace = "http://www.intersystems.com/urlmap" ]
{
■<Routes>
<Route Url="/fhir/Patient/:pid" Method="GET" Call="getPatient"/>
</Routes>
}
```

次に getPatient クラスメソッドを実装します。getPatient メソッドの中身が空になっていますので、 IRIS4H 演習補助資料の MTA.PROD.RESTServices.getPatient.txt を参照し、内容をコピーペーストし

てください。



この実装により、<hosturl>/<csp アプリ設定パス>/fhir/Patient/12345 という URL で呼び出された REST の GET リクエストを受信すると getPatient メソッドが実行されます。

3. MTA.PROD.SQLOperation クラスの実装を行います。

SQLOperation クラスは、送られてきた PID 情報を検索キーとして、MTA_SQL.Patient テーブルからデ ータを取り出し、応答メッセージに結果を格納します。

通常は SQL アダプタを使用して外部データベースに ODBC/JDBC 経由で検索を行うのが一般的な実装で すが、今回は同ネームスペースに存在するテーブルに対して検索を実行しています。

スタジオで MTA.PROD.Operation クラスを開きます。

getPatient メソッドの中身が空ですので、 IRIS4H 演習補助資料の MTA.PROD.SQLOperation.getPatient.txtを開きその内容をコピーペーストします。

```
eMethod getPatient(pPid As %String, Output pResponse As MTA.MSG.PatientInfo) As %Status

{
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
    #
```

Interoperability 機能のテスト機能を使用して、このコードが正常に動作するか確認します。 管理ポータルのトップページから Interoperability→Configure→Production を選択します。 セットアップ時にインポートしたソースに含まれていた Production を開きます。右側ウィンドウの 「Actions」タブから「Open」をクリックし、ポップアップした画面で MTA.PROD.Production を選択し ます。

Settings Q	tion Settings ueue Log Messages Jobs Actions Document Production documentation							
5	Export Export for deployment		الله الله الله الله الله الله الله ال	У-ト: МТА »	名前 日付 ^ PRO	検索 D »	Production	× ファインダ
3	Re-Export Amend existing deployment package		右側のオブション からクラスを選択	MTAPKG »				
Z	New Build a new Production							
	Open Open a different Production	_					+ vンセ	и ок

プロダクションが起動していないので、「Start」をクリックして開始します。

MTA.PROD.SQLOperation を選択し、右側メニューの Action タブから「Test」をクリックします。

Production Co	onfiguration	Start	itop		Q	ソート: Name Status Number 表示:
Production Running Services	Processes 🔹	Category: All	• Operations	Legend	Production Settings	MTA.PROD.SQLOperation
MTA.PROD.RESTServices	MTA.PROD.SQLTOFHIR	Process (MTA.PROD	.SQLOperation	on •	Settings Queue Log Messages Jobs Actions

起動したテストツールのページでパラメータとして Pid:「1001」, Type:「Patient」を入力し、「Invoke ~」をクリックしてテスト実行します。Test Results で患者情報が表示されるとテスト成功です。

MTA.PROD.SQLOperation	☐ Test Re Session Reques Respon	ults Id: 30 Visual Trace Sent: 2019-09-11 00:29:49.977 se Received: 2019-09-11 00:29:49.98
		G.Patientinfo
		4004
auest Type: MTA.MSG.EHIRRequest	Given	iame 大郎
	Family	Name LLI
Request Details	Given	lameKana タロウ
	Family	NameKana ヤマダ
Defect	Postal	ode 1600023
ype: Patient	Addres	s 東京都新宿区西新宿6丁目
Invoke Testing Service	DateOf	Birth 01/01/1970
Invoke resting service	Gende	男

「Visual Trace」リンクをクリックしてテストされた内容を確認することができます。

Services EnsLib.Testing Service	Processes EnsLib.Testing Process	Operations MTA.PROD SQLOperation	<pre>« »</pre>	Header Body Co	entinfo
				<オブジェクトld>	31
[1] 2019-09-11 00 Testing Perue	:29:49.978			PID	1001
- Testing.Reque				GivenName	太郎
	[2] 2019-09-11 00	29:49.979		FamilyName	山田
	FHIRRequest			GivenNameKana	タロウ
	2019-09-1	1 00 29 49 980		FamilyNameKana	ヤマダ
		PatientInfo		PostalCode	1600023
				Address	東京都新宿区西新宿6丁目
				DateOfBirth	01/01/1970
				Gender	男
				Phone	0312345678

4. MTA.DTL.Patient クラスの新規作成と実装を行います。

MTA.MSG.PatientInfo クラスを FHIR の Patient リソースにマッピングするための、データ変換クラス (DTL クラス)を新規作成します。ネームスペースが MTA であることを確認し、Interoperability→Build →Data Transformations を選択します。

InterSystems IRIS Data Platform	管理ポータル			木-
サーバ 18f0ac2f3377 ネームスペース MTA	、変更 ユーザ <u>_system</u>	ライセンス第	E ISC Learning Services	- Student Key
ようこそ, _SYSTE	▶ 表示:			
	Configure	>	Business Processes	0
	Build	> (Data Transformations	Ū
Health	View	>	Business Rules	追加オプションや詳細
- Health	List	>	Record Maps	0
Analytics	Monitor	>	CSV Record Wizard	0
	Manage	>	Complex Record Maps	0
	Interoperate	>	Java Business Hosts	()
Monitor Interoperability	Test	>		

「New」 ボタンをクリックし、新しいデータ変換クラスを作成します。

Package(パッケージ名):MTA.DTL

Name(クラス名):Patient

Source Class(ソースクラス名): MTA.MSG.PatientInfo

Target Class(ターゲットクラス名): HS.FHIR.DTL.vSTU3.Model.Resource.Patient

Source/Target クラスは直接入力でもかまいませんが、ウィザードの

虫眼鏡ボタンをクリックして

探すこともできます。

ソースクラスのほうは「メッセージクラス」を選択します。



ターゲットクラスのほうは「システムクラスではないすべてのクラス」を選択します。類似クラスにご注意ください。



〕InterSystems ™ 管理ポータル	н
サーバ 18f0ac2f3377 ネームスペース MTA 変更 ユーザ _SYSTEM ライセンス先 ISC Learning Services - Studer	nt K
Interoperability > Data Transformation Builder	
New Open Save Save As Compile 100% V -Add Action- V	×
Source	×
Description Source Type • All Messages HL7 X12 Source Class Source Document Type Actions • All Messages HL7 X12 Astions Target Class Target Class Target Class Target Class	
# Action HS.FHIR.DTL.vSTU3.Model.Resource.Patient Cancel OK	_

次のような画面が表示され、左側にソースクラスとなる MTA.MSG.PatientInfo のプロパティ、右側には ターゲットクラスとなる HS.FHIR.DTL.vSTU3.Model.Resource.Patient クラスのプロパティが表示さ れます。右側のクラスは項目数が非常に多いですが、この画面の黄色矢印部分をドラッグすることで表示 位置を変更することができます。

	Source MTA.MSG.PatientIn	fo		Target HS.FHIR.DTL.vSTU3.Model.Resource.Patient	
\\ v s	ource	p			
	PID	o	⊳⊽ta	arget	
	GivenName	0		⊳ text	
	FamilyName	p	1 1 ≥	⊳ meta	
	GivenNameKana	p	Þ	implicitRules	
	FamilyNameKana	o	Þ	▷ modifierExtension()	
	PostalCode	o	Þ	id	
	Address	p		▷ contained()	
	DateOfBirth	p		▷ extension()	
	Gender	p	Þ	language	
	Phone	p	Þ	▷ newResource()	
			Þ	newResourceReference	
			Þ	> primitiveExtension()	
			Þ	resourceType	
Actions				1	v 🗢
# Action	Condition	Property	Value	Key / Transform	

まずは患者 ID をあらわす「PID」プロパティを FHIR の Patient リソースの Identifier エレメントの value に格納します。先ほどの図の黄色矢印部分をドラッグし、「Identifier」の先頭の>をクリックして展開し、 左側の「PID」と右側の「value」を線で結びます。

MTA.MSG.PatientInfo	HS.FHIR.DTL.vSTU3.Model.Resource.Patient
MTA.MSG.PatientInfo	HS.FHIR.DTL.vSTU3.Model.Resource.Patient
Phone	> Value > > > > > > > assigner > active

右側の Target 側がオレンジになった状態で離すと、下の図のように二つの要素が関連付けられます。

MTA.MSG.PatientInfo	HS.FHIR.DTL.vSTU3.Model.Resource.Patient
▼ source ● PID ● GivenName ● FamilyName ● GivenNameKana ● PostalCode ● Address ● DateOfBirth ● Gender ● Phone ●	Image: constraint of the system Image: constraint of the system

次は FHIR の name エレメントを作成します。FHIR では name エレメントは HumanName データタイ プを使用しますが、IRIS for Health にも該当する「HS.FHIR.DTL.vSTU3.Model.Datatype.HumanName」 クラスがあるので、まずはそのクラスのインスタンスを作成します。

右側の Target から name プロパティを選択し、上部の「--Add Action-」メニューから「set」を選択し ます、

-Add Action-	. 🗙 I	J (🇊 表示: 🔡 🕕
set clear remove			Target HS.FHIR.DTL.vSTU3.Model.Reso
insert		->>> ide	entifier()
-Conditional Actions- if		⊳ act	tive
for each		⊳ ⊽ nai	ime()
switch			id
- Case			·

Value 欄に新規インスタンスを生成するコマンド

「##class(HS.FHIR.DTL.vSTU3.Model.Datatype.HumanName).%New()」を入力します。(大文字 小文字に注意してください)コピーペーストする場合は IRIS4H 演習補助資料の

MTA.DTL.Patient(HumanName).txt を参照ください。

	HS.FHIR.DTL.vSTU3.Model.Re	Ê	»
			Transform Action Tools
-0	→>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>		Details for the selected action
	⊳ active		assign
		-	Set the value of a target property.
		•	View documentation
	id		Action
			set V
	\triangleright extension()		Property
	>>> newResource()	-	target.name.(1) Property whose value will be set. Double-clicking on a target property in the diagram
			will place that property in this field.
	newResourceReference	. (Value
	>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>>	- (##class(HS.FHIR.DTL.vSTU3.Model.Datatype.HumanName).
			place that property in this field.
	⊳ use		Key
	⊳ text		For collection properties, this string specifies the member of the collection that is the
			target or this assignment.

次に、「use」を選択し、同じように「set」Actionを選んで、value に"official"を入力します。

-Add Action-	🗙 🎚 🐧 🌍 👼 🚛	
set clear remove	HS.FHIR.DTL.vSTU3.Model.Re	Details for the selected action
- insert - Conditional Actions- if for each break	▷[♡ name() ▷[id ▷[▷ extension()	assign Set the value of a target property. View documentation Action set Property
switch case -Advanced Actions- code	▷ newResource() ▷ newResourceReference ▷ primitiveExtension()	Property whose value will be set. Double-clicking on a target property in the diagram will place that property in this field. Value Value Value Conficial Value Value Conficial Value Value Value Value Value Value Value Value Value Value Value Value Value Value Value Value
trace subtransform -Annotation Actions-	▶ use ▷ text	Key " " For collection properties, this string specifies the member of the collection that is the target of this assignment. Anablad

左側ソースの GivenName、FamilyName を右側ターゲットの given(), family に、PID の時と同様に線

	PID GivenName FamilyName GivenNameKana FamilyNameKan PostalCode Address DateOfBirth	a aa	م -م -م م م م - م	-0-	>>> >>>>>>>>>>>>>>>>>>>>>>>>>>>>		
Actions (DDGG X	u 🔋 🗓					▼ 🤤
# Action	Condition	Property	Value			Key / Transform	
1 set		target.identifier.(1).value	source.PID				
2 set		target.name.(1)	##class(HS.FHIR.DTL.vSTU3.Model.Datatype.HumanName).9	New()			
3 set		target.name.(1).use	"official"				
4 set		target.name.(1).given.(1)	source.GivenName			**	
5 set		target.name.(1).family	source.FamilyName				

ここで一度コンパイルを行います。



DTLの開発は Studio でも行うことができ、Visual 開発ツール以外に XML 表記のコードも確認できます。

Studio で MTA.DTL.Patient クラスを開きます。

Azure_IRIS4	H/MTA@_SYSTEM - Default_SY	STEM.prj - スタジオ - [MTA.DTL.P	atient.dtl]					
37711(F)	編集(E) 表示(V) プロジェクト(P)	ビルド(B) デバッグ(D) ツール(T) :	ューティリティ(U) ウィンドウ(W) ヘルプ(H)	-				
🖹 , 🚅 , 🖌	i 🖉 🕝 i 🖸 🖸 i 👗 🐚 🙇	- ") (* 🖃 🖻 🔧 🕾	- 🧆 🖄 🖻	è 🗿 🔄 i 🕹 i 🗔 🕻	3 💭 U	x i 🙀 🕨 🖬 💷 💷 🖓 💷 🖉 👘 🖉 👘		
🔭 📎 🗞	2 7 2 6 6 % 1							
а 🦯 🔩 мт	A.PROD.RESTServices.cls 🦂 M	TA.PROD.SQLOperation.cls	MTA.SQLPatient.ds	t			Þ×	ワークスペース
100%	-Add Action-	M 🗙 🛛 🗘 ⊃	表示:				^	■ · · · · · · · · · · · · · · · · ·
	MTA.MSG.Patier	ntInfo	HS.FHIR.DTL.vSTU3.Mode	I.Resource.Patie	ent ^	»		⊕-ta CSPX ⊕-ta Ens
\\ \(\not\) \$0	urce	o(⊳⊽target			Transform Action Tools		EnsLib EnsPortal
C	PID		⊳(⊳ text			Details for the selected action		⊕- <u>™</u> HS
C	GivenName		⊳(⊳ meta			No action selected		ı HSMOD
0	FamilyName		implicitRules			Alternatively, you can select an item in the Actions table beneath the		B-B MTA
0	GivenNameKana		▷(▷ modifierExtension()			diagram.		- deservationList
C	FamilyNameKana		⊳⊂id					🍫 Patient
0	PostalCode		▷(▷ contained()					PulseRate
0	Address	o(⊳(⊳ extension()					B B MSG
C	DateOfBirth	p	⊳ language					B SQL
0	Gender)o	⊳(⊳ newResource()					IIII → IIII MTAPKG
C	Phone	þ	newResourceReferen	ce				SchemaMap
			▷ primitiveExtension()					B→ ルーチン B→ CSPファイル
Actions	1100 × 1	C)		V	٢			∞ 🗀 その他
# Action	Condition Property	,	Value	Key / Transform				
1 set	target.identifier.(1)	.value source.PID						
2 set	target.name.(1)	##class(HS.FHIR.I	DTL.vSTU3.Model.Datatype.Human					
3 set	target.name.(1).us	e "official"					~	H + ▶ H \ プロダクション \ プロジェクト \
<	Anna A. (19) at							7-72K-2 122K09-MTAD
								a more than a more a

メニューから 表示→他のコードを表示 を選択するか、 アイコンをクリックすると、XML 表記のコ ードを表示することができます。この XML コードを変更することもできます。今回は、残りのデータ変換 コードをコピーペーストで入力します。IRIS4H 演習補助資料の MTA.DTL.Patient.txt を開きその内容を 最後の<assign>タグと</transform>タグの間にコピーペーストします。

4 At MTA.DTL.Patient.cls
Class MTA.DTL.Patient Extends Ens.DataTransformDTL [DependsOn = (MTA.MSG.PatientInfo, HS.FHIR. {
Parameter IGNOREMISSINGSOURCE = 1;
Parameter REPORTERRORS = 1;
Parameter TREATEMPTYREPEATINGFIELDASNULL = 0;
■XData DTL [XMLNamespace = "http://www.intersystems.com/dtl"]
<pre> transform sourceClass='MTA.MSG.PatientInfo' targetClass='HS.FHIR.DTL.vSTU3.Model.Resource.Pa</pre>

コンパイルを実行後、管理ポータル側のブラウザを再読み込みすると XML 表記のコードの変更が反映されていることがわかります。

Source MTA.MSG.PatientInfo	Target HS.FHIR.DTL.vSTU3.Model.Resource.Patient
(⊽ source)C	resourceType
PID	
GivenName	> active
FamilyName	● name()
GivenNameKana	
FamilyNameKana	gender
PostalCode	birthDate
Address	deceasedBoolean
DateOfBirth	deceasedDateTime
Gender)	
Phone C	⊳ → D → D → D → D → D → D → D → D → D →
	MultipleBirthBoolean

この DTL を呼び出す MTA.PROD.SQLTOFHIRProcess は今回はすでに実装されているものを使用します。

管理ポータルからこのビジネスプロセスを開いて内容を確認します。

管理ポータルのトップページから Interoperability→Configure→Production で Production 画面を 開き、MTA.PROD.SQLFHIRProcess を開きます。Classname 属性の横の虫眼鏡ボタンをクリックして、 ビジネスプロセスを表示します。

Production C	Configuration	Start Stop	ソート: Name Status Number 表
Production Humning Services MTA.PROD.RESTServices	Category: All Processes	Legend Production Settings Operations MTA PROD.SQLOperation	MTA.PROD.SQLTOFHIRProcess Settings Queue Log Messages Jobs Actions Apply
			Description

Data Transformation Class として「MTA.DTL.Patient」が指定されていることを確認します。



Production 画面に戻り、SQLOperation と同じ手順で SQLTOFHIRProcess もテストを行います。

	MTA.PROD.SQLTOFHIRProcess Production MTA.PROD.Production	
Reque	st Type: MTA.MSG.FHIRRequest	2
🗆 Req	uest Details	
Pid:	1001	
Type:	Patient	
= Test	Invoke Testing Service Results	
Sess Requ Resp MTA	ion 1d: 34 Visual Trace test Sent: 2019-09-11 00:43:57.390 ponse Received: 2019-09-11 00:43:57.404 MSG.FHIRResource プジェクトtab [34]	

応答は Stream で取得しているのでテスト画面では表示されませんが、Visual Trace 画面から確認するこ とができます。最後の応答メッセージを選択して、Contents タブをクリックします。



5. ブラウザから REST サービスを呼び出す

ブラウザから <u>http://<Azure</u>の ip アドレス>/csp/mta/fhir/Patient/1001 にアクセスし、患者情報が FHIR 形式で表示されることを確認します。

Ē € ☐ 104.41.173.6 X + ∨	
\leftrightarrow \rightarrow \circlearrowright \bigcirc 104.41.173.6:52773/csp/mta/fhir/Patient/1001	
<pre>{ "resourceType": "Patient", "address": [</pre>	● 10441.173.652773/csp/mta/fhir × + ← → C ① 保護されていない遺信 104.41.173.652773/csp/mta/fhir/Patient/1002 * "resourceType": "Patient",
, Edge での表示例	Chrome での表示例

- 6. Observation の実装にトライしてみてください。
 - (ア)MTA.SQL.PulseRate クラスをコンパイルして、サンプルデータ ¥DemoData_PulseRate.csv をインポートしてください。
 - (イ) MTA.PROD.RESTServices クラスの UrlMap に/fhir/Observation のエントリを 追加してください。
 - (ウ)<u>http://<azure_ip>:52773/csp/mta/fhir/Observation?patient=1001へのア</u> クセスをテストします。

4. 発展編







目次

- 1:サーバアクセスの基本となるフレームHTMLを作成する
- 2: JavaScriptでサーバにアクセスする
- 3:データを取得しJSON形式でメモリに格納する
- 5:Graph.jsライブラリを参照する
- 4:条件式を用いて取り出す
- 6:Graph.jsのデータ入力形式に合わせてデータフォーマットを作成する
- 7:Graph.jsの表示フォーマットを整える

8:総合テスト

1:サーバアクセスの基本となるフレームHTMLを 作成する



HTMLの構造



Gunma University Hospital SIC, Confidential

1:サーバアクセスの基本となるフレームHTMLを 作成する



Gunma University Hospital SIC, Confidential

1:サーバアクセスの基本となるフレームHTMLを 作成する



HTMLの構造

HTMLの構造



1:サーバアクセスの基本となるフレームHTMLを 作成する





Gunma University Hospital SIC, Confidential

2: JavaScriptでサーバにアクセスする



1



バニマクタフオス 1-1-2- Corint 7:44

2:JavaScriptでリーハにアクビス9つ	
<pre></pre>	
<pre>{</pre>	
, var.xhr.=new_XMLHttpRequest(); // HTIPサーバへの接続インスタンスを作成。 xhr.responseIype = text ; // サーバからの応答をテキスト形式に指定。 xhr.open(GET、http://iris4h-fhirapi.japaneast.cloudapp.azure.com:52778/csp/sqltofhir/fhir/observation/pulserate/1001~, xhr.send(); // サーバに送る:	.true); // RESTのGETコマンドで、リソースを指定する+
* xhr.onreadystatechange = function(){ // サーバから広答があったときに実行される箇所: if (xhr.readyState=4 && xhr.status==200) [// readyState=4[100ME (操作完了) かつstatus=200[10K (要求通りの処理/ var.data = JSNLparse(xhr.responseText); // !重要 JSN形式を構造体として解釈する これ以降はJSNNタグをプロバティとして記 // 追加した部分: //変更した部分: var.outtext.=、xhr.responseText + ' patient id =' + data.id + ' ;/ data.idと記述されている~id~idJSNN内のidタグ?	がされた)↓ D述できる↓ を指す↓
//支史しに部分↓ target.innerHTML = outtext: //読み取ったタグの値をHTMLとして出力↓ 】↓ 】:↓	
↓]↓ ↓	
<pre></pre>	
<pre>{ dody>4 <form action="" id="id_form1" name="form1">4 <form action="" id="id_form1" name="form1">4</form></form></pre>	
↓ ↓	



3:データを取得しJSON形式でメモリに格納する

	<pre> (near) * (near) * (NF+8) * (NF+8</pre>	
⊥⊥ ∖ ⊸ ° ⊔	<title>HHRIESI Step 2</title>	
サンノル	_ background: #eeeeee;↓	
]↓	
	<pre>{script type="text/javascript" language="javascript">+</pre>	
	function onButtonClick() {+ tarset = document.setElementById("output");+	
	var xhr = new XMLHttpRequest(); // HTTPサーバへの接続インスタンスを作成。	
	xhr.responseType = 'text'; // サーバからの応答をテキスト形式に指定。 xhr.open('GET', http://iris4h-fhirapi.japaneast.cloudapp.azure.com:52773/csp/sqltofhir/fhir/observation/pulserate/1001",true); // RESTのGETコマンドで、リソー yhr.seed()・// サーバに详る」	スを指定する↓
	xhr.onreadystatechange = tunction()[// サーハから)は合かのったとさに実行される圏所1 f (xhr.readyState=4& xhr.status=200) (// readyState=4(dDNE (特代完了) かつstatus=200(は0K (要求通りの処理がされた))	
	var data = JSON.parse(xhr.responseText); // !!重要 JSON形式を構造体として解釈する これ以降はJSONタグをフロバティとして記述できる」 //追加した部分」	
	//愛史し/記が♪ var.outtext = xhr.responseText + 'dpr> patient id =' + data.id + 'dpr>': // data.idと記述されている″id″(はJSON内のidタグを指す⊥	
	//変更した部分	
	target.innerHTML = outtext; //読み取ったタグの値をHTMLとして出力	
	31	
	+	
	<^/head>↓	
	Shi>↓	
	↓ Shody≥↓	
	[™] (form name="form1" id="id=form1" action;"")↓ = " +" = +" = +"	
	<pre><input (late="button)(late=button)(</th" <input="" id="[g]textBox " name="textBox " type="button" value="//i"/><th></th></pre>	
	(Sp2) (Site ide Tourback De Adia)	
	(b))	
	↓ /hody>↓	
	+	

Graph.jsの元コードでは、<body>内に<script>タグを置き、その中でChartインスタンスを生成しながら データを割り当てている



4:Graph.jsライブラリを参照する

Graph.jsの元コードでは、<body>内に<script>タグを置き、その中でChartインスタンスを生成しながら データを割り当てている



4:Graph.jsライブラリを参照する



4:Graph.jsライブラリを参照する



サンプル



4:Graph.jsライブラリを参照する

サンプル



JSON構造体を単純に読み出す関数(JSON配列の再帰呼び出しのアルゴリズムを理解する好例)







JSON構造体を単純に読み出す関数(JSON配列の再帰呼び出しのアルゴリズムを理解する好例)











7:Graph.jsの表示フォーマットを整える



表示項目を追加する方法(このフォーマット自体をライブラリ化してもよい)



7:Graph.jsの表示フォーマットを整える



表示項目を追加する方法(このフォーマット自体をライブラリ化してもよい)



7:Graph.jsの表示フォーマットを整える



異なるグラフを表示する方法



7:Graph.jsの表示フォーマットを整える

異なるグラフを表示する方法



お疲れさまでした。 コードは動きましたか?





サンプル



IRIS for Health 評価ライセンスについて

本チュートリアルにご参加いただいました皆様は、IRIS for Healthをローカルマシンなどに インストールするための期間限定(本日より約2ヶ月)の評価ライセンスをご提供することも可能です。

もちろん、本日のチュートリアルで作成された環境も期間限定(本日より約2ヶ月)で継続利用すること も可能です。

もし、別環境にIRIS for Healthをインストールして検証されたい場合は、以下までご連絡お願い致します。

インターシステムズ ジャパン株式会社

紺木(こんき) <u>tkonki@intersystems.com</u> 03-5321-6200

IRIS QuickStartsについて

- QuickStartsは、5分で終わる InterSystems IRIS の学習コンテンツです。
- 機能ごとに内容が分かれ、概要説明のための数分間のビデオと操作を体
- 験するためのサンプル提供、実行手順のガイドが付いています。● ちょっと試したい場合や、機能全般を網羅しながら体験されたい方に最

https://learning.intersystems.com/course/view.php?id=1055&ssoPass=1



IRIS Experience について

- Experience は、InterSystems IRISの興味ある機能について60分程度 で学習・体験できる、オンラインラーニングです。
- QuickStartsのような簡単な流れではなく、じっくり1つの機能を体験・学習されたい方に最適です。
- オンラインラーニング専用画面を利用して、機能解説ビデオの視聴、 確認用クイズへの挑戦、IRIS演習環境を利用した演習が行えます (コースごとにコース専用 2週間の限定環境が用意されます)。

https://www.intersystems.com/learn-play/

Big Data Analytics	InterSystems in the Cloud	Java & InterSystems IRIS 🎂
Use the cutting-edge features available in interSystems IRIS Data Platform™ to powerfully and efficiently analyze your data and scale your big data systems	Improve the efficiency and flexibility of your applications with InterSystems Cloud Manager's automation in provisioning infrastructure and cloud- based deployment.	Unleash your Java application development with InterSystems IRIS** to leverage the full stack of technologies available to you.
Experience It!	Experience It!	Experience It!
NET & InterSystems IRISOU	InterSystems on FHIR	RESTful FHIR & Messaging
Streamine your development of performant. NET applications by connecting to interSystems IRIS using ADO NET, Entity Framework, and more.	Experience the power of connecting HL7 V2® FHIR® applications to a unified health record using the latest version of HealthShare.	Use Health Connect messaging and FHIR components to transform existing formats (C-CDA and HL7 V2) into FHIR.
Experience It!	Experience It!	Experience It!

技術リソースへのリンク集

インターシステムズでは、様々な技術情報を提供しています。ぜひご活用ください。

- InterSystems IRIS Experience (英語)
- InterSystems IRIS QuickStarts(英語)
- オンラインラーニング(英語)
- インターシステムズFAQ
- Developer Community(英語)
 *Google翻訳版
- 日本語ドキュメント
- 各種トレーニングコース

- https://learning.intersystems.com/course/view.php?id=1055&ssoPass=1
 - https://learning.intersystems.com

https://www.intersystems.com/learn-play/

- http://faq.intersystem.co.jp
- https://community.intersystems.com

http://bit.ly/2wh28v2

- http://www.intersystems.com/jp/library/documentation/
- http://www.intersystems.com/jp/services-support/learning-services/