

Shazam4.0 User's Guide

エレクトロデザイン株式会社

2019.1.22

内容

1	初めに.....	3
2	Shazam Data Model.....	4
3	Shazam Power Query.....	4
3-1	Shazam Power Query.....	5
3-1-1	Summarize Data.....	5
3-1-2	Format Dates.....	7
3-1-3	Format Numbers.....	9
3-2	テーブルを使った作業.....	12
3-2-1	Add Tables.....	12
3-2-2	Rename Fields.....	13
3-2-3	Rename Table.....	13
3-2-3	Remove Table.....	14
3-2-5	Move/Resize Tables.....	14
3-2-6	Move/Resize Tables.....	14
3-3	複数のテーブルを使った作業.....	14
3-3-1	Add Join.....	14
3-3-2	Edit Join.....	16
3-3-3	Remove Join.....	18
3-3-4	同じテーブルの複数のコピーを使用する.....	18
3-4	値とフィルタについて.....	19
3-4-1	Add Value.....	19
3-4-2	Edit Value.....	20
3-4-3	Edit Filter.....	22
3-4-4	Edit Expression.....	24

1 初めに

このヘルプファイルは **Shazam Power Query (SPQ)** と **Shazam Report Wizard (SRW)** をユーザの視点から説明しています。 **Shazam** を使用して、ソフトウェアアプリケーションからアドホッククエリやレポートを即座に作成できます。

Shazam は、リレーショナルデータベースの言語である **SQL** の知識がなくても必要な情報を記述するのに役立つビジュアルツールを提供しています。 **Shazam** はビジュアルクエリを **SQL** コードに変換し、それを処理のためにデータベースエンジンに渡します。

Shazam Power Query と **Shazam Report Wizard** の優れた点は、パワー、柔軟性、および使いやすさです。データベースから意味のある情報を簡単に見つけ出すのに役立つツールを提供することが **ShazamWare Data Systems, Inc** の使命です。

F1 キーを押すと、製品のすべての主要画面から状況依存ヘルプが表示されます。追加情報とよくある質問は、以下から入手できます。

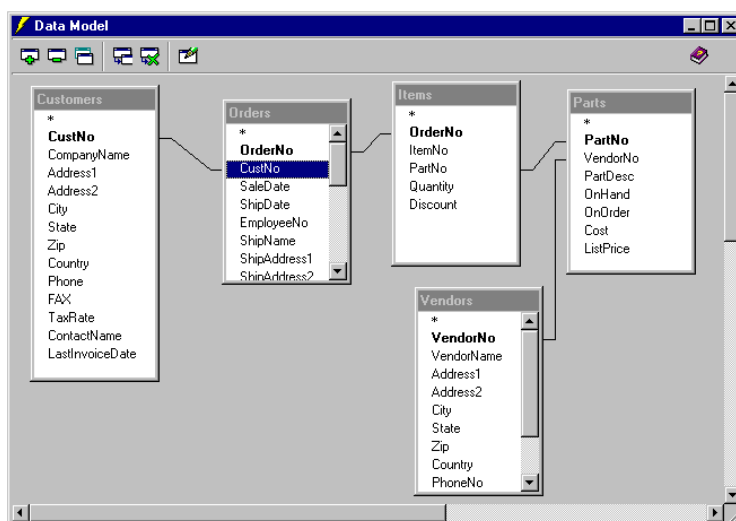
http : ~~¥¥~~ www.shazamware.com

Shazam は高度にモジュール化されているので、このヘルプファイルに記述されている特定のオプションがアプリケーションにインストールされていないかもしれません。必要なオプションを追加できるかどうかはソフトウェア開発者に問い合わせ下さい。

ShazamWare Data Systems, Inc はこのプログラムに対するサポートは行っていません。

2 Shazam Data Model

データベースのモデルは下記のようなものです。



データモデルはテーブルとフィールド、エイリアスと結合に関する情報を格納します。エイリアスは、テーブルとフィールドをわかりやすくするためにテーブルとフィールドに付けられる代替名です。結合

Query | Layout | SQL | Preview

QSLCARD

- *
 - CALL
 - QISODate
 - Year
 - Month
 - Day
 - QSOTime
 - L_QSONR
 - Duration
 - Band
 - QIRG
 - M-17

Header	QSLSortKey	CALL	NAME
Table	QSLCARD	QSLCARD	QSLCARD
Field	QSLSortKey	CALL	NAME
Show	Show	Show	Show
Sort	Ascending		

はテーブル間の関係です。Shazam Data Model はデフォルトの結合を保存します。これは Shazam Power Query にテーブルが追加されると自動的に適用されます。

SWISSLOG では QSLCARD というテーブルが使われています。

QSL 印刷ではこの部分は新たに作成するより、既存のファイルを編集して使用します。

注：テーブル名とフィールド名、またはデフォルトの結合情報を変更するには、テーブルを選択するか結合を選択してマウスの右ボタンを押すと、いつでもオプションを確認できます。

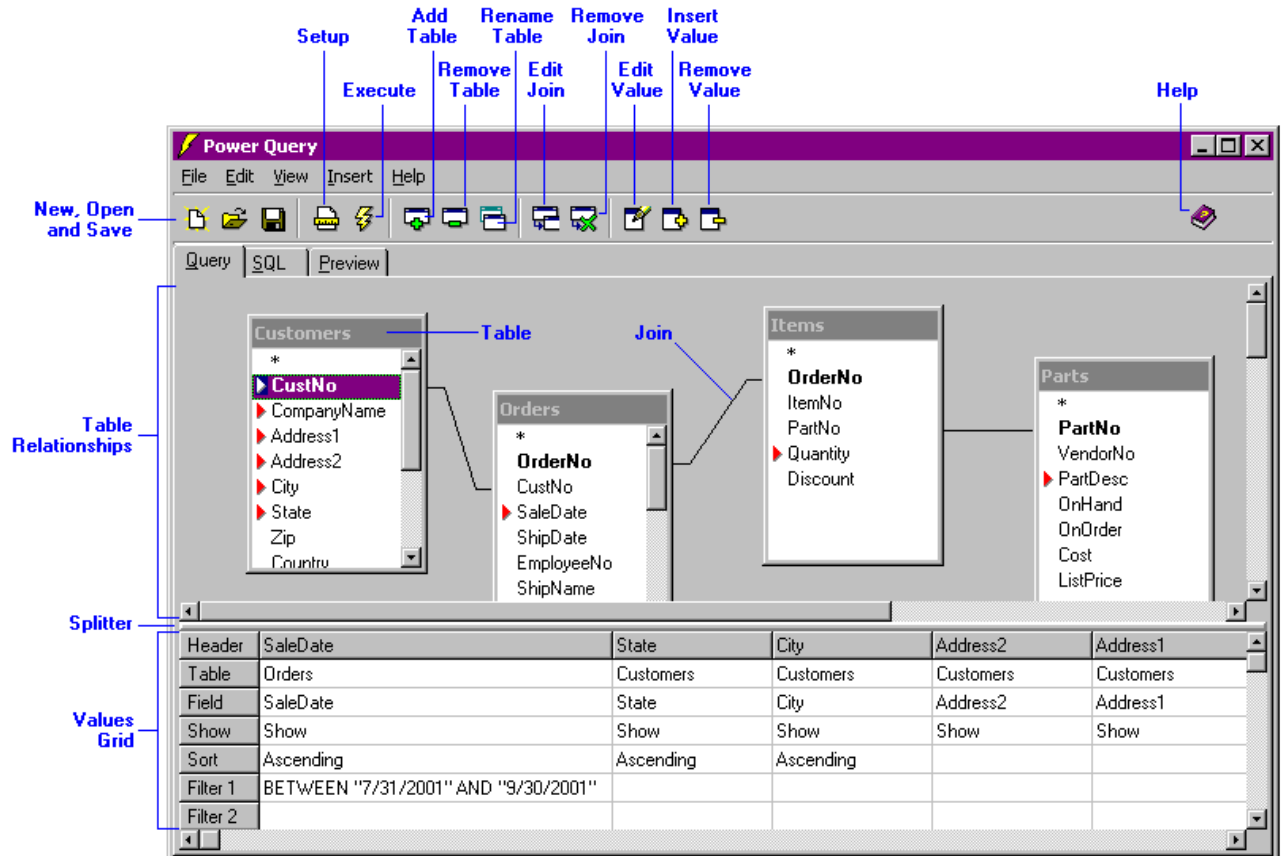
3 Shazam Power Query

Shazam Power Query は、手掛かりと一致する証拠を求めて国側を検索する探偵のようなものです。ユーザとしてのあなたの仕事は、謎を解くのに必要な手がかりを提供することです。これは、実際には、Query-By-Example (QBE) という名前が付けられたところです。どのレコードを検索に含めたり除外したりする必要があるかを示すデータの例を示します。

注意：QBE と SQL は、1970 年代に競合する可能性のある標準として開発されました。時間が経つにつれて、それらは補完的になりました。SQL は現在データベースを検索するための業界標準言語ですが、QBE は SQL コードを素早く簡単に生成するための最も一般的な手法です。

3-1 Shazam Power Query

Shazam Power Query は、コンピュータ言語によらず結果が欲しいユーザのために強力な QBE インターフェースを実装しています。下の画像をクリックすると、関連トピックにジャンプします。



3-1-1 Summarize Data

サマリークエリについて

Show プロパティは、レポートにフィールドをどのように表示するかを決定します。有効な Show プロパティ値は次のとおりです。

Show : レポートにフィールドを表示します。これがデフォルト設定です。

Group : 1 つ以上のフィールドの Show [表示] の設定が Sum [合計]、Count [件数]、Average [平均]、Minimum [最小]、または Maximum [最大] の場合に使用されます。これらの設定のいずれも持たないどんなフィールドも Group に設定されなければなりません。この情報は、レポートの 1 つ以上のフィールドに [表示] の [合計]、[件数]、[平均]、[最小]、または [最大] が設定されている場合にのみ適用されます。

Hide : レポートにフィールドが表示されないようにします。フィールドを検索する必要があるがそのフィールドを表示したくない場合は、非表示を使用してください。例 : **Company** が "Unisco" に等しいすべての注文を表示します。会社は 1 つだけなので、会社名を繰り返す必要はほとんどありません。

Sum : レコードセット内のすべての値を合計します (フィールドは数値である必要があります)。

Count : レコードセット内の発生数をカウントします。

Average : レコードセットの平均値を表示します。

Minimum : レコードセットの最小値を表示します。

Maximum : レコードセットの最大値を表示します。

集計計算で表示を理解する

Sum、**Count**、**Average**、**Minimum**、および **Maximum** の各値は集約関数と呼ばれます。集約関数を持つ列で使用されるフィルター値は、異なる方法で処理されます。例えば :

Maximum : レコードセットの最大値を表示します。

「**AmountPaid** が 500 を超えた顧客を表示する」は、500 を超える **AmountPaid** を超える **RECORDS** を表示します。

「**AmountPaid** の合計が 500 を超えた顧客を表示する」は、**AmountPaid** の総計 **TOTAL** が 500 を超えるレコードを表示します。

最初の例は、レコードの実際の値に対してフィルタ値を適用します。2 番目の例は、計算がレコードのセットに対して実行された後に値を適用します。集約関数を使用したレポートでは、必要に応じて他のフィールドやフィルタ値を使用することができますが、それらには **Show** [表示] 設定の **Group** [グループ] または **Hide** [非表示] が必要です。

Show を支配する主な規則

Show プロパティを使用する際の注意点

表示されているだけで表示されていないフィールドを非表示にするには、表示値 "Hide" を使用します。

他の列で集計値を計算するには、"**Group**" の表示値を使用します (つまり、「**AmountPaid** が 500 を超える都市ごとにすべての顧客を表示する」)。**City** フィールドには「グループ」の表示設定があり、**AmountPaid** フィールドには「合計」の表示設定があります。

レポートに集計関数が含まれていない場合は、**Show** 設定の "**Group**" を使用しないでください。

3-1-2 Format Dates

Format Date プロパティは、**Date**、**Time**、または **DateTime** フィールドの表示方法を決定します。有効なフォーマットマスクは次のとおりです。

c

ShortDateFormat システム設定によって指定された形式を使用して日付を表示し、その後に **LongTimeFormat** システム設定によって指定された形式を使用して時刻を表示します。 **DateTime** 値の小数部がゼロの場合、時刻は表示されません。

d

先頭にゼロを付けない数値として日を表示します (1~31)。

dd

日を先行ゼロ付きの数字として表示します (01-31)。

ddd

ShortDayNamesystem 設定で指定された文字列を使用して、日を省略形 (Sun-Sat) として表示します。

dddd

LongDayNames システム設定で指定された文字列を使用して、日をフルネーム (日曜日から土曜日) として表示します。

dddddd

ShortDateFormat システム設定で指定された形式を使用して日付を表示します。

ddddddd

LongDateFormat システム設定で指定された形式を使用して日付を表示します。

m

月を先行ゼロなしの数値として表示します (1~12)。 **m** 指定子が **h** または **hh** 指定子の直後に続く場合は、月ではなく分が表示されます。

mm

月を先行ゼロ付きの数字として表示します (01-12)。 **mm** 指定子が **h** または **hh** 指定子の直後に続く場合は、月ではなく分が表示されます。

mmm

ShortMonthNames システム設定で指定された文字列を使用して、月を省略形 (Jan-Dec) で表示します。

mmmm

LongMonthNames システム設定で指定された文字列を使用して、月をフルネーム（1月から12月）として表示します。

yy

年を2桁の数字（00～99）で表示します。

yyyy

年を4桁の数字（0000～9999）で表示します。

h

先頭にゼロを付けずに時間を表示します（0～23）。

hh

時間を先行ゼロ（00-23）で表示します。

n

先頭にゼロを付けずに分を表示します（0～59）。

nn

分を先行ゼロ（00-59）で表示します。

s

秒を先行ゼロなしで表示します（0-59）。

ss

秒を先行ゼロ（00-59）で表示します。

t

ShortTimeFormat システム設定で指定された形式を使用して時間を表示します。

tt

LongTimeFormat システム設定で指定された形式を使用して時間を表示します。

am/pm

前の **h** または **hh** 指定子に12時間表示を使用し、正午までの任意の時間には「**am**」、正午以降の任意の時間には「**pm**」を表示します。**am / pm** 指定子は、小文字、大文字、または大/小文字混合を使用でき、それに応じて結果が表示されます。

a/ p

前の **h** または **hh** 指定子に 12 時間制を使用し、正午までの任意の時間には '**a**'を、正午以降の任意の時間には '**p**'を表示します。 **a / p** 指定子は、小文字、大文字、または大/小文字混合を使用でき、それに応じて結果が表示されます。

ampm

直前の **h** または **hh** 指定子に 12 時間表示を使用し、正午までの任意の時間の **TimeAMString** システム設定の内容と、正午以降の任意の時間の **TimePMString** システム設定の内容を表示します。

/

DateSeparator システム設定で指定された日付区切り文字を表示します。

:

TimeSeparator システム設定で指定された時間区切り文字を表示します。

'xx' / "xx"

一重引用符または二重引用符で囲まれた文字はそのまま表示され、書式設定には影響しません。

フォーマット指定子は、大文字でも小文字でも書くことができます - どちらも同じ結果になります。

Format パラメーターで指定されたストリングが空の場合、日付と時刻の値は、'**c**'フォーマット指定子が指定されているかのようにフォーマットされます。

3-1-3 Format Numbers

！ もし ! 文字がマスクに表示されたら、先頭の空白はデータに表示されません。 ! キャラクターが存在しない場合、末尾の空白はデータに表示されません。

> マスク内に>文字が表示されている場合、それ以降のすべての文字はマスクの最後まで大文字になります。 または<文字が見つかるまで。

< マスク内に<文字がある場合、それ以降のすべての文字はマスクの最後まで小文字で表示されます。 または>文字が見つかるまで。

<> これら 2 つの文字がマスクと一緒に現れると、大文字と小文字の区別は行われず、データはフォーマットされます。

\\ ユーザがデータを入力するために使用する場合、\に続く文字はリテラル文字です。データ内のリテラルとしてのマスク特殊文字のいずれかを許可したい場合はこの文字を使用してください。

L L 文字は、この位置にのみ英字だけを必要とします。 米国の場合、これは **A-Z**、**a-z** です。

I I 文字は、この位置にアルファベット文字だけを許可しますが、それを必須としません。

A **A** 文字には、この位置にのみ英数字が必要です。 米国の場合、これは **A** から **Z**、**a** から **z**、**0** から **9** です。

A **a** 文字はこの位置に英数字を使用できますが、使用する必要はありません。

C **C** 文字には、この位置に文字が必要です。

C **c** 文字はこの位置にある文字を許可しますが、必須ではありません。

0 **0** 文字には、この位置にのみ数字が必要です。

9 **9** 文字は、この位置に数字を許可しますが、必須ではありません。

**#** 文字では、この位置に数字またはプラスまたはマイナス記号を使用できますが、必須ではありません。

: **:** 文字は時間を、時、分、秒に区切る分離する文字として使用されます。

時間、分、秒は、コンピュータのコントロールパネルユーティリティの国際設定によって異なります。コンピュータシステムでは、その文字が代わりに使用されます。

/ **/** 文字は、日付で月、日、年を区切るために使用されます。

月、日、年は、コントロールパネルユーティリティの国際設定によって異なります。

コンピュータシステムでは、**/**の代わりにその文字が使用されます。

; **;** 文字はマスクを区切るために使用されます。

_ **_** 文字は、編集ボックスに自動的に空白を挿入します。ユーザがフィールドに文字を入力すると、カーソルは空白文字をスキップします。 **EditMask** プロパティエディタを使用するときは、

ブランクを表すために使用される文字。プログラムでこの値を変更することもできます。下記の表を参照下さい。

これらの文字（前の表で既に説明した）は、**Mask** ユニットで宣言された型付き定数であり、その値は実行時に変更することができますが、これに対する必要性は制限されるべきです：

Typed

Constant Initial value 意味

DefaultBlank **_** マスク内の空白は **_** 文字で表されます。

MaskFieldSeparator **;** ; 文字はマスクのフィールドを区切ります。

MaskNoSave **0** **0** 文字は、マスクがデータの一部として保存されないことを意味します。 **1** **1** の文字は、マスクがデータの一部として保存されることを意味します。たとえば、電話番号には、市外局番の一部として括弧を付けることができます。

MaskNoSave が 0 の場合、データの一部にはなりません。データのサイズが少し小さくなります。

```
procedure TForm1.FormCreate(Sender: TObject);
```

```
begin
```

```
    Mask.MaskFieldSeparator := ',';
```

```
    Mask.DefaultBlank := '@';
```

```
    MaskEdit1.EditMask := '999-999,1,@';
```

```
end;
```

Format プロパティは、数値フィールドの表示方法を決定します。有効なフォーマットマスクは次のとおりです。

0

数字のプレースホルダ。フォーマットされている値のフォーマット文字列の '0' が表示されている位置に数字がある場合は、その数字が出力文字列にコピーされます。それ以外の場合は、出力文字列のその位置に「0」が格納されます。

#

数字のプレースホルダ。フォーマットされている値のフォーマット文字列内の '#' が表示されている位置に数字がある場合は、その数字が出力文字列にコピーされます。そうでなければ、出力文字列のその位置には何も格納されません。

.

小数点。最初 '.' フォーマット文字列内の文字は、フォーマットされた値内の小数点の位置を決定します。追加の「.」文字は無視されます。出力文字列で小数点として使用される実際の文字は、DecimalSeparator システム設定によって決まります。のデフォルト値 DecimalSeparator は、Windows の[コントロールパネル]の[インターナショナル]セクションの[数値の形式]で指定されています。

,

千の区切り記号 フォーマット文字列に 1 つ以上の ';' 文字が含まれている場合、出力には小数点の左側の 3 桁の各グループの間に 1000 個の区切り文字が挿入されます。フォーマット文字列内の ';' 文字の配置と数は、千単位の区切り記号が必要であることを示す場合を除いて、出力には影響しません。出力で千単位の区切り文字として使用される実際の文字は、ThousandSeparators システム設定によって決まります。ThousandSeparator のデフォルト値は、Windows の「コントロールパネル」の「インターナショナル」セクションの「数値フォーマット」で指定されています。

E +

科学表記法 フォーマット文字列に文字列 'E +', 'E-', 'e +', または 'e-'のいずれかが含まれている場合、数値は科学表記法を使用してフォーマットされます。最大 4 文字の「0」文字のグループは、指数の最小桁数を決定するために、「E +」、「E -」、「e +」、または「e -」の直後に続けることができます。「E +'および 'e +'フォーマットでは、正の指数の場合はプラス記号が出力され、負の指数の場合は負符号が出力されます。「E-'および 'e-'フォーマットは、負の指数に対してのみ符号文字を出力します。

'xx' / "xx"


一重引用符または二重引用符で囲まれた文字はそのまま出力され、フォーマットには影響しません。

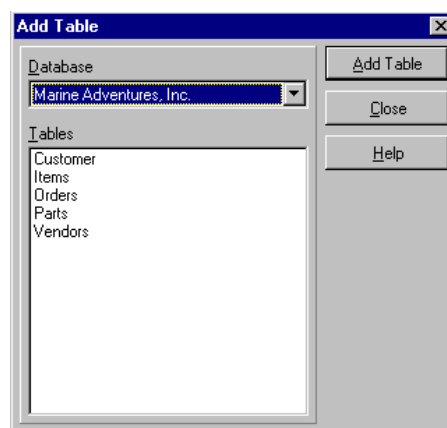
;

フォーマット文字列内で正数、負数、およびゼロ数のセクションを区切ります。

3-2 テーブルを使った作業

3-2-1 Add Tables

ツールバーの  ボタンを押して "Add Tables" [テーブルの追加] ダイアログを表示するか、"Edit | Add Tables" [編集 | テーブルの追加] メニューを使用します。以下のダイアログが表示されます。



"Add Table" [テーブルの追加] ダイアログは、次の要素で構成されています。

Database データベース

上部のコンボボックスに利用可能なデータベースが一覧表示されます。データモデルからこのダイアログにアクセスしている場合は、利用可能なすべてのデータベースが表示されます。**Power Query** または **Report Wizard** からこのダイアログにアクセスしている場合は、現在のデータモデルの名前が表示されません。

Tables テーブル

以下のいずれかを実行して、データモデル、クエリ、またはレポートにテーブルを追加できます。

- テーブル名をダブルクリック

- テーブル名を強調表示して[Enter]を押します。
- [Add Table] [テーブル追加]ボタンを押す

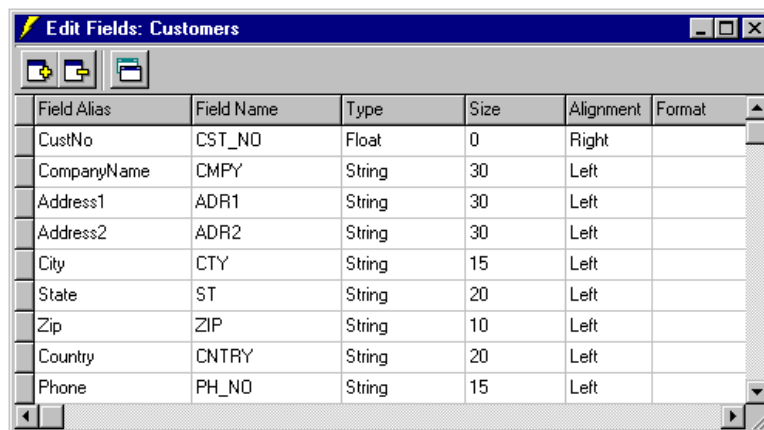
興味のあるテーブルをすべて追加したら、[閉じる]ボタンを選択して、テーブルを視覚的に意味のある順序に並べ替えます。

System Tables システムテーブル (データモデルのみ)

データモデルにテーブルを追加している場合は、**System Tables** [システムテーブル]チェックボックスが表示されることがあります。このボックスを選択すると、データベース管理システムが **Microsoft SQL Server** の "dbo.sysobjects"のように、そのテーブルシステムの一部と見なすテーブルが追加されます。データベース管理システムにシステムテーブルがない場合、このチェックボックスは無効になります。

3-2-2 Rename Fields

Rename Fields [フィールド名の変更]ダイアログにはデータモデルからアクセスできます。フィールド名のエイリアス、フィールドの物理フィールド名、データ型、その他の属性の変更ができます。テーブルを選択してマウスの右ボタンを押し、"**Edit Fields**" [フィールドの編集]を選択してこのダイアログを表示します。



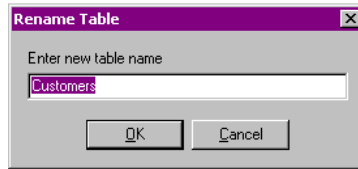
Field Alias	Field Name	Type	Size	Alignment	Format
CustNo	CST_NO	Float	0	Right	
CompanyName	CMPY	String	30	Left	
Address1	ADR1	String	30	Left	
Address2	ADR2	String	30	Left	
City	CTY	String	15	Left	
State	ST	String	20	Left	
Zip	ZIP	String	10	Left	
Country	CNTRY	String	20	Left	
Phone	PH_NO	String	15	Left	

3-2-3 Rename Table

Rename Table [テーブル名の変更]ダイアログを表示するには、テーブルを選択してマウスの右ボタンを押し、"**Rename Table.**" [テーブル名の変更]を選択します。テーブル名の変更ダイアログには2つの形式があります。

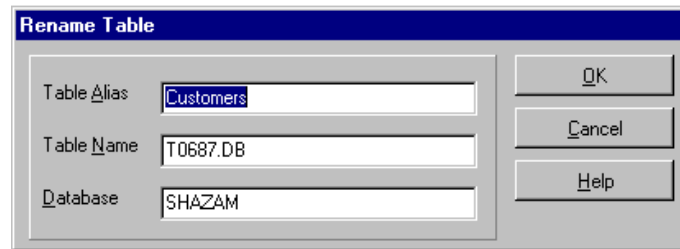
Power Query:

Rename Table [テーブル名の変更]ダイアログでは、現在のテーブルのテーブルエイリアスを設定できません。



Data Model :


Rename Table [テーブル名の変更]ダイアログでは、現在のテーブルの **Table Alias** [テーブルエイリアス], **Table Name**[テーブル名] と **Database**[データベース]を設定できます。



3-2-3 Remove Table

レポートからテーブルを削除するには、2つのステップがあります。

選択したテーブルの上でマウスの左ボタンを押して、削除するテーブルを選択します。

ツールバーの  ボタンを押してテーブルを削除するか、"Edit | Remove Table" [編集|テーブルの削除]メニューを使用するか、マウスの右ボタンを押して"Remove table." [テーブルの削除]を選択します。

3-2-5 Move/Resize Tables

テーブルを移動するには、テーブルのヘッダをマウスの左ボタンで押して移動します。

テーブルのサイズを変更するには、テーブルの両側にマウスを置き、マウスの左ボタンを押してサイズを変更します。

3-2-6 Move/Resize Tables

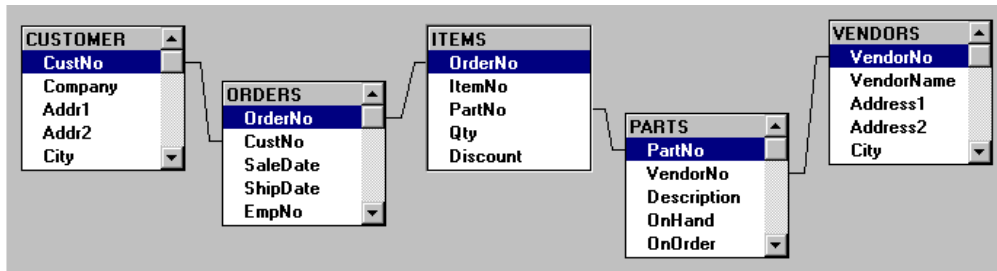
テーブルを移動するには、テーブルのヘッダをマウスの左ボタンで押して移動します。

テーブルのサイズを変更するには、マウスをテーブルの両側に置き、マウスの左ボタンを押してサイズを変更します。

3-3 複数のテーブルを使った作業

3-3-1 Add Join

Shazam Power Query はテーブルの関係を視覚的に表します。これにより、複雑な関係をはるかに理解しやすくなります。以下の図は、5つのテーブルの結合を表しています。

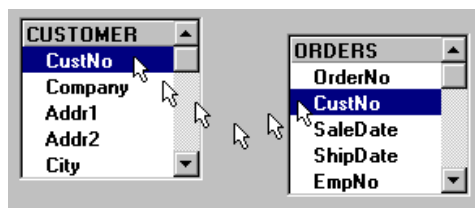


パフォーマンスに関する注：ほとんどのデータベース管理システムは、一度に1つの照会で7つを超える表を効率的に処理することはできません。1つのクエリに対してより多くのテーブルを処理する必要がある場合は、クエリを複数のステップに分割し、各ステップの結果を一時テーブルにエクスポートし、そのテーブルを後続のステップにリンクすることができます。

Joining tables

テーブルを結合するには、次のように関連フィールドをあるテーブルから別のテーブルにドラッグアンドドロップします。

- 1.左のテーブルのリンクフィールドをポイントしてマウスの左ボタンを押します。
- 2.マウスを右テーブルのリンク欄にドラッグします。
- 3.マウスの左ボタンを離します。



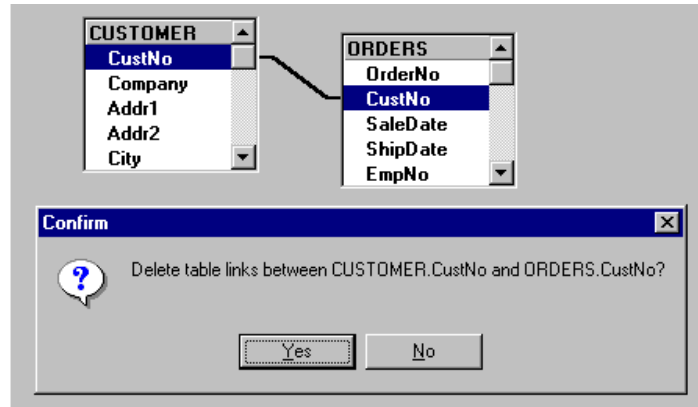
結合に複数のフィールドが必要な場合は、各フィールドについて上記のプロセスを繰り返します。

Removing Table Joins

クエリからテーブル結合を削除することは、2段階のプロセスです。

- 1.結合の上にマウスポインタを置いて削除する結合を選択し、マウスの左ボタンを押します。
- 2.ツールバーボタンを押してテーブルリンクを削除するか、[編集]、[結合の削除]メニューを使用します。

次の質問が表示されます：



Yes と回答して結合を削除します。クエリを実行する前に、テーブル結合を置き換えるか、テーブルを削除するようにしてください。それ以外の場合は、テーブルがリンクされていないとエラーが発生する可能性があります。関連付けられているテーブルをクエリから削除して、テーブル結合を削除することもできます。

OK

[OK]ボタンは、リンクオペレータとリンクタイプに加えた変更を受け入れて、"Link Tables" ダイアログを閉じます。

Cancel キャンセル

Cancel [キャンセル]ボタンをクリックすると、変更を確定せずに"Link Tables" ダイアログを閉じます。

Delete

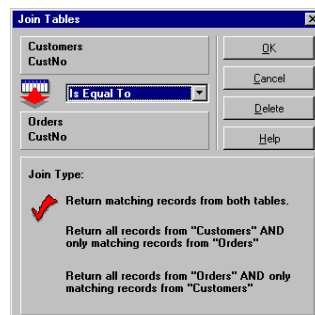
Delete [削除]ボタンをクリックすると、現在のリンクを削除して"Link Tables" ダイアログを閉じます。

Help

Help [ヘルプ]ボタンをクリックすると、"Link Tables" ヘルプトピックが表示されます。

3-3-2 Edit Join

通常、2つのテーブルを関連付けて、他に何もしなくてもテーブル結合のデフォルトの特性を受け入れることができます。デフォルト設定を変更する必要がある場合は、2つのテーブルの間の行をダブルクリックして"Join Tables" [テーブルの結合]ダイアログを表示します。



"Join Tables" [テーブルの結合]ダイアログは、次の要素で構成されています。

Join Operator

"Join Tables" [結合テーブル]ダイアログの左上部分には、現在選択されている結合演算子が表示されます。上記の例は次のようになります。**Customer** テーブルの **CustNo** フィールドは **Order** テーブルの **CustNo** フィールドと同じです。「等しい」という句は、2つの結合フィールド間の比較方法を決定します。

利用可能な選択肢は次のとおりです。

Join Operator	Symbol
等しい	=
大きい	>
等しいか大きい	>=
小さい	<
等しいか小さい	<=
同じで無い	<>

レポートが2つの列の値を比較し、その比較に基づいてレコードを選択する必要がある場合は、「等しい」以外の結合演算子が非常に役立ちます。この場合は、実際にはテーブル **Join** を検索フィルタとして使用しています。結合演算子は、リレーショナルデータベースに精通していて、これらの演算子がクエリに与える影響がある場合にのみ変更してください。

Join Type

"Join Tables" [結合テーブル]ダイアログの下部で結合の種類が決まります。これは、どのテーブルにも存在しないレコードの処理方法を制御します。

OK

[OK]ボタンは、結合演算子と結合タイプに加えた変更を受け入れて、"Join Tables" ダイアログを閉じます。

Cancel

何も変更せず"Join Tables" ダイアログを閉じます。

Delete

現在の **Join** を取り除きダイアログを閉じます。

3-3-3 Remove Join

Join の消去は下記のステップで行います。

1. join を選択
2. マウスの右ボタンをクリック
3. Remove Join を選択

3-3-4 同じテーブルの複数のコピーを使用する

次のように、製品 A、C、または D を注文した顧客を検索するのが一般的です：

HEADER	Name	Product Code
TABLE	CUSTOMER	ORDERS
FIELD	Name	ProductCode
SHOW	Show	Show
SORT	None	None
FILTER 1		Equals 'A' OR 'C' OR 'D'

これは、製品 A、C、または D のいずれかを注文した顧客を返します。

ただし、3つすべての製品を注文した顧客を返し、1つまたは2つのリストされた製品を注文しなかった顧客を除外するクエリが必要です（リストされたもの以外に追加の製品を注文した顧客も受け入れられます）。次のレポートは、ほとんどのデータベースシステムでこの目標を達成しています。

HEADER	Name	Code A	Code C	Code D
TABLE	CUSTOMER	ORDERS	ORDERS_1	ORDERS_2
FIELD	Name	ProductCode	ProductCode	ProductCode
SHOW	Show	Show	Show	Show
SORT	None	None	None	None
FILTER 1		Equals 'A'	Equals 'C'	Equals 'D'

解決策は、ORDERS テーブルの 3 つの「インスタンス」を追加し、それぞれを同じ CUSTOMERS テーブルに結合して、ORDERS テーブルの各インスタンス内で特定のコードを検索することです。3 つの製品すべてを持つ顧客だけがレポートに含まれます。

Windows 3.1 用の Paradox と dBase のソリューション

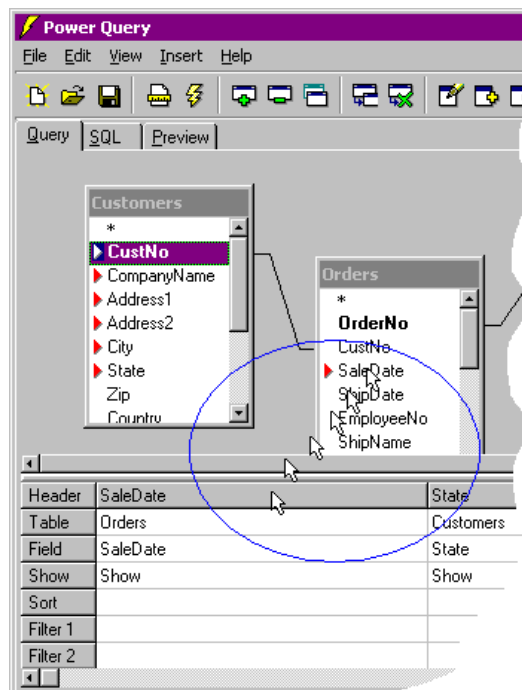
ローカルの Paradox テーブルと dBase テーブル (BDE 2.x の制限) に対して検索するとき上記の手法は Windows 3.1 で機能しません。解決策：上記の複合照会を 3 つの別々の照会に分割し、それぞれが CUSTOMER と ORDERS を 1 回使用して、それぞれ特定の製品コードを検索します。次に結果を ORDER-A.DB、ORDER-C.DB、および ORDER-D.DB にエクスポートします。最後に、上の図と同じ方法で、CUSTOMER を 3 つの一時テーブルすべてに結合する 4 番目のレポートを作成します。結果は同じになります。

3-4 値とフィルタについて

3-4-1 Add Value

クエリに値を追加するのは、簡単な 3 ステップのプロセスです。

1. 選択したフィールド上でマウスの左ボタンを押します
2. クエリグリッドまでマウスをドラッグします。
3. マウスの左ボタンを離します。



マウスの左ボタンを離すと、そのフィールドがクエリグリッドに表示されます。ハードウェアとメモリによってのみ制限されるあなたのクエリに必要なだけ多くの値を持つことができます。

Adding, Removing and Editing Values 値の追加、削除、編集



このボタンを使用して、既存の値の間に新しい空白の値を挿入します。



このボタンを使用して、クエリから既存の値を削除します。



このボタンを使用して、値の設定とフィルタを変更します。

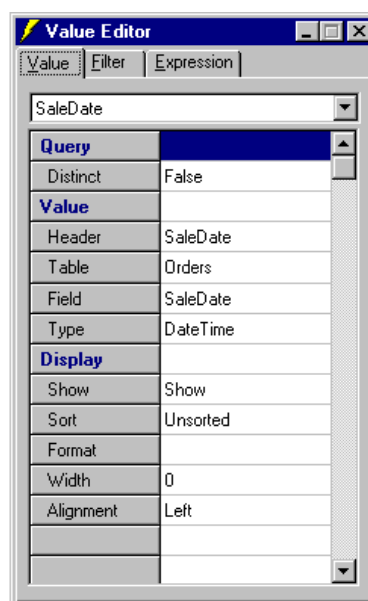
値の並べ替え

値を並べ替えるには、次の手順を実行します。

1. 値ヘッダをポイントする
2. マウスの左ボタンを押したままにします
3. フィールドを左右にドラッグします
4. マウスの左ボタンを放してフィールドをドロップします

3-4-2 Edit Value

フィールドの特性またはプロパティを変更するには、このボタンを押すか、クエリグリッド内のフィールドを直接ダブルクリックするか、"Edit | Edit Field" [編集|フィールドの編集]メニューを選択します。これにより、"Value Editor" [値エディタ]ダイアログが表示されます。



"Value" [値]タブは、次の要素で構成されています。

Header : ヘッダープロパティは、レポートに表示されるフィールドの説明です。空白にすることはできません。デフォルトはデータベース内のフィールドの名前です。一部のフィールド名は、ビジネスレポート上ではわかりにくいか受け入れられないでしょう。**Header** プロパティを使用すると、レポートに表示されるフィールドの説明をカスタマイズできます。スペースや予約語を含むヘッダは **Shazam** によって自動的に管理されます。ヘッダを空白にすることはできません。

Type : **Type** プロパティはフィールドのデータ型です。たとえば、サンプルレポートの **SaleDate** フィールドは有効な **Date** 値のみを受け入れることができます。入力するフィルタ値はデータ型と一致している必要があります。このプロパティは、値がクエリに追加されたときに割り当てられるため、データモデルで値を変更しても、以前に作成されたクエリでは変更されません。

注 : 必要に応じて **Type** プロパティを変更できます。たとえば、データベースエンジンの中には、グラフィックを汎用フォーマット、**BLOB** (Binary Large Object) で格納するものがあります。これもメモの一般的なデータ型です。**Shazam Report Wizard** を使用していて、**BLOB** からグラフィックを表示する必要がある場合は、その **Type** を **Graphic** に変更します。

Show : 表示プロパティは、レポートにフィールドをどのように表示するかを決定します。最も一般的な2つの設定は

Show プレビューレコードにフィールドを含める

Hide は、**Preview Records** および **Preview Report** からフィールドが非表示になる

表示する必要がないフィールドを検索するには、**Hide** [非表示]設定を使用します。

注 : データベースエンジンによっては、隠しフィールドでのソートが許可されていません。

Sort : 並べ替えプロパティは、フィールドを使用して、レコードがレポートに表示される順序を並べ替えるかどうかを決定します。複数の列がソートに選択されている場合、一括ソート順は"**Define Search**" [検索の定義]グリッド内のこれらの列の左から右への順序に基づきます。並べ替えプロパティの値は次のとおりです。

None の場合はソートなし

"**A-Z**" か "**1-9**"の場合は昇順のソート (データ型に依存)

"**Z-A**" か "**9-1**"の場合は降順ソートの場合は (データ型に依存)

Format: **Format** プロパティは、フィールドの表示方法を決定します。詳細については、日付フォーマットマスクと数値フォーマットマスクを参照してください。

Width : 幅プロパティは、フィールドがレポートに表示される期間を決定します。フィールドを非表示にしたい場合は、**Show** プロパティを **Hide** に変更します。

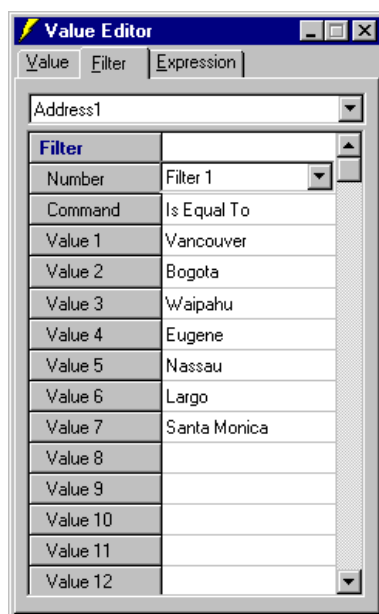
Align : **Align** プロパティは、表示されているフィールドの幅内でフィールド内の値を左、右、または中央に表示するかどうかを決定します。ほとんどの日付データ型と数値データ型は、自動的に右寄せになります。

"Value Editor" [値エディタ]ダイアログから移動する

Value Editor [値エディタ]ダイアログを表示したら、このダイアログを離れることなく値間を移動できます。次の関心分野をクリックしてください。以前に選択したフィールドへの変更は自動的に保存されます。

3-4-3 Edit Filter

"Value Editor" [値エディタ]ダイアログの **filter** [フィルタ]タブには、検索したい1から20の個々のフィルタ値のリストが表示されます。入力できる値の数は、選択されたコマンドによって異なります（下記のコマンドを参照）。



Parameters

次のように角括弧を使ってパラメータを定義します。

[開始日]

[Start Date]は、パラメータのタイトルまたは名前です。次のようにデフォルト値を追加することもできます。

[Start Date] = 1/1/1999

等号に続く値がデフォルト値です。デフォルト値は必要ありません。

パラメータ名は、クエリの複数のフィルタ行と列で繰り返すことができますが、それらは同じ値として扱われます。これは、1つの列に同じ値を含める必要があり、その値が偶然パラメータになるマルチフィルタクエリがある場合に役立ちます。

Power Query が実行されると、"**Enter Values**" [値の入力]ダイアログが表示されます。クエリが実行される前に、すべてのパラメータに有効な値が含まれている必要があります。

注意 : "**Enter Values**" 「値の入力」ダイアログは、ソフトウェアプロバイダによって隠されていると表示されない場合があります。

特別なキーワード

フィルタとして使用できる 2 つの特別なキーワードがあります。

今日 : 今日の日付を示します。 **TODAY** には、 **SaleDate** - **30** などのフィールド名またはカスタム式を使用できます。

BLANK : フィールドに値がないことを示します (データベース用語では **NULL**) 。

コマンド

検索フィルタは、フィルタコマンドと 1 つ以上の検索値で構成されています。有効なフィルタコマンドは次のとおりです。

Is Equal To : 最大 5 つの値のいずれかに等しいレコードを含めます。

Is Between : 2 つの値の範囲内にあるレコードを含めます。

Is Greater Than : 単一の値より大きいレコードを含めます。

Is Greater or Equal To : 単一の値と等しいか以上のレコードを含めます。

Is Less Than : 単一の値より小さいレコードを含めます。

Is Less or Equal To : 単一の値と等しいか以下のレコードを含めます。

Contains : フィールド内に最大 5 つの一致する値を含むレコードを含めます。 **value** で始まるすべてのレコードを検索するには、 "**value%**" を使用してください。値で終わるすべてのレコードを検索するには、 "**%value**" を使用します。 "**%value%**" を使用して、フィールド内の任意の場所に値を含むすべてのレコードを検索します。

Is Not Equal To : 最大 5 つの値に一致しないレコードを除外します。

Is Not Between : 2 つの値の範囲内にあるレコードを除外します。

Does Not Contain : 値を含むレコードを除外します (上記の含むを参照) 。

集計計算を使用したフィルタについて

Show[表示]設定の, Sum [合計], Count[カウント], Average[平均], Minimum [最小]又は Maximum [最大]の設定があるフィールドでは、フィルタ値の扱いが異なります。 例えば：

Maximum:レコードセットの最大値を表示します。

「AmountPaid が 500 を超えた顧客を表示する」は、500 を超える AmountPaid を超える RECORDS を表示します。

「AmountPaid の合計が 500 を超えた顧客を表示する」は、AmountPaid の総計 TOTAL が 500 を超えるレコードを表示します。

最初の例は、レコードの実際の値に対してフィルタ値を適用します。 2 番目の例は、計算がレコードのセットに対して実行された後に値を適用します。 集約関数を使用したレポートでは、必要に応じて他のフィールドやフィルタ値を使用することができますが、それらには Show setting [表示]設定の Group [グループ]または Hide [非表示]が必要です。

3-4-4 Edit Expression

"Value Editor" [値エディタ]ダイアログの"Expression" [式]タブを表示するには、次の 2 つの手順に従います。

クエリグリッドの空白のフィールドにカーソルを置きます。

空白のフィールドをダブルクリックするか、このボタンを押してください。



Using Custom Expressions カスタム式の使用

Tables/Fields テーブル/フィールドおよび **Operators/Functions** 演算子/関数をダブルクリックして、それらをカスタム式フィールドに追加します。カスタムフィールド式を使用するときは、必ず次の規則に従ってください。

カスタム式にフィールド名を追加するときは、手動で入力するのではなく、常にテーブルのアウトラインからフィールドをダブルクリックしてください。フィールドがテーブルのアウトラインから選択されると、**Shazam** は自動的に予約語と句読点をチェックします。

カスタム式では (**Shazam** によってエイリアスされた) テーブル名を使用してください。

句読点やスペースがある、または二重引用符で囲まれた予約語であるサラウンドテーブルおよびフィールドの名前。例: **CUSTOMER "Active"**と **"CUST-95"**。 **"Active"**の両方に引用符が必要です。テーブルのアウトラインから選択すると、それらは自動的に引用されます。

一部のデータベースエンジンは、**Paradox** や **Dbase for Windows 3.1** などのカスタム式に対するフィルタをサポートしていません。回避策については、**Paradox** と **dBase** を参照してください。

これらの規則に忠実に従わないと、生成された **SQL** コードが失敗する可能性があります。レポートの実行後に **TOKEN** エラーが発生した場合、問題はカスタム式にテーブル名がないか、または二重引用符なしでは正しく処理できない名前である可能性があります。

トークンエラーの解決

フィールドがテーブルのアウトラインから選択されると、**Shazam** は自動的に 270 以上の予約語、スペース、句読点をチェックします。ただし、テーブルのアウトラインから選択したフィールド名で **TOKEN** エラーが発生した場合は、カスタムフィールド式を使用して問題を回避できます。これは 2 段階のプロセスです。

フィールド名と一致する場合は、ヘッダに引用符を追加してください。

カスタムフィールド式からフィールドを選択して引用符を追加します。

注: 既存のデータベースフィールドが既に選択されていて、タブを切り替えて **Custom Expression** カスタム式を作成した場合は、**[OK]**を押すと元のフィールドがクエリグリッドで置き換えられます。これにより、**"Values Editor"** [値エディタ]ダイアログを終了せずに特定のフィールドをカスタム式に変換できます。