

いまさら人には聞けない・・・

# 正しいクエリの書き方

# シンプルなクエリ

製造元がAirbusまたはBoeingで座席数が400以上の機体が欲しい

航空機	
ID	2 <sup>32</sup>
製造元	A
名称	A
タイプ	A
エンジンタイプ	A
カテゴリー	A
エンジン数	2 <sup>32</sup>
座席数	2 <sup>32</sup>
型番	A
航続距離	2 <sup>32</sup>
最高速度	2 <sup>32</sup>
写真	[写真]
ファミリー名	A
初飛行日	17

# シンプルなクエリ

製造元がAirbusまたはBoeingで座席数が400以上の機体が欲しい

航空機	
ID	2 <sup>32</sup>
製造元	A
名称	A
タイプ	A
エンジンタイプ	A
カテゴリー	A
エンジン数	2 <sup>32</sup>
座席数	2 <sup>32</sup>
型番	A
航続距離	2 <sup>32</sup>
最高速度	2 <sup>32</sup>
写真	□
ファミリーナ	A
初飛行日	17

**QUERY**([航空機];[航空機]座席数>=400)  
**QUERY SELECTION**([航空機];[航空機]製造元="Airbus";\*)  
**QUERY SELECTION**([航空機]; | ;[航空機]製造元="Boeing")

リクエスト #1

**QUERY**([航空機];[航空機]製造元="Airbus";\*)  
**QUERY**([航空機]; | ;[航空機]製造元="Boeing";\*)  
**QUERY**([航空機]; & ;[航空機]座席数>=400)

リクエスト #2

**QUERY BY FORMULA**([航空機];\([航空機]製造元="Airbus"\| [航空機]製造元="Boeing") & \[航空機]座席数>=400)

単一のリクエスト

単一のリクエスト

# 条件の絞り込み順序が分からぬ?

- ▶ **QUERY**([航空機];[航空機]製造元="Airbus";\*)
- QUERY**([航空機]; | ;[航空機]製造元="Boeing";\*)
- QUERY**([航空機]; & ;[航空機]座席数>=400)



# 複雑なクエリ

初飛行が1986, 1987または1990年で 4 エンジン搭載の機体が知りたい

航空機	
ID	2 <sup>32</sup>
製造元	A
名称	A
タイプ	A
エンジンタイプ	A
カテゴリー	A
エンジン数	2 <sup>32</sup>
座席数	2 <sup>32</sup>
型番	A
航続距離	2 <sup>32</sup>
最高速度	2 <sup>32</sup>
写真	[CAMERA]
ファミリー名	A
初飛行日	17

# 複雑なクエリ

初飛行が1986, 1987または1990年で4エンジン搭載の機体が知りたい

航空機	
ID	2 <sup>32</sup>
製造元	A
名称	A
タイプ	A
エンジンタイプ	A
カテゴリー	A
エンジン数	2 <sup>32</sup>
座席数	2 <sup>32</sup>
型番	A
航続距離	2 <sup>32</sup>
最高速度	2 <sup>32</sup>
写真	B
ファミリーナ	A
初飛行日	17

**ARRAY INTEGER(\$\_year;0)**  
**APPEND TO ARRAY(\$\_year;1986)**  
**APPEND TO ARRAY(\$\_year;1987)**  
**APPEND TO ARRAY(\$\_year;1990)**

~~\$date1stjan:=Add to date(!00/00/0000!;\$\_year{1};1;1)  
\$date31dec:=Add to date(!00/00/0000!;\$\_year{1};12;31)  
QUERY([航空機];[航空機]初飛行日>=\$date1stjan;\*)  
QUERY([航空機]; & ;[航空機]初飛行日<=\$date31dec;\*)  
For (\$i;2;Size of array(\$\_year);1)  
    \$date1stjan:=Add to date(!00/00/0000!;\$\_year{\$i};1;1)  
    \$date31dec:=Add to date(!00/00/0000!;\$\_year{\$i};12;31)  
    QUERY([Aircraft]; | ;[航空機]初飛行日>=\$date1stjan;\*)  
    QUERY([Aircraft]; & ;[航空機]初飛行日<=\$date31dec;\*)  
End for  
QUERY([航空機];[航空機]エンジン数=4)~~

# 複雑なクエリ

初飛行が1986, 1987または1990年で4エンジン搭載の機体が知りたい

航空機	
ID	2 <sup>32</sup>
製造元	A
名称	A
タイプ	A
エンジンタイプ	A
カテゴリー	A
エンジン数	2 <sup>32</sup>
座席数	2 <sup>32</sup>
型番	A
航続距離	2 <sup>32</sup>
最高速度	2 <sup>32</sup>
写真	[img]
ファミリーナ	A
初飛行日	17

▶ **QUERY([航空機];[航空機]初飛行日>=\$date1stjan\_1986;\*)**  
**QUERY([航空機]; & ;[航空機]初飛行日<=\$date31dec\_1986;\*)**  
**QUERY([航空機]; | ;[航空機]初飛行日>=\$date1stjan\_1987;\*)**  
**QUERY([航空機]; & ;[航空機]初飛行日<=\$date31dec\_1987;\*)**  
**QUERY([航空機]; | ;[航空機]初飛行日>=\$date1stjan\_1990;\*)**  
**QUERY([航空機]; & ;[航空機]初飛行日<=\$date31dec\_1990;\*)**  
**QUERY([航空機];[航空機]エンジン数=4)**



# 複雑なクエリ

初飛行が1986, 1987または1990年で4エンジン搭載の機体が知りたい

航空機	
ID	2 <sup>32</sup>
製造元	A
名称	A
タイプ	A
エンジンタイプ	A
カテゴリー	A
エンジン数	2 <sup>32</sup>
座席数	2 <sup>32</sup>
型番	A
航続距離	2 <sup>32</sup>
最高速度	2 <sup>32</sup>
写真	[img]
ファミリーナ	A
初飛行日	17

```
QUERY([航空機];[航空機]エンジン数=4) → リクエスト #1
CREATE SET([航空機];"4エンジン") → リクエスト #2
For ($i;1;Size of array($_year);1)
    $date1stjan:=Add to date(!0000/00/00!;$_year{$i};1;1)
    $date31dec:=Add to date(!0000/00/00!;$_year{$i};12;31)
    QUERY([航空機];[航空機]初飛行日>=$date1stjan;*)
    QUERY([航空機]; | ;[航空機]初飛行日<=$date31dec) → リクエスト #3,5,8
    If ($i=1)
        CREATE SET([航空機];"年") → リクエスト #4
    Else
        CREATE SET([航空機];"年2")
        UNION("年";"年2";"年") → リクエスト #6,9
    End if
End for
INTERSECTION("年";"4エンジン";"年") → リクエスト #11
USE SET("年") → リクエスト #12
CLEAR SET("年") → リクエスト #13
CLEAR SET("年2") → リクエスト #14
CLEAR SET("4エンジン") → リクエスト #15
```

v11以来

# 複雑なクエリ

初飛行が1986, 1987または1990年で4エンジン搭載の機体が知りたい

航空機	
ID	2 <sup>32</sup>
製造元	A
名称	A
タイプ	A
エンジンタイプ	A
カテゴリー	A
エンジン数	2 <sup>32</sup>
座席数	2 <sup>32</sup>
型番	A
航続距離	2 <sup>32</sup>
最高速度	2 <sup>32</sup>
写真	B
ファミリーナ	A
初飛行日	17

```
ARRAY INTEGER($_year;0)
APPEND TO ARRAY($_year;1986)
APPEND TO ARRAY($_year;1987)
APPEND TO ARRAY($_year;1990)

$myRequest:=""
For ($i;1;Size of array($_year);1)
    If ($i#1)
        $myRequest:=$myRequest+" OR "
    End if
    $myRequest:=$myRequest+("[初飛行日]>=" + String($_year{$i}) + "-01-01"
    $myRequest:=$myRequest+" AND [初飛行日]<=" + String($_year{$i}) + "-12-31' )"
End for
$myRequest:=$myRequest+" AND [エンジン数]=4"
QUERY BY SQL([航空機];$myRequest)
```

单一のリクエスト

# クエリの連結

```
QUERY([航空機];[航空機]カテゴリー="Land")
If ($nbOfSeatsMin#0)
    QUERY SELECTION([航空機];[航空機]座席数>=$nbOfSeatsMin)
        If ($nbOfSeatsMin#0)
            QUERY ([航空機];[航空機]座席数>=$nbOfSeatsMin;*)
        End if
        If ($nbOfSeatsMax#0)
            QUERY ([航空機];[航空機]座席数<=$nbOfSeatsMax;*)
        End if
        If ($nbOfEngines#0)
            QUERY ([航空機];[航空機]座席数=$nbOfEngines;*)
        End if
    QUERY([航空機];[航空機]カテゴリー="Land")
```

1~4リクエスト

单一リクエスト

# クエリの連結

```
QUERY([航空機];[航空機]カテゴリー="Land";*)
If ($nbOfSeatsMin#0)
    QUERY([航空機];[航空機]座席数>=$nbOfSeatsMin;*)
End if
If ($nbOfSeatsMax#0)
    QUERY([航空機];[航空機]座席数<=$nbOfSeatsMax;*)
End if
If ($nbOfEngines#0)
    QUERY([航空機];[航空機]エンジン数=$nbOfEngines;*)
End if
QUERY([航空機];[航空機]ID#0)
```

無駄なインデックス検索

# クエリの連結

v3以来

```
QUERY([航空機];[航空機]カテゴリー="Land";*)
If ($nbOfSeatsMin#0)
    QUERY([航空機];[航空機]座席数>=$nbOfSeatsMin;*)
End if
If ($nbOfSeatsMax#0)
    QUERY([航空機];[航空機]座席数<=$nbOfSeatsMax;*)
End if
If ($nbOfEngines#0)
    QUERY([航空機];[航空機]エンジン数=$nbOfEngines;*)
End if
QUERY([航空機])
```

単一リクエスト

クエリ実行

v3以来

# Case文の簡略化

## Case of

```
: ($operator="=")
    QUERY([航空機];[航空機]座席数=$nbOfSeats)
: ($operator=">")
    QUERY([航空機];[航空機]座席数>$nbOfSeats)
: ($operator="<")
    QUERY([航空機];[航空機]座席数<$nbOfSeats)
:($operator=">=")
    QUERY([航空機];[航空機]座席数>=$nbOfSeats)
: ($operator="<=")
    QUERY([航空機];[航空機]座席数<=$nbOfSeats)
: ($operator="#")
    QUERY([航空機];[航空機]座席数#$nbOfSeats)
```

End case

**QUERY([航空機];[航空機]座席数;\$operator;\$nbOfSeats)**

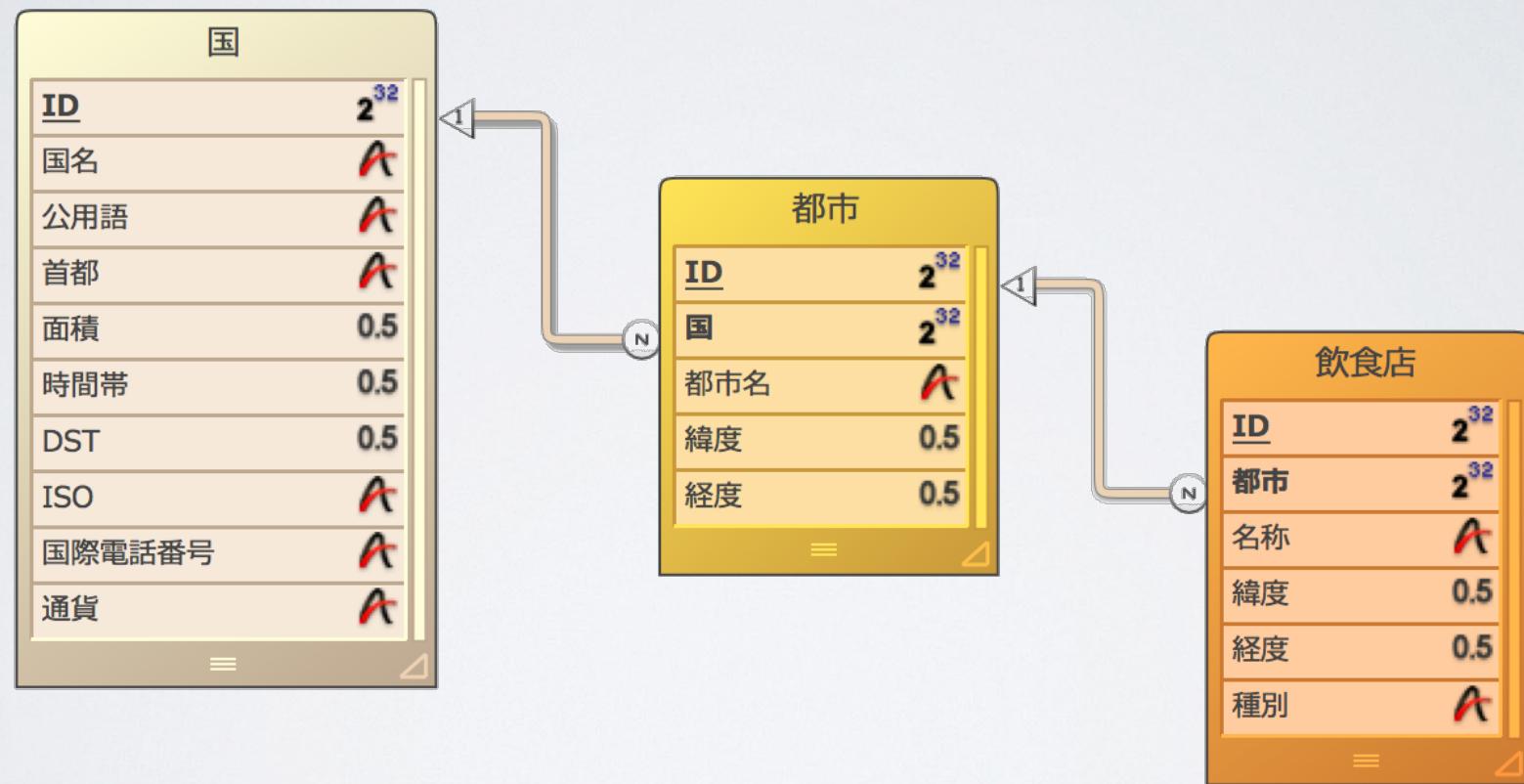


# ループ内でクエリ

```
// 地域の国を検索
QUERY([国];[国]地域="欧洲")
DISTINCT // 地域の国を検索
QUERY([国];[国]地域="欧洲")
// その後...
DISTINCT VALUES([国]ID;<>id_country)
// 史跡を検索
QUERY([史跡];[史跡]種別="廃墟")
// 地域の史跡を検索
RELATE ONE SELECTION([史跡];[都市])
$nb_of_countries = 0
QUERY([都市];[都市]国="中国")
For ([都市]Country; $nb_of_countries) {
    QUERY SELECTION([都市], | , [都市]国=<>id_country{$nb_of_countries})
}
End for
QUERY SELECTION([都市]; | ;[都市]国=<>id_country{$nb_of_countries})
```

# リレーションとクエリ

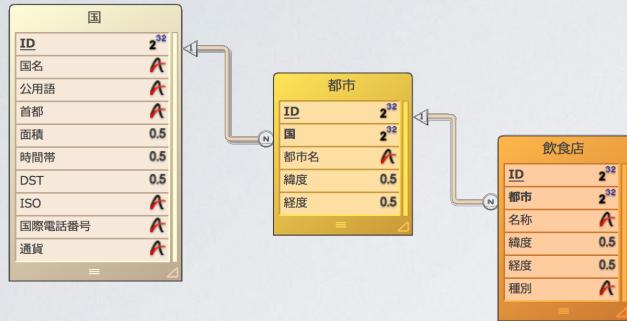
フランス国内の飲食店すべてが知りたい



# リレーションとクエリ

v11以来

フランス国内の飲食店すべてが知りたい



3リクエスト  
+3カレントセレクション

```
QUERY([国];[国]国名="フランス")
RELATE MANY SELECTION([都市]国)
RELATE MANY SELECTION([飲食店]都市)
```

```
QUERY([飲食店];[国]国名="フランス")
```

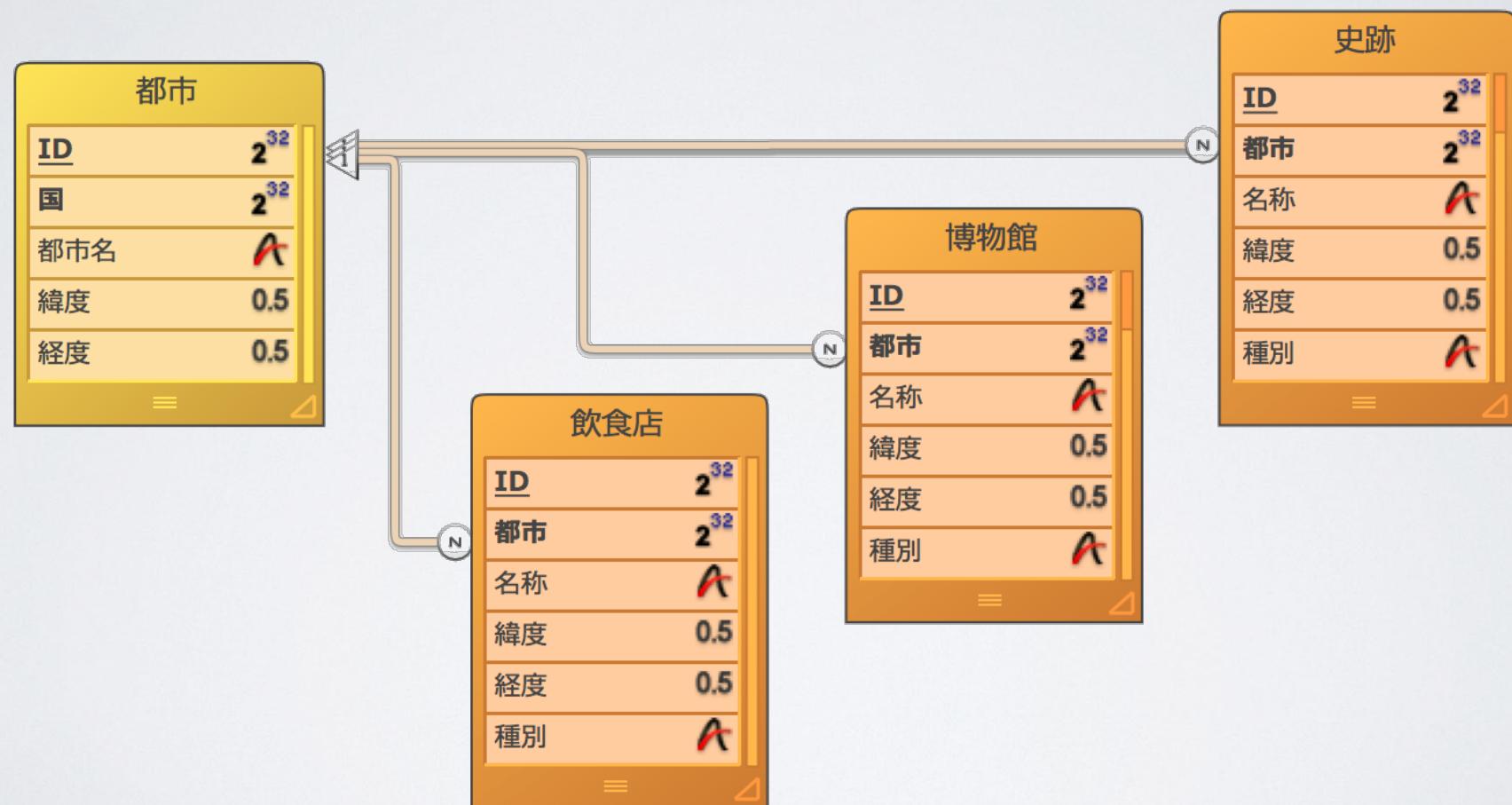
単一リクエスト

インデックス使用

カレントセレクションを改変  
するのは[飲食店]のみ

# 複数のリレーションテーブル

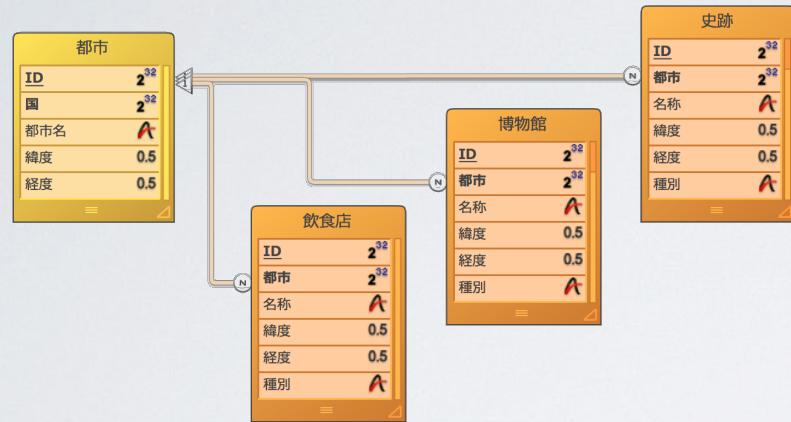
名称に『ハウス』が含まれる飲食店、博物館または史跡が所在する都市が知りたい



# 複数のリレーションテーブル

v11以来

名称に『ハウス』が含まれる飲食店、博物館または史跡が所在する都市が知りたい



**QUERY**([都市];[飲食店]名称=\$search;\*)  
**QUERY**([都市]; | [博物館]名称=\$search;\*)  
**QUERY**([都市]; | [史跡]名称=\$search)

単一リクエスト  
カレントセレクションを改変するのは[都市]のみ

\$search:="@ハウス@"

**QUERY**([飲食店];[飲食店]名称=\$search)  
**RELATE ONE SELECTION**([飲食店];[都市])  
**CREATE SET**([都市];"飲食店")

**QUERY**([博物館];[Museum]Name=\$search)  
**RELATE ONE SELECTION**([博物館];[都市])  
**CREATE SET**([都市];"博物館")

**QUERY**([史跡];[史跡]名称=\$search)  
**RELATE ONE SELECTION**([史跡];[都市])  
**CREATE SET**([都市];"史跡")

**CREATE EMPTY SET**([都市];"都市")  
**UNION**("飲食店";"博物館";"都市")  
**UNION**("都市";"史跡";"都市")  
**USE SET**("都市")  
**CLEAR SET**("都市")  
**CLEAR SET**("飲食店")  
**CLEAR SET**("博物館")  
**CLEAR SET**("史跡")

# JOINでクエリ



# JOINでクエリ



```
QUERY([史跡];[史跡]種別="廃墟")  
CREATE SET([史跡];"廃墟")
```

```
USE SET("廃墟") // [史跡]  
SELECTION TO ARRAY([史跡]都市;$_uuid_city)  
QUERY WITH ARRAY([都市]ID;$_uuid_city)
```

185,430バイト入力  
236,278バイト出力  
27ミリ秒/サーバー処理

```
USE SET("廃墟") // [史跡]  
DISTINCT VALUES([史跡]都市;$_uuid_city)  
QUERY WITH ARRAY([都市]ID;$_uuid_city)
```

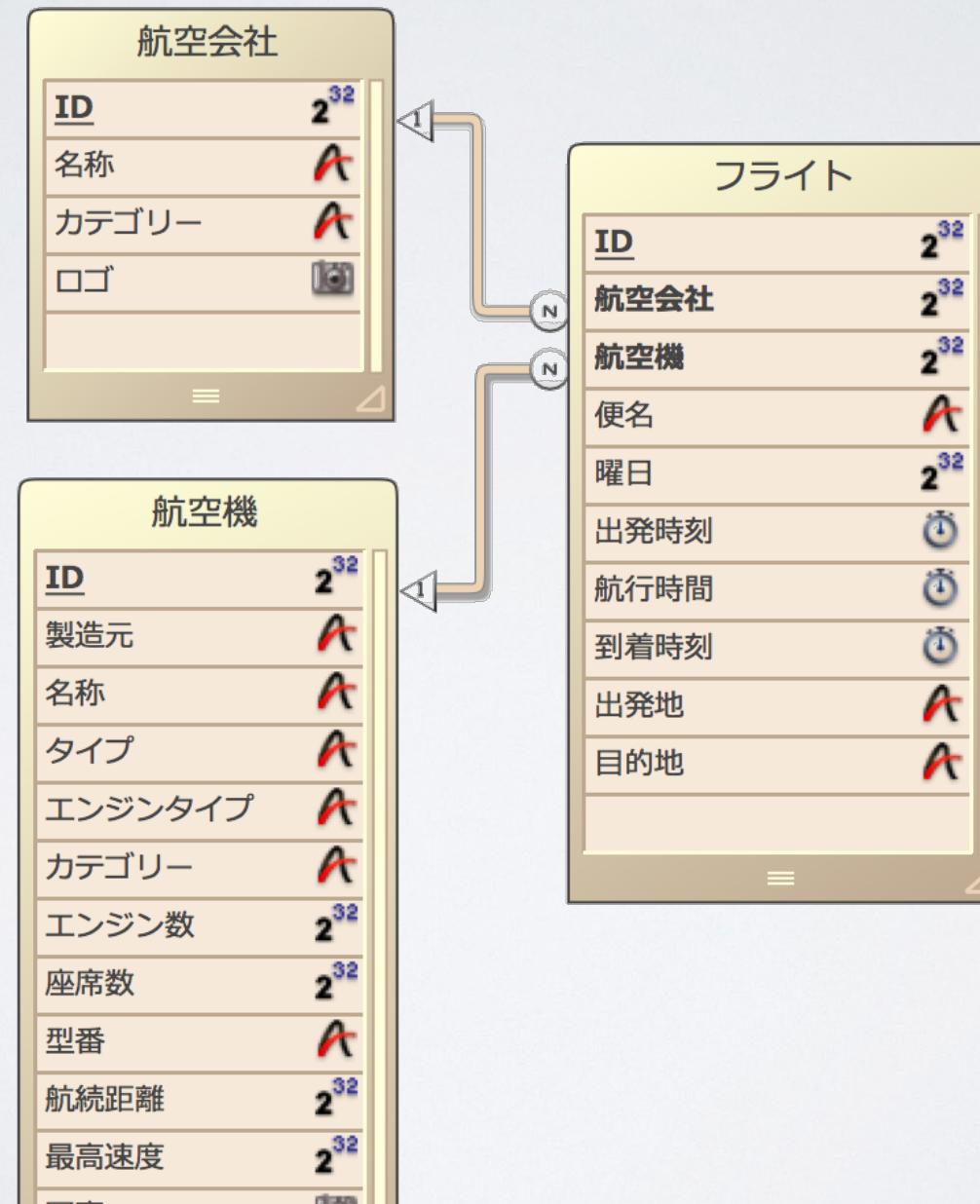
35,728バイト入力  
164,874バイト出力  
21ミリ秒/サーバー処理

```
USE SET("廃墟") // [史跡]  
RELATE ONE SELECTION([史跡];[都市])
```

166バイト入力  
4,800バイト出力  
2ミリ秒/サーバー処理

# JOINでクエリ

San Franciscoに到着するAmerican Airline社の航空機すべてが知りたい



# JOINでクエリ

San Franciscoに到着するAmerican Airline社の航空機すべてが知りたい



QUERY([航空会社];[航空会社]名称="American Airlines@")

RELATE MANY SELECTION([フライト]航空会社)

QUERY SELECTION([フライト];[フライト]目的地="San Francisco@")

RELATE ONE SELECTION([フライト];[航空機])

QUERY BY FORMULA([航空機];\

([航空会社]名称="American Airlines@")\

& ([フライト]目的地="San Francisco@")\

& ([航空会社]ID=[フライト]航空会社)\

& ([フライト]航空機=[航空機]ID))

# QUERY BY FORMULA

v11以来

v11の最重要的変更点のひとつ。変換データベースでは有効化しなければならない。

The screenshot shows the 'Database Settings' dialog for version 11. The top navigation bar includes icons for General, Interface, Compiler, Database, Move, Backup, C/S, Web, SQL, PHP, Security, and Interoperability. The 'Web' section is expanded, showing several checkboxes related to web functionality. The 'Database' section is also expanded, showing a checkbox for using 'PRINT SELECTION' to reload forms. A large green checkmark is present on the left side of the dialog, indicating that 'QUERY BY FORMULA' is enabled. A message box at the bottom states '新規データベースはデフォルトで有効 (>11.2)' (New databases are enabled by default (>11.2)).

2003 - データベース設定

一般 インターフェース コンパイラーデータベース 移動 バックアップ C/S Web SQL PHP セキュリティ 互換性

▼ Web

- ブラケットの代わりに4DVARコメントを使用する
- 新しいコンテキスト参照モードを使用しない
- 未知のURLから"/"を取り除く
- Web変数に値を自動的に代入する

▼ インターフェース

- ダイアログボックスでフィールドを入力可にする
- ラジオボタンを変数名でグループ化する
- 外部からのドラッグ&ドロップを許可しない

▼ クライアント/サーバー

- QUERY BY FORMULAをサーバーで実行する

▼ データベース

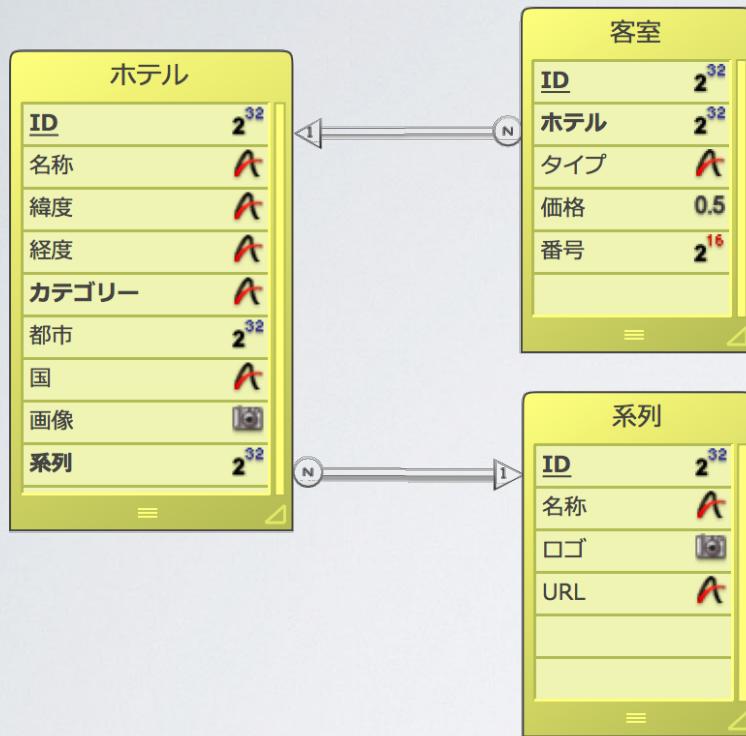
- PRINT SELECTIONでレコード毎にフォームをリロードする

SET DATABASE PARAMETER(Query by formula joins;2)

新規データベースはデフォルトで有効 (>11.2)

初期設定 OK

# QUERY BY FORMULA



QUERY([系列];[系列]名称="Marriott")

検索条件

QUERY([ホテル];[ホテル]系列=[系列]ID)  
DISTINCT VALUES([ホテル]ID;\$\_uuid)  
QUERY WITH ARRAY([客室]ホテル;\$\_uuid)

QUERY BY FORMULA([客室];\

([ホテル]系列=[系列]ID)\

& ([客室]ホテル=[ホテル]ID))

JOIN定義

DESCRIBE QUERY EXECUTIONが  
何も返さない

\$uuid\_chain:=[系列]ID

QUERY BY FORMULA([客室];\

([ホテル]系列=\$uuid\_chain)\

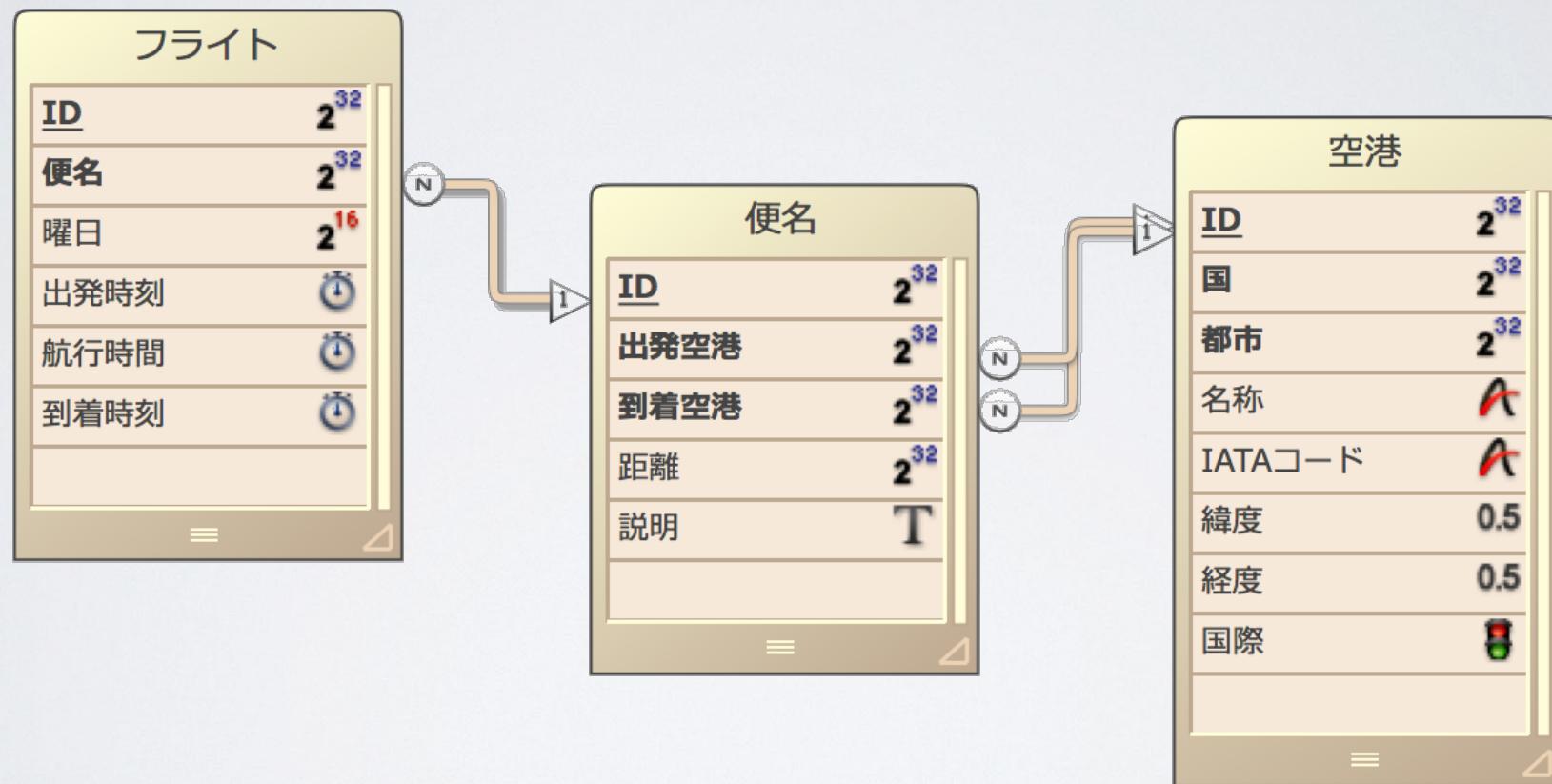
& ([客室]ホテル=[ホテル]ID))

検索条件

DESCRIBE QUERY EXECUTIONで  
有効なパスが返される

# 複数の経路

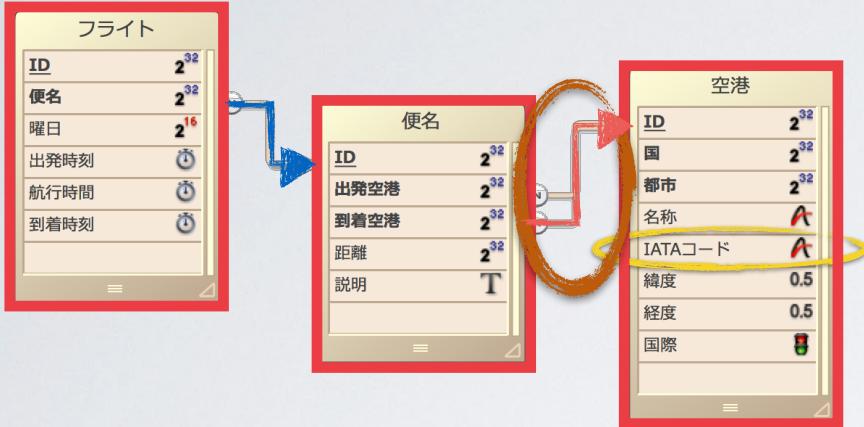
San Franciscoに到着するフライトすべてが知りたい



# 複数の経路

v11以来

San Franciscoに到着するフライトすべてが知りたい



3リクエスト

```
QUERY([空港];[空港]IATAコード="SFO")  
RELATE MANY SELECTION([便名]到着空港)  
RELATE MANY SELECTION([フライト]便名)
```

**QUERY([フライト],[空港])** AIAコード = "SFO")

単一リクエスト

```
QUERY BY FORMULA([フライト];([空港]IATAコード="SFO")\\
& ([フライト]便名=[便名]ID) & ([便名]到着空港=[空港]ID))
```

# 経路を検証するには?

## DESCRIBE QUERY EXECUTION(True)

```
QUERY BY FORMULA([フライト];([空港]IATAコード="SFO")\
& ([フライト]便名=[便名]ID) & ([便名]到着空港=[空港]ID))
```

```
$path:=Get last query path(Description in text format)
```

```
SET TEXT TO PASTEBOARD($path)
```

## DESCRIBE QUERY EXECUTION(False)

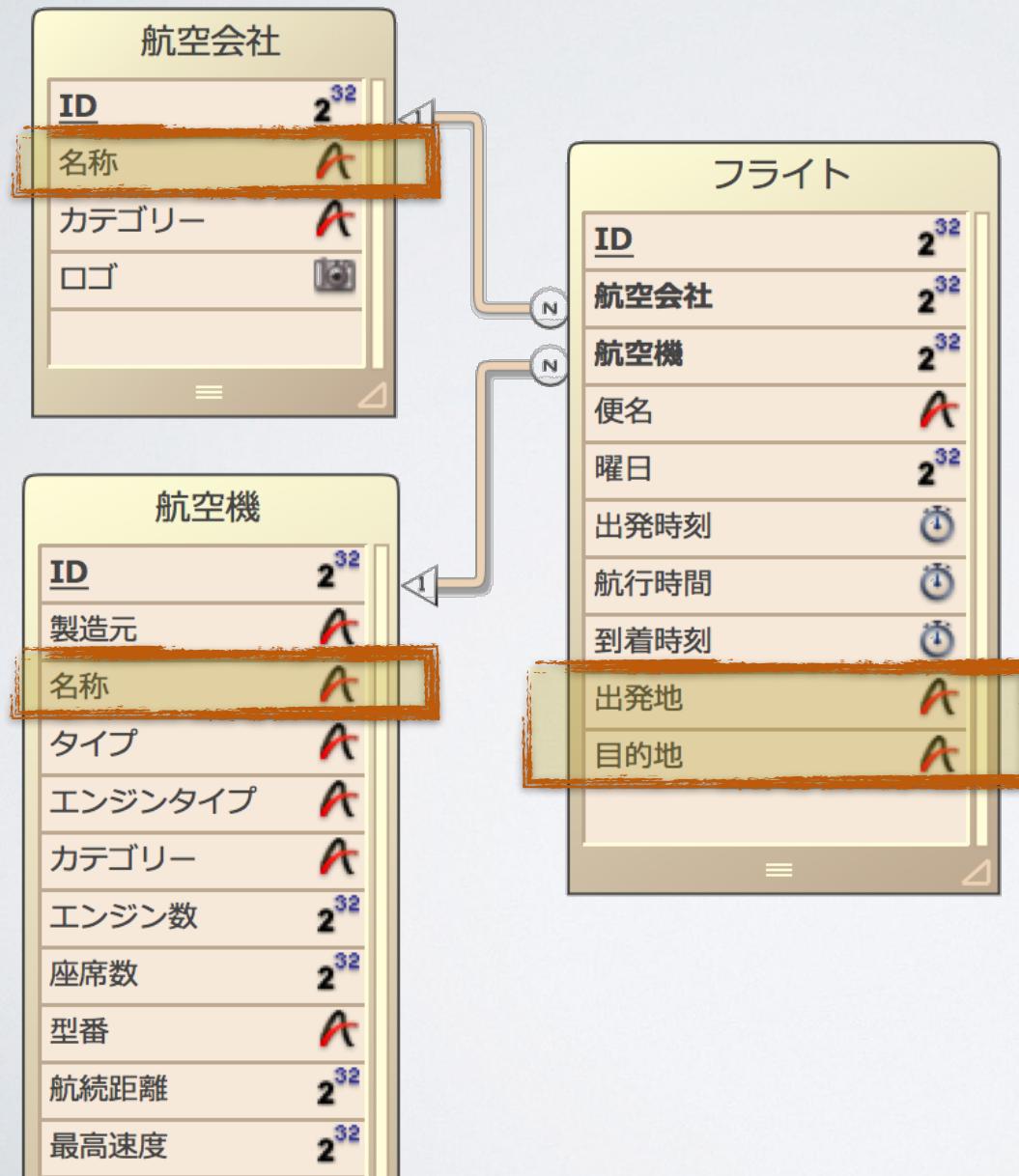
```
Join on Table : Line : フライト.便名 = 便名.ID
Join on Table : 空港 : 便名.到着空港 = 空港.ID
[index : 空港.IATAコード] = SFO (1 record found in 0 ms)
--> 142 records found in 0 ms
--> 442 records found in 4 ms
```

# リレーションのルール

ルール#1: すべてのリレーションはマニュアルモードに設定する

ルール#2: 設定に確信が持てないときにはルール#1を適用する

# セレクションから配列に

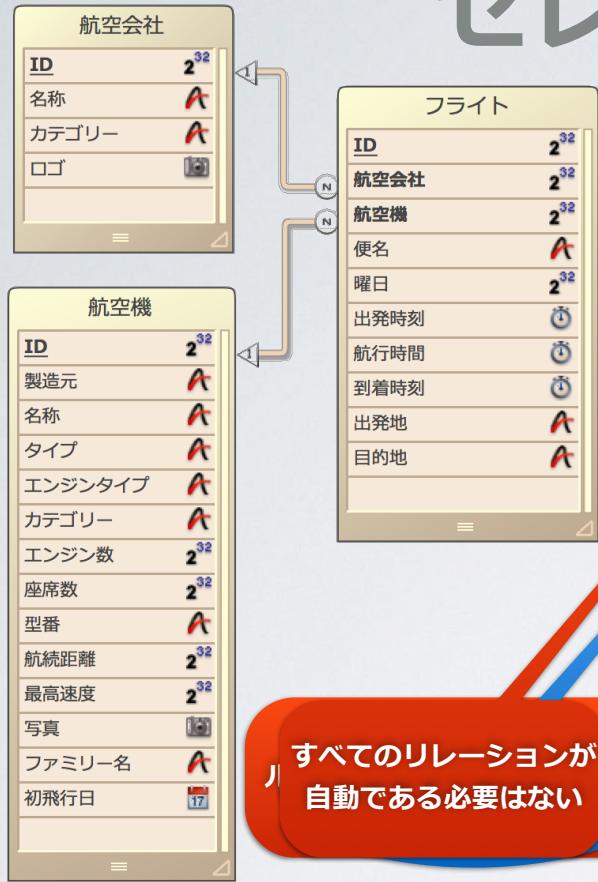


配列に表示したいフィールド:

- 出発地
- 目的地
- 航空会社名
- 航空機名

2004以来

# セレクションから配列に



```

SELECTION TO ARRAY([フライト]出発地;$出発地; \
SET FIELD RELATION([フライト]To:$目的地); \
SET AUTOMATIC([航空機];$航空機;Automatic;Do not modify) \
SELECTION TO ARRAY([フライト]航空機;$航空機ID; \
SET FIELD RELATION([フライト]航空会社:$航空会社ID); \
SET AUTOMATIC([航空機];$航空機;Automatic;Do not modify) \
SELECT [フライト]航空会社:$航空会社ID; \
SELECT [フライト]出発地:$出発地; \
$nb:=Size of array($目的地); \
[フライト]船舶機:$船舶機ID; \
ARRAY TEXT($航空機;$nb); \
ARRAY TEXT($航空会社;$nb); \
For ($i:1:$nb:1) \
[航空機]名稱:$航空機航空会社; \
QUERY([航空機];[航空機]ID=$航空機ID{$i}); \
SET AUTOMATIC([航空機];$航空機;Structure configuration; \
SET FIELD RELATION([フライト]航空機:$航空機;Structure configuration; \
QUERY([航空会社];[航空会社]ID=$航空会社ID{$i}); \
$航空会社{$i}:=[航空会社]名称; \
SET FIELD RELATION([フライト]航空会社:$航空会社;Structure configuration; \
End for \
Do not modify)
  
```

すべてのリレーションが  
自動である必要はない  
要発生

# SET FIELD RELATION

- 必要なコマンド

- SELECTION TO ARRAY
- ORDER BY

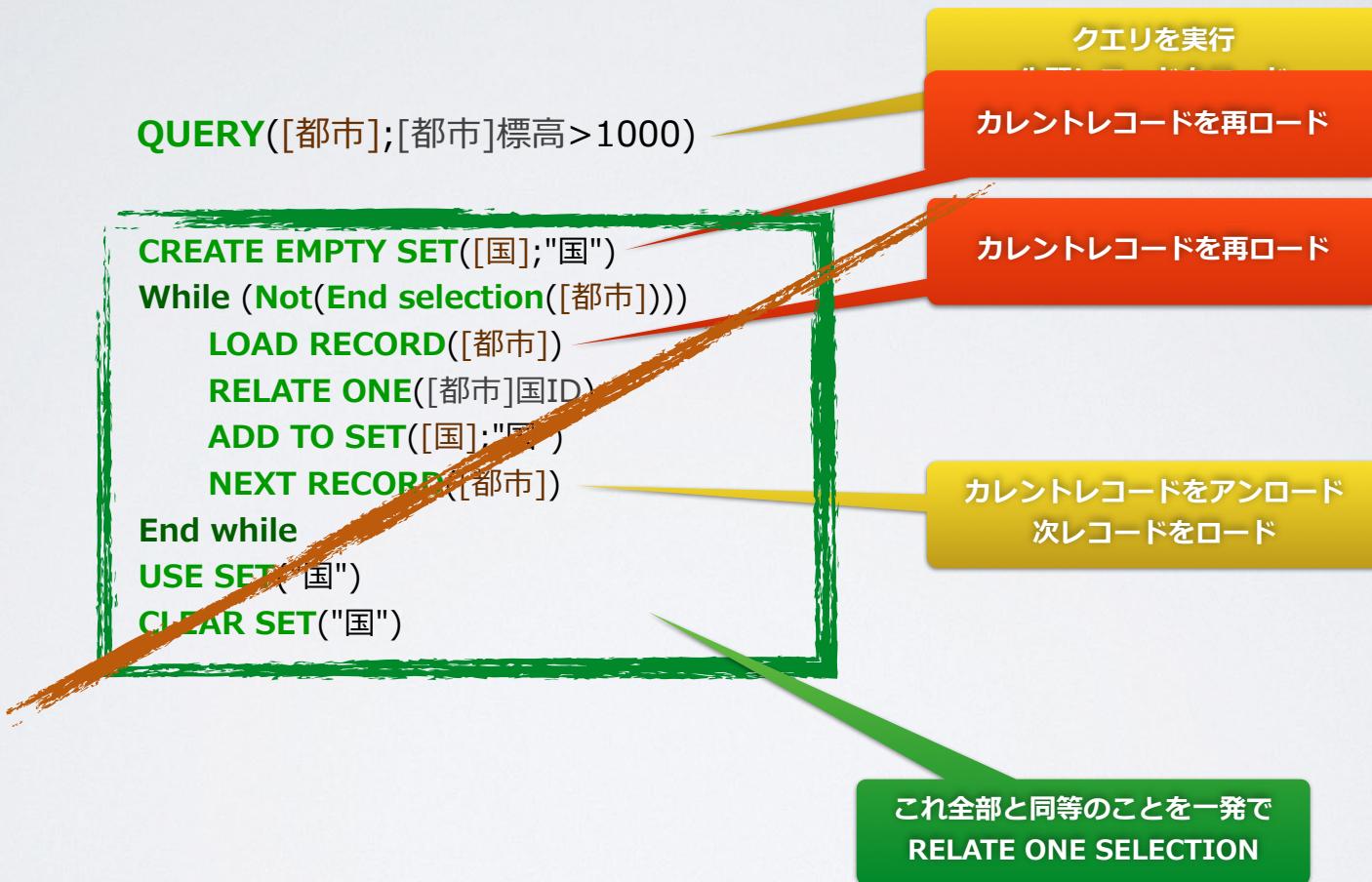
- 不要なコマンド

- QUERY

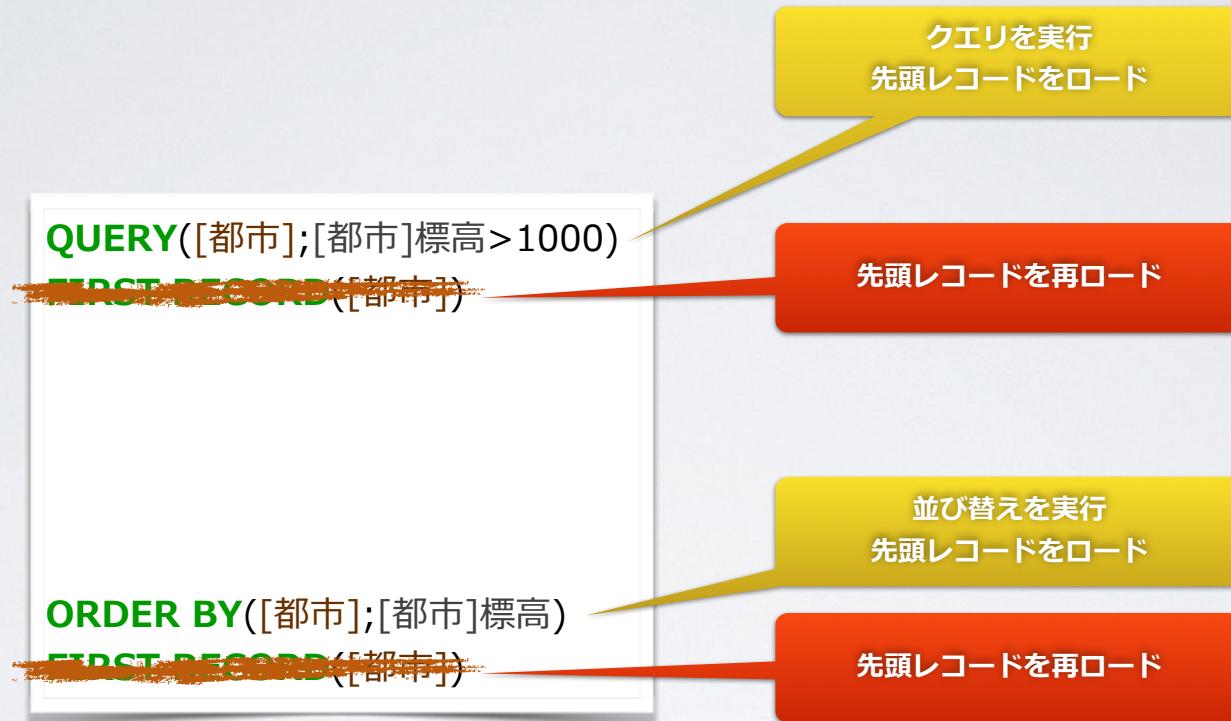
以下の表に掲げたコマンドは、コマンドの実行中にリレート先のリレートレコードをロードするために自動リレートを使用します。これらのコマンドはすべてn対1の自動リレートを使用します。1対nの欄が○になっているコマンドのみが自動の1対nリレートに対応します。

コマンド	1対n
ADD RECORD	○
ADD SUBRECORD	×
APPLY TO SELECTION	×
DISPLAY SELECTION	×
EXPORT DIF	×
EXPORT SYLK	×
EXPORT TEXT	×
EXPORT DATA	×
MODIFY RECORD	○
MODIFY SUBRECORD	×
MODIFY SELECTION	○(データ入力中)
ORDER BY	×
ORDER BY FORMULA	×
QUERY BY FORMULA	○
QUERY SELECTION	○
QUERY	○
PRINT LABEL	×
PRINT SELECTION	○
QR REPORT	×
SELECTION TO ARRAY	×
SELECTION RANGE TO ARRAY	×

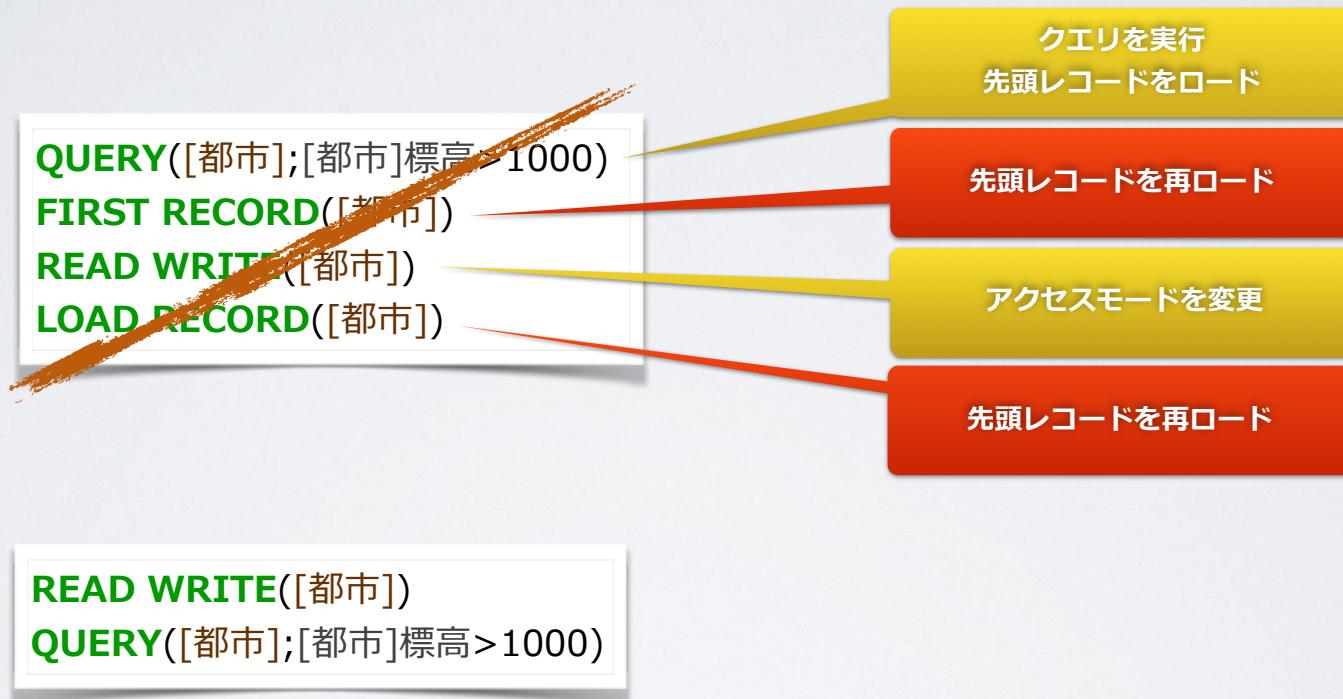
# 冗長なデータ処理



# 無駄な処理



# 出鱈目なコマンドの実行順序



# 無意味な汎用メソッド

```
QUERY([都市];[都市]人口>100000)
While (Not(End selection([都市])))
    Load_Record (->[都市])
    [都市]名称:=Uppercase([都市]名称)
    SAVE RECORD([都市])
    UnLoad_Record (->[都市])
    NEXT RECORD([都市])
End while
```

```
// メソッド : Load_Record
C_POINTER($1)
```

```
READ WRITE($1->)
LOAD RECORD($1->)
```

ループの実行ごとに4リクエスト

```
// メソット : UnLoad_Record
C_POINTER($1)
```

```
READ ONLY($1->)
UNLOAD RECORD($1->)
```

```
READ WRITE([都市])
QUERY([都市];[都市]人口>100000)
While (Not(End selection([都市])))
    [都市]名称:=Uppercase([都市]名称)
    SAVE RECORD([都市])
    NEXT RECORD([都市])
End while
READ ONLY([都市])
```

ループの実行毎に2リクエスト

# 過剰なセレクション

**COPY NAMED SELECTION([ホテル];"選択したホテル")**

カレントセレクションをメモリにコピー  
命名セレクションを作成

**QUERY([ホテル];[客室]価格<70)**  
**\$average\_price:=Average([客室]価格)**

カレントセレクションを破棄  
クエリを実行  
カレントセレクションを作成

**USE NAMED SELECTION("選択したホテル")**  
**CLEAR NAMED SELECTION("選択したホテル")**

カレントセレクションを破棄  
命名セレクションを復元

メモリを解放

**CUT NAMED SELECTION([ホテル];"選択したホテル")**

カレントセレクションを命名  
同時にカレントセレクションは消滅

**QUERY([ホテル];[客室]価格<70)**  
**\$average\_price:=Average([客室]価格)**

クエリを実行  
カレントセレクションが作成される

**USE NAMED SELECTION("選択したホテル")**

命名セレクションをカレントに転用  
同時に命名セレクションは消滅  
(解放不要!)