

Access 2000 ホントのところ

販売管理システムを作ろう

第1回 システム開発のスタート

文/FunTech Naoki Koga

この連載では、Access 2000を利用して販売管理システムを構築しながら、データベース構築の基礎的な事柄を説明します。とはいっても、数多く発売されている一般書籍のように正面から懇切丁寧に説明をするのではなく、できるだけ筆者の主観を交えて、Accessの現実的な利用方法を中心に話を進めてみたいと考えています。データベース入門書にギブアップしてしまった方や、これからAccessを使ってみようと思っていた方を対象にしたわかりやすい解説を目指しますので、データベースの初心者の方もAccess 2000を操作しながら挑戦してみてください。

この連載の目標

この連載では、販売管理システムという非常に一般的な業務システムを例に取り上げることになります。なぜ、販売管理システムにしたのかということについては、後ほど説明します。とりあえず全5回を目標にして、それなりに利用可能な販売管理システムを構築していきます。そして、ここが一番のポイントですが、この連載ではタイトル通り「Access 2000を本当の意味で活用する」ことを目標にして、一般的なシステムにとって不必要であったり、使わなくても問題のない機能についての説明は、パッサリと省略することになります。つまり、データベース入門書にありがちな「過度なデータベースの概要」、「必須ではない機能」、「初めのうちは不要な機能」などについての説明を省略もしくは後回しにして、システムを完成させるために必要な機能と知識を確実に習得できるようにしていきたいと思えます。

データベースとは？

それでは早速データベースシステムの開発を開始しましょう、といきたいところですが、やはり残念ながら、その前に最低限理解しておかなければならない事柄があります。

まず、データベースとは何でしょうか。本来、データベースとはコンピュータ上で管理する情報の集まりのことを指します。もちろん、一般にはこれらの情報を管理するソフトウェアもデータベースと呼びますが、このソフトウェアは正式にはデータベース管理システム(Database Management System, 以下DBMS)と言います。DBMSの代表的なものとしては、Oracle, SQL Server, Informix, DB2などが挙げられます。なお、通常DBMSにはデータベースの制御を行う「データベースエンジン」と、データベースエンジンをユーザーが操作するための「管理ツール」が含まれます。つまり、「データベース

を使う」という場合には、「DBMSを利用すること」と「データベースシステムを操作すること」という2つの意味があることとなります。

Access 2000とは？

それでは、Accessというソフトウェアは何でしょうか。「そんなのDBMSに決まっているじゃないか」という声が聞こえてきそうですが、実はAccessはデータベースエンジン+管理ツール+システム開発環境です。つまり、純粋なDBMSではなく、DBMSにシステム開発環境を追加したものです。図1に示すように、Accessはデータベースエンジンとして、Jetデータベースエンジンを利用してデータを管理しています。なおAccess 2000からは、データベースエンジンとしてSQL Serverデータベースを利用することもできるようになりましたが、この連載ではJetデータベースエンジンのみを使用することにします。

システム開発環境としては、フォームやレポートといったAccess独自のオブジェクトと共に、データベース自体とそれらのオブジェクトを操作するためのマクロとモジュールが搭載されています。このモジュールとは、Officeアプリケーションで共通のプログラム言語であるVisual Basic for Applications（以下VBA）のことです。

つまり、AccessはDBMSであると同時に、データベースシステムを開発するための機能と開発言語が搭載された統合開発環境であるというわけです。

Access 2000の機能概要

次に、Accessに用意されている機能を簡単に説明します。なお、ここで紹介している画面は、以前筆者の会社で開発した「学生募集データ管理システム（販売：社団法人日本能率協会）」を使用しています。

Access 2000で作成できるオブジェクトとしては、テーブル、クエリ、フォーム、レポート、データアクセスページ（単にページとも表記します）、マクロ、モジュールがあります。これらのオブジェクトは、データベースウィンドウを使用して、オブジェクトごとに管理されます（図2）。

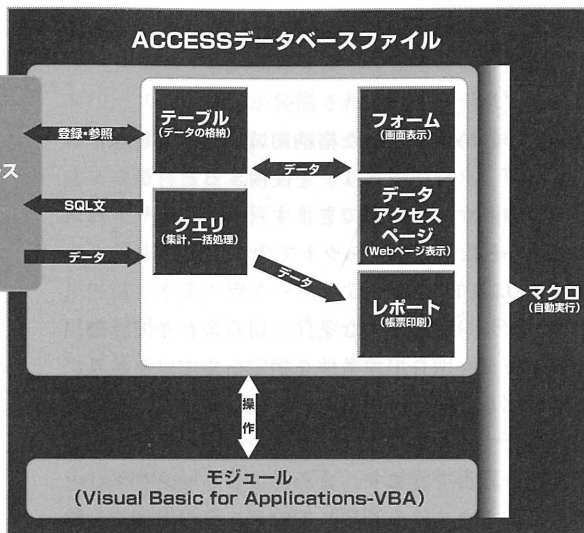


図1 Access 2000の構成



図2 Access 2000の作業画面

図2を見るとわかるように、上記の7種類のオブジェクトがそれぞれのグループごとに格納されています。たとえば、図2はフォームのオブジェクトグループを表示している状態です。その下の「グループ」という分類は、それぞれのオブジェクトを管理するためのショートカット用フォルダを提供しているものです。

ここで、再び図1を参照すると、テーブルとクエリの2つのオブジェクトだけが直接Jetデータベースエンジンと連携していることに気づくはずですが、Jetデータベースエンジンによって、テーブルに登録されたデータが格納され、クエリでそのデータを操作する指示を行います。

Accessの7種類のオブジェクトについて把握しておく必要のある事柄を、以下に整理しておきます。

●テーブル

あらかじめ定義された格納領域にデータを保管します。データシートビューを使用すると、データを登録し参照することができます(図3)。データベースの中心となるオブジェクトです。

●クエリ

テーブルをさまざまな条件で組み合わせて、抽出するデータを取り出す条件を指定します。クエリにはテーブルと違いデータそのものは格納されませんが、実行するたびにテーブルの最新データを利用して情報を取得できます。また、アクションクエリという特殊なクエリを利用して、データの追加や削除等の一括処理を実行することもできます。Accessの内部ではクエリをSQL文に変換してJetデータベースエンジンに渡しますが、ユーザーは通常変換後の内容を知る必要はありません(図4)。

●フォーム

テーブルやクエリを基にして、データベースに格

納されたデータを画面に表示します。フォームを利用すると、データ参照や入力だけでなく、データの検索や並べ替え、フィルタなどの操作を行うことができます。このフォームの機能が、システムの使いやすさを大きく左右します。また、フォーム上でさまざまな動作を制御するために、モジュールを利用できます(図5、6)。

●レポート

テーブルやクエリのデータを印刷するために使用します。出力される内容は、印刷プレビューを利用して画面上で確認することもできます。レポートはフォームとは異なり、印刷専用であるため、複数レベルでの集計を行うことができます。たとえば、小計と合計を出力することが可能になります。また、ラベルや宅配便の伝票などを印刷することも可能です(図7)。

●データアクセスページ

Webページとしてテーブルやクエリのデータを参照したり、データを登録することができます。Webページを表示するためには、Internet Explorer 5.0以

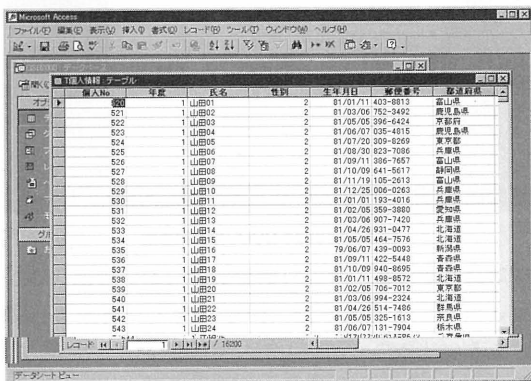


図3 テーブルの例

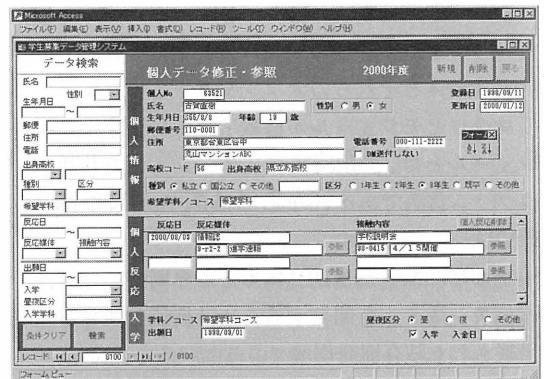


図5 フォームの例

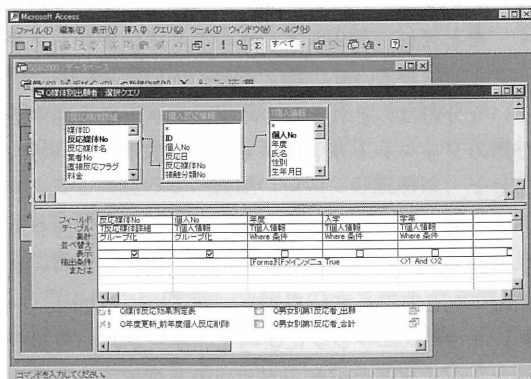


図4 クエリの例

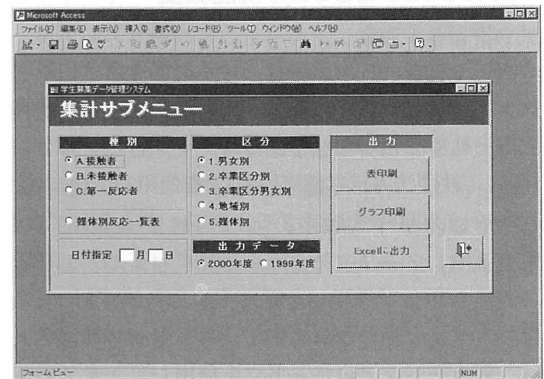


図6 フォームの例(メニュー)

降を使用します。ただし、この機能はAccess 2000で初めて装備されたもので、残念ながらまだ洗練されておらず実用的なものではありません(図8)。

●マクロ

Accessの機能を自動的に実行できます。ただし、Access 2000では、ほとんどの機能をモジュールで利用することができるため、モジュールを利用できるユーザーにとっては不要です。しかし、作成方法が単純なため初心者にも理解しやすいという利点があり、モジュールの習得が難しいユーザーであればマクロを利用する選択肢も考えられます(図9)。

●モジュール

VBA言語で記述したプログラムにより、データベースやAccessのオブジェクトを自在に操作することができます。ある程度の機能を持つシステムの構築を考えているのであれば、VBAの習得は必須と言えます。ただし、VBAは簡易言語ではなく、通常のプログラミング言語なので、利用するためにはそれなりの学習と経験が必要になります(図10)。

ここで、Access 2000をこれから使い始めようと考

えている方に、少しだけヒントをお教えしておきます。まず、データベースウィザードという自動データベース作成機能が装備されていますが、これは実務にはほとんど役に立ちません。Accessの機能の参考になる程度だと考えてください。また、機能全般にウィザードが装備されていますが、ウィザードでのみ操作可能なもの(テキストのインポートなど)以外は、本来の機能を覚えてから補助的にウィザードを利用するようにしてください。特にテーブルやクエリをウィザードでしか作成できないと、はっきり言ってAccessを利用する意味も価値もありません。Office Professional以上のパッケージでは、ExcelやWordなどと組み合わせて販売されていますが、Accessはそれらのアプリケーションとはまったく次元の異なるアプリケーションです。しかし、ある程度の学習は必要となりますが、他のアプリケーションとはレベルの違う利用効果を得られることは保証できます。

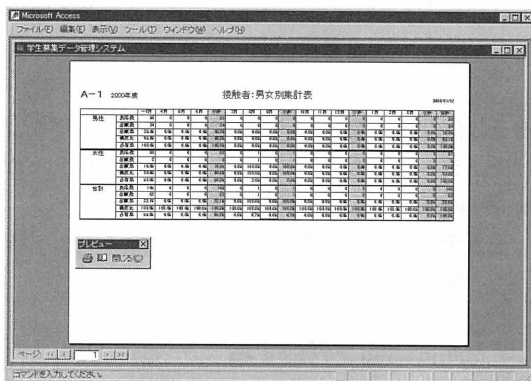


図7 レポートの例

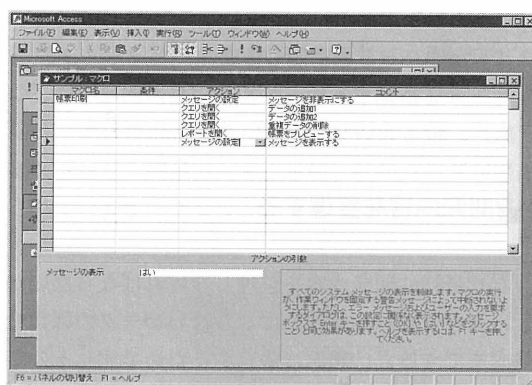


図9 マクロの例

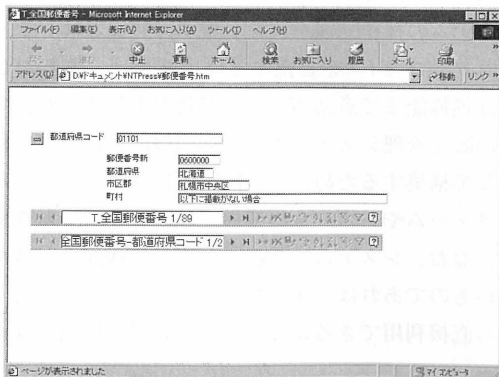


図8 データアクセスページの例

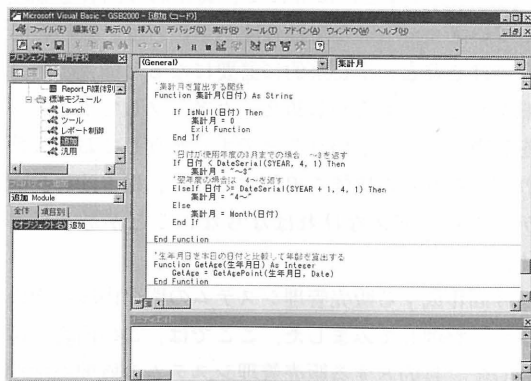


図10 モジュールの例

システム開発を始める前に

それでは、Accessを利用したシステム開発の手順を説明します。Excelの場合には、ワークシートをとりあえず作り始めてから、不備な部分を後で修正するというやり方が通用しましたが、残念ながらAccessではあまりこのやり方は得策ではありません。Accessだけに限りませんが、データベースシステムを開発する場合には、事前に決定しておくべき事柄がExcelを利用する場合に比べると格段に多くなります。本当は何度も失敗して身をもって体験することをお勧めしたいところですが、本業以外の方には酷なことでしょうから、ここで事前にまとめておくべきことを整理しておくことにします。

1. 開発内容の整理

「どのようなデータを登録するか」、「どのような画面が必要か」、「どのような帳票が必要か」、「どのような機能が必要か」ということをまとめておきます。

2. 画面遷移図の作成

画面遷移図とは、システムに装備される機能がメニューから実行される順番を表すチャート図です。

3. データベース構造図の作成

データを格納するテーブルの相互の関連を図示します。最も大切なものといえます。

開発内容を整理する

なぜ最初に開発内容を整理しなければならないかということ、登録しなければならないデータの種類や構成をできるだけ正確に決定するためです。そのために、データ、フォーム、レポート、その他の機能（データ生成など）に分けて、具体的な内容を整理していきます。ここで注意すべきことは、最初の時点では必要ないが将来的に必要になる可能性のある帳票なども、あらかじめピックアップしてポイントを整理しておくということです。そうしておかなければ、それらの機能を追加する際に、根本的にシステムを作り変えなければならないことがあり得ます。

今回作成する販売管理システムの開発内容を次のように整理してみました。ここでは、この内容を確認して、目的とする販売管理システムの概要を把握しておいてください。

●データ

顧客データ 顧客名、郵便番号、住所1、住所2、電話番号、担当者名

商品データ 商品名、単価

売上データ 売上日、顧客名、商品名、個数、消費税率、販売担当者名

担当者データ 社員コード、担当者名

●フォーム

顧客登録フォーム

商品登録フォーム

担当者登録フォーム

売上登録フォーム

●レポート

顧客リスト ただし、過去1年間に売上のない顧客はリストアップしない

月間売上一覧表 指定月の売上明細を日付順で一覧にする

商品顧客別分析表 商品と顧客ごとに売上金額の合計を集計する

顧客ラベル DM発送用に全顧客のラベルを作成する

●その他の機能

売上データ出力 Excelで詳細の分析を行うためのデータを出力する

過去データ一括削除 一年以上経過したデータを削除する

画面遷移図を作成する

次に大まかなユーザーインターフェイス、つまりシステムの操作方法を決定します。特殊なユーザーインターフェイスを装備するのであれば、あらかじめ詳細機能まで煮詰めておく必要がありますが、今回の販売管理システムはAccessの機能を標準的に利用して構築するため、この段階ではメニューからの各フォームやレポートの遷移だけを整理しておきます。なお、システムがほんのわずかな機能しか持たないものであれば、すべての機能がメインメニューから直接利用できるようになっていても問題ありません。

しかし、多くの機能が存在していたり（目安とし

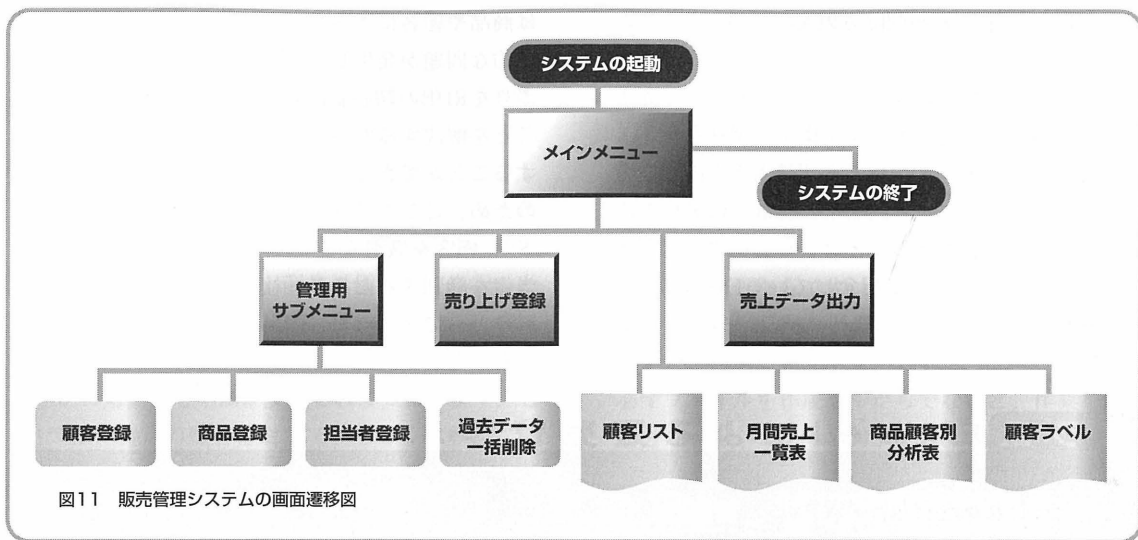


図11 販売管理システムの画面遷移図

で8つ以上), 通常は利用しない機能をメインメニューには配置しておきたくない場合には, サブメニューの利用を検討します. 特にデータの一括削除など, 操作ミスが取り返しのつかない結果につながる可能性のある機能については, できるだけ通常表示するメインメニューには配置しない方がよいでしょう.

以上のことを検討した結果, 販売管理システムの画面遷移図は図11のようになりました.

データベース構造図を作成する

まずは, できあがった販売管理システムのデータ

ベース構造図を参照してください (図12). 図をよく見ると, 先ほど開発内容の整理でピックアップしたデータとは異なる点があることに気付くはずで, ここではあまり詳しくは説明しませんが, 実はまさしくこの違いにAccessなどのリレーショナルデータベース (以下RDB) の難しさが含まれています.

それぞれのデータ (テーブル) にはいくつかの項目 (フィールド) が追加されていたり, 削除されています. たとえば, 顧客データには顧客コード, 商品データには商品コード, 売上データには売上ID, 顧客コード, 商品コード, 単価, 社員コードが追加されています. そして, 売上データから顧客名, 商

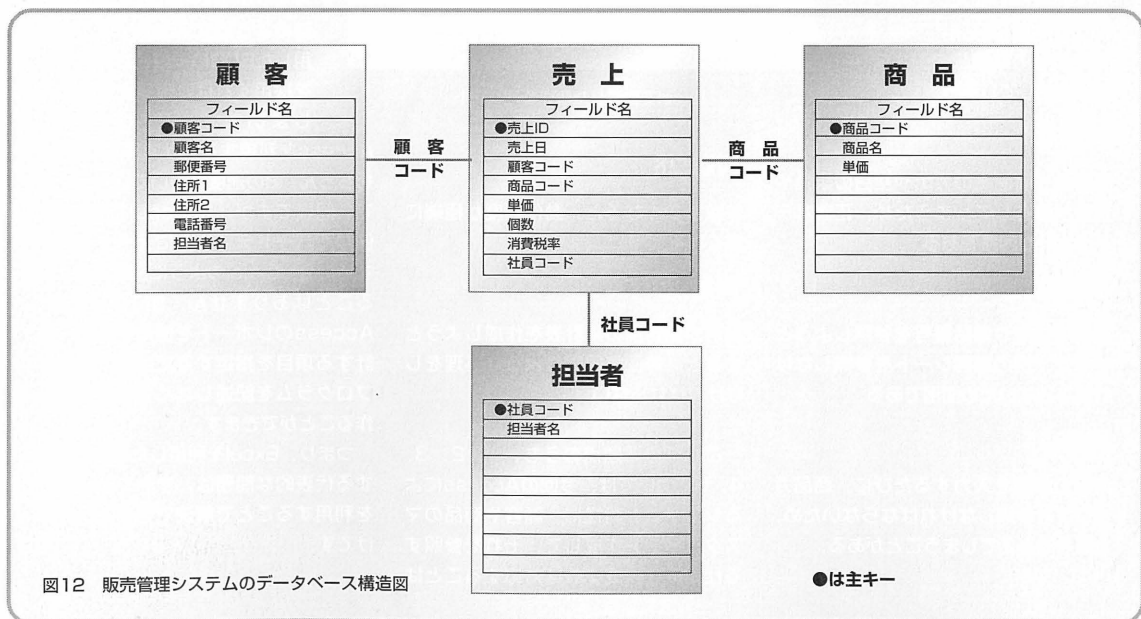


図12 販売管理システムのデータベース構造図

●は主キー

品名、販売担当者名が削除されています。これは、なぜでしょうか。

ここで、この販売管理システムで中心となる売上データを考えてみましょう。手作業で業務を行うことを考えると、データ（たとえば請求書の内容を考えてみてください）には商品や顧客はその名称のみが必要です。これをそのままコンピュータ上に置き換えたものが、コラムで紹介しているExcelによる販売管理シートです。しかし、このようなシートで

は商品や顧客によってデータを集計する場合に、根本的な問題が発生してしまいます。実はこれと同じ誤りをRDBの初心者は犯してしまうのです。以上のことを解決するためには、商品や顧客を個別に識別することができるようにしなければなりません。そのため、ここでは商品を識別するための「商品コード」、顧客を識別するための「顧客コード」、販売担当者を識別するための「社員コード」を追加したというわけです。なお、商品データや顧客データのよ

コラム Excelによる販売管理

ここで、Excelで販売管理シートを作成した場合を考えてみましょう。難しいことを考えずにワークシートを作成すれば、図のようになるのではないのでしょうか。実際に業務でこのようなシートを活用している方もいます。この販売管理シートで業務を行った場合、次のような問題が発生することがあります。

問題点1

有限会社が株式会社になったり、個人のお客様が結婚して姓が変わることなどの理由により顧客名が変わってしまい、その結果過去のデータの集計がうまくいかなくなる。

問題点2

同姓同名のお客様の売上データが混在してしまう。

問題点3

顧客名や商品名のカナや空白の半角や全角が混在してしまうため、正確なデータ集計が困難である。

問題点4

顧客の住所や電話番号を参照しようとすると、別の顧客シートを参照しなければならないため面倒である。

問題点5

商品の単価を入力するたびに、商品カタログを参照しなければならないため、金額を間違えてしまうことがある。

問題点6

他の販売担当者がワークシートを開いていると、同時に修正することができなくなってしまう。

問題点7

1枚のシートに最大65,535件までしか登録できない。

日付	顧客名	商品名	数量	単価	総金額	税率	消費税	税込金額	振込金額
2000/2/10	ABC商事	ボストンチップス	1500	200	300000	5%	15000	315000	315000
2000/2/10	2次アクリル樹脂	ペンチ	80	45	36000	5%	1800	37800	37800
2000/2/10	2次アクリル樹脂	ボルト	100	100	10000	5%	500	10500	10500
2000/2/11	ぽんぽんや	チョコレート	2000	200	400000	5%	20000	420000	420000
2000/2/11	ぽんぽんや	ボストンチップス	1500	1500	2250000	5%	112500	2362500	2362500
2000/2/11	山田商店	ボルト	1000	5	5000	5%	250	5250	5250
2000/2/13	山田商店	ボストンチップス	1500	5	7500	5%	375	7875	7875
2000/2/13	ABC商事	チョコレート	1500	25	37500	5%	1875	39375	39375
2000/2/14	2次アクリル樹脂	チョコレート	2000	200	400000	5%	20000	420000	420000
2000/2/14	2次アクリル樹脂	チョコレート	80	10	800	5%	40	840	840
2000/2/14	ABC商事	チョコレート	2000	10	20000	5%	1000	21000	21000
2000/2/14	ぽんぽんや	ボルト	1000	200	200000	5%	10000	210000	210000
2000/2/14	ABC商事	ボルト	1000	5	5000	5%	250	5250	5250
2000/2/14	ABC商事	ボルト	1000	5	5000	5%	250	5250	5250

Excelで作成した販売管理シート

可能です。ただし、VLOOKUP関数などを利用した数式では、件数が多いときにとてつもない計算時間がかかるようになってしまいます。もしくは、VBAでプログラムを記述するしかありません。

6, 7, 8についてはExcelの宿命で、現時点ではどうしようもありません。これらの問題が発生しないような状況

でのみ利用するしかありません。9については、VBAを利用したプログラムを作成することで対応可能ですが、シート上のデータのセル位置が毎回変動するため、思った以上に大変なプログラムになることが多いはず。

Accessを利用した場合、これらの問題はどのようになるでしょうか。1~5については、今回のようにマスタテーブルを使用したRDBの一般的な開発手法を採用すれば、自然と解決して

しまいます。6, 7, 8は、Accessを利用するだけで解決します。もちろん、8の速度についてはAccessでも同様ですが、Accessは件数に関係なくハードディスクにデータを格納するため、Excelのように極端に遅くなることはありません。9については、Accessのレポートを使用すると、集計する項目を指定するだけで、特別なプログラムを記述しなくても集計表を作ることができます。

つまり、Excelを利用した場合に発生する代表的な問題は、すべてAccessを利用することで解決できるというわけです。

問題点8

件数が多くなると、動作速度が極端に遅くなる。

問題点9

月別や顧客別の集計表を作成しようとすると、毎回毎回同じような処理をしなければならない。

これらの問題点のうち、1, 2, 3, 4, 5については、今回のAccessによるシステムと同様に、顧客や商品のマスタシートを作成して、それを参照する仕組みを作成すれば対応することは

うに、常に追加されるデータから参照するデータを「マスタ」と呼びます。

次に、売上データに単価を追加した理由を説明しましょう。最初の考えでは、商品マスタに単価を用意しておいて、その数値を利用する予定でした。しかし、このような商品マスタに格納されている単価をシステム全体に利用すると、大抵の場合2つの問題が発生します。1つは、商品単価の改定を行った場合です。商品マスタに単価を格納した場合には、単価を変更したときに、蓄積されている過去の売上データに対しても単価が変更されてしまいます。もう1つは、特定の顧客に対して、もしくは数量に応じて値引きを行った場合です。これらの問題を解決する最も単純な方法として、商品マスタの単価をデータ登録時の参照用のみ利用することにして、実際の売上データの中に単価を格納するようになっています。

なぜ、販売管理システムなのか？

さて、ここで今回の連載の課題として、販売管理システムを採用した理由を明らかにしておきましょう。まず1つめは、誰にでもその内容が把握しやすいということです。もう1つの理由は、販売管理システムが「多対多」のリレーションシップを持っているためです。リレーションシップとは、テーブル間の結合のことを指します、つまり、商品マスタと売上データ、顧客マスタと売上データの間にリレーションシップが存在します（なお、ここで説明しているリレーションシップとは論理的なものを指しており、Accessの機能としてのリレーションシップではありません）。この2つのリレーションシップは、それぞれ「一対多」のリレーションシップと言えます。つまり、ある商品（一側）が複数の売上データ（多側）に登録されることがあるということです。また、ある顧客（一側）が複数の売上データ（多側）に登録されることがあります。そして、顧客と商品から見ると、互いに複数の関連データを持つ可能性があるため、「多対多」の関係になっているというわけです。この最も複雑な形態である「多対多」の関係を持つテーブル構造を把握できれば、他の複雑なシステムでも、大抵の場合はこれを応用することで対応できます。とりあえず現時点では、リレー

ションシップについては、これ以上言及しないことにしておきましょう。

データベースファイルを作成する

それでは前準備が終わったので、いよいよAccessの操作に取り掛かります。まずは、データベースファイルの作成からです。Accessが利用するファイルは、ExcelやWordなどのOfficeアプリケーションといくつかの点で違いがあります。一番大きな違いは、必ずファイルを作成してからでないと、作業を開始できないということです。Excelなどのアプリケーションでは、新規ドキュメントを作成して、最終的に必要であればファイルを保存するということが可能ですが、Accessではデータベースを利用するためにファイルに作成される管理用のシステムオブジェクトが必要になるため、あらかじめファイルを保存しておかなければなりません。

[スタート] - [プログラム] - [Microsoft Access] をクリックして、Access 2000を起動します。スタートアップダイアログボックスで、[空のデータベース] を選択して、[OK] をクリックします(図13)。次に [データベースの新規作成] ダイアロ



図13 Access 2000のスタートアップダイアログボックス

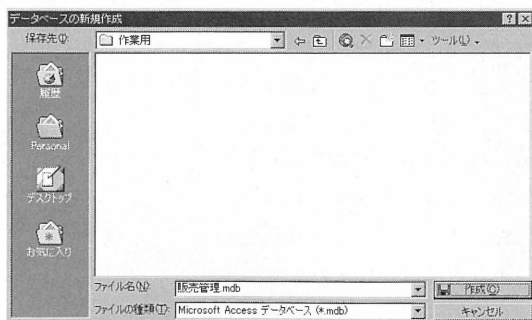


図14 [データベースの新規作成] ダイアログボックス

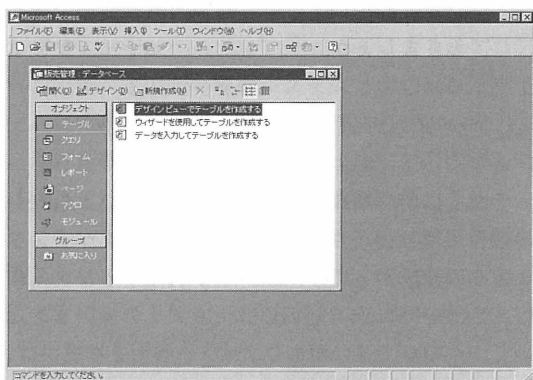


図15 データベース作成直後のデータベースウィンドウ

グボックスでファイル名を指定して、[作成] ボタンをクリックします（前頁図14）。ここでは、とりあえず適当なフォルダに、「販売管理」という名前でデータベースファイルを作成します。これで、新規にデータベースファイル（拡張子.mdb）が作成され、空のデータベースウィンドウが表示されます（図15）。

このようにして作成されたAccessデータベースには、P177図1に示したテーブル、クエリ、フォーム、レポート、データアクセスページ、マクロ、モジュールといったオブジェクトと各種設定情報が格納されます。つまり、テーブルなどのAccessオブジェクトを作成すると、それらのオブジェクトがデータベースウィンドウに格納され、mdbファイルに保存されます。なお、データアクセスページだけは、通常のファイルとして保存されるhtmlファイルにリンクするショートカットのみが格納され、オブジェクト自体は含まれません。

■ テーブルを作成する

それでは、いよいよテーブルの作成に入ります。既に説明したように、このテーブルがデータベースの中心であり、最も注意深く作成しておかなければならないものです。基本的には、システム開発を始めた後には修正することができないぐらいの気持ちで作成してください。

ひとことでテーブル作成といっても、実際には次のような処理が含まれます。

- フィールド名の定義
- データ型の指定
- フィールドプロパティの設定
- 主キーの指定
- インデックスの指定

「フィールド」とは、テーブルに含まれるそれぞれの項目のことです。たとえば、商品マスタの場合には、「商品コード」、「商品名」、「単価」などがフィールドになります。これらを定義する際に、フィールド名には記号や空白、半角カナを使用しないように注意してください。そして、それぞれのフィールドごとに、値を格納する枠として「データ型」を決定します。このデータ型をあらかじめ定義すること、つまりデータの格納領域と格納方法を定義することは、データベースが安定して高速に動作するために必要なことです。この点も、自由なセルにデータを登録できるExcelなどの表計算ソフトとの本質的な違いと言えます。

「フィールドプロパティ」とは、それぞれのフィールドに定義できる属性で、[書式] プロパティや[既定値] プロパティが存在します。利用した方がよいプロパティや便利なプロパティもたくさんありますが、当面はデータのサイズを定義する[フィールドサイズ] プロパティ以外は使用しません。

「主キー」は、既に説明した「商品コード」や「顧客コード」などのように、データを個別に識別するフィールドのことです。「インデックス」は、検索や並べ替え、フィルタといったデータ操作を高速にするためのものです。ここでは、このインデックスも後回しにしておきます。

■ データ型を指定する

Access 2000では、9種類のデータ型を利用できます。以下に、データ型について簡単にまとめておきます。

● テキスト型

指定した文字数までのテキストを登録できます。最大255文字という制限があります。

● メモ型

65,535文字までのテキストを登録できます。

●数値型

数値を登録できます。

●日付/時刻型

日付や時刻を登録できます。

●通貨型

小数点以下4桁までの精度を持つ数値を登録できます。

●オートナンバー型

自動的にナンバリングされる数値が登録されます。

●Yes/No型

「はい」と「いいえ」のいずれかであることを表します。

●OLEオブジェクト型

写真やExcelワークシートなど、他のアプリケーションのデータを格納することができます。

●ハイパーリンク型

イントラネットやインターネットのアドレスを格納します。

このうち、日付/時刻型、OLEオブジェクト型、ハイパーリンク型については、すぐに理解できるはずですが、テキスト型とメモ型は、登録するデータの最大文字数で使い分けます。メモ型は長い文章を登録できますが、その代わりに動作速度が遅く、並べ替えができないなどの制限があります。そのため、

通常のフィールドにはテキスト型を利用して、備欄などにのみメモ型を定義します。

Accessに限らず、データベースで最も判断が困難なデータ型は数値に関するものです。Accessでは、数値を格納できるデータ型として、数値型、オートナンバー型、通貨型の3つが用意されています。このうちオートナンバー型については、基本的には主キーとして自動ナンバリングの数値を使用する場合にのみ定義します。つまり、伝票にナンバリングマシンを使用して、毎回異なる数値を割り当てているような状況で利用することになります。通貨型は通貨だけでなく、正確な小数値の演算が必要なケースで利用できます。ただし、小数値は4桁までしか格

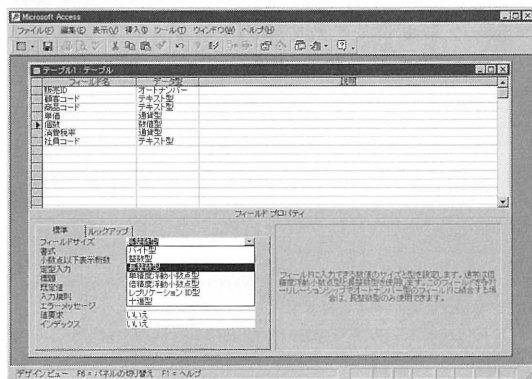


図16 数値型の7種類のサブタイプ

●表1

「顧客マスタ」テーブル

フィールド名	データ型	フィールドサイズ	主キー
顧客コード	テキスト型	7	●
顧客名	テキスト型	50	
郵便番号	テキスト型	8	
住所1	テキスト型	50	
住所2	テキスト型	50	
電話番号	テキスト型	20	
担当者名	テキスト型	20	

「商品マスタ」テーブル

フィールド名	データ型	フィールドサイズ	主キー
商品コード	テキスト型	4	●
商品名	テキスト型	30	
単価	通貨型		

「担当者マスタ」テーブル

フィールド名	データ型	フィールドサイズ	主キー
社員コード	テキスト型	5	●
担当者名	テキスト型	20	

「売上データ」テーブル

フィールド名	データ型	フィールドサイズ	主キー
売上ID	オートナンバー型		●
売上日	日付/時刻型		
顧客コード	テキスト型	7	
商品コード	テキスト型	4	
単価	通貨型		
個数	数値型	長整数型	
消費税率	通貨型		
社員コード	テキスト型	5	

納できないので注意してください。これら以外の場合には、すべて数値型を利用することになりますが、ややこしいことに数値型はさらに7種類のサブタイプを持っています。これらのサブタイプは、前頁図16のように「フィールドサイズ」プロパティのコンボボックスで設定できます。それぞれに利用できる用途や精度、必要な領域等が異なりますが、ここではまず「整数値のみの場合には長整数型」、「小数値が必要な場合には倍精度浮動小数点型」としておけばほぼ問題はありません。

以上のことを踏まえて、前頁表1のようにテーブルの定義を行います。作成方法は、データベースウィンドウでテーブルの一覧が表示された状態で、「デザインビューでテーブルを作成する」をダブルクリックします。これで空のテーブルウィンドウが表示されるので、先頭行にフィールド名を入力し、データ型を選択して、ウィンドウ下部の「フィールドサイズ」プロパティにフィールドサイズを入力または選択します。すべてのフィールドを定義したら、主キーのフィールドの行にカーソルを合わせて、ツ

ールバーの「主キー」ボタンをクリックします。主キーとして設定されたフィールドは、左側のフィールドセレクタに鍵のアイコンが表示されます。たとえば、完成した「顧客マスタ」テーブルは図17のようになります。最後に、ツールバーの「上書き保存」ボタンをクリックして、テーブルの定義を保存します。4つのテーブルを作成すると、データベースウィンドウは図18のようになります。これで、システム開発の大きな第一歩を踏み出したというわけです。なお、ここまでのデータベースファイルは、Windows 2000 PRESSのホームページからダウンロードすることができます（注）。

注 ダウンロード先
<http://www.gihyo.co.jp/ntpress/accessdb.htm>

■ ④ テーブルの内容を表示する

これである意味ではデータベースが完成したともいえるわけですが、通常の場合はこの後でフォームやレポートを使用したユーザーインタフェースを作成します。しかし、フォームやレポートを使用しなくても、テーブルにデータを登録したり、テーブルに格納されているデータを参照することは可能です。

データベースウィンドウで対象となるテーブルをダブルクリックすると、選択したテーブルがデータシートビューで表示されます。図19では、「顧客マスタ」テーブルを表示しています。この図では、まだ1件もデータが登録されていない状態なので、空白行が1行だけ表示されています。この空白のセル

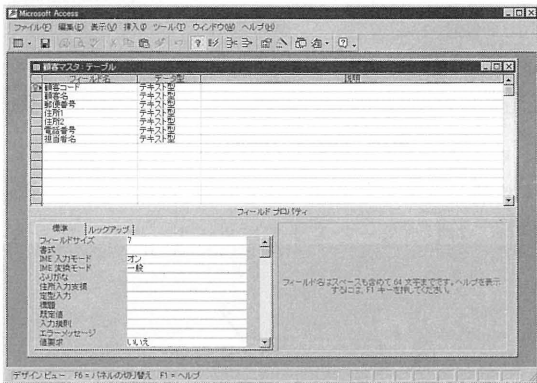


図17 「顧客マスタ」テーブル

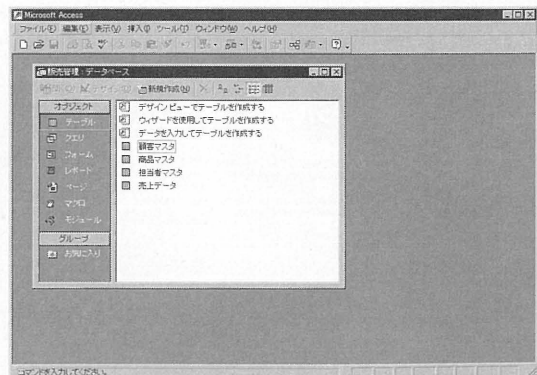


図18 4つのテーブルを含むデータベースウィンドウ

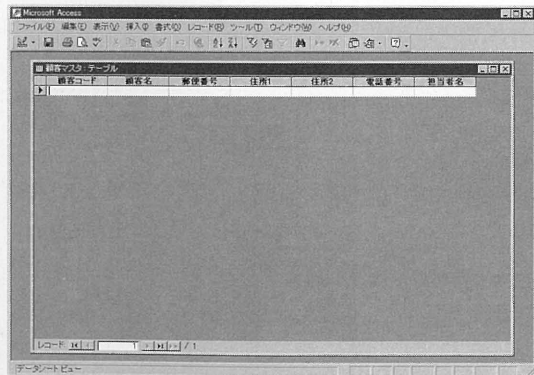


図19 データシートビューで表示した「顧客マスタ」テーブル

に、新規データを登録することができます。登録されたデータはそのままデータシートに表示され、常に空白行が最下行に配置されるようになります。何件かのデータを登録すると、図20のようになります。

このデータシートビューでは、以下のようにExcelワークシートと同等の機能が装備されています。なお、フィルタについての詳細は、フォームの作成のときに説明することになります。

- **追加** 最下行の空白にデータを追加できる。
- **修正** データシート上のデータを書き換えることができる。
- **削除** 削除する行にカーソルを合わせて、[編集] - [レコードの削除] をクリックして、確認メッセージで [はい] をクリックすると、選択したデータを削除できる。
- **検索** ツールバーの [検索] ボタンをクリックすると、登録されているデータを検索できる。
- **置換** [ファイル] - [置換] をクリックすると、検索したデータを置換することができる。
- **フィルタ** ツールバーの [選択フィルタ] ボタンや [フォームフィルタ] ボタンをクリックすると、フィルタ機能 (絞り込み機能) を利用することができる。

- **並べ替え** ツールバーの [昇順で並べ替え] ボタンと [降順で並べ替え] ボタンのいずれかをクリックすると、現在カーソルのあるフィールドの値で並べ替えを実行できる。

さて、今回のシステム開発はここまでです。なかなか進みが遅いような気がするかもしれませんが、残念ながら実際にシステムを開発する場合でも、最初のうちはこんな感じです。次回は、フォームの開発について説明することになります。

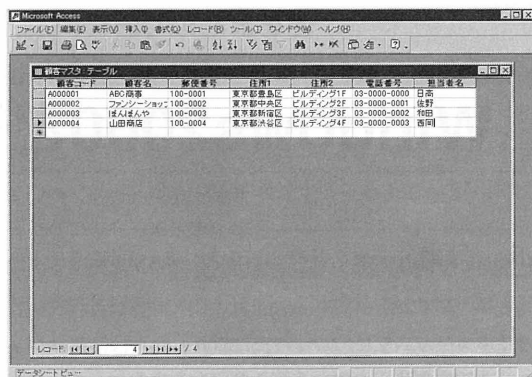


図20 数件のデータを登録した「顧客マスタ」テーブル

ここまでのできあがり

顧客コード	顧客名	郵便番号	住所1	住所2	電話番号	担当者名
A000001	ABC商事	100-0001	東京都豊島区	ビルディング1F	03-0000-0000	日高
A000002	ファンシーショップ田中	100-0002	東京都中央区	ビルディング2F	03-0000-0001	佐野
A000003	まんまんや	100-0003	東京都新宿区	ビルディング3F	03-0000-0002	新田
A000004	山田商店	100-0004	東京都渋谷区	ビルディング4F	03-0000-0003	徳川

商品コード	商品名	単価
0001	ガム	¥100
0002	キャンディー	¥80
0003	チョコレート	¥200
0004	ポテトチップス	¥150

社員コード	担当者名
S0001	宮元 肇
S0002	古賀 清美
S0003	山本 奈那子
S0004	竹之内 徹

売上ID	売上日	顧客コード	商品コード	単価	個数	消費税率	社員コード
1	00/02/10	A000001	0004	¥150	20	¥5	S0002
2	00/02/10	A000002	0002	¥80	25	¥5	S0003
3	00/02/10	A000002	0001	¥100	10	¥5	S0003
4	00/02/11	A000003	0004	¥150	10	¥5	S0001
5	00/02/11	A000003	0003	¥200	20	¥5	S0001
6	00/02/12	A000004	0001	¥100	5	¥5	S0004
7	00/02/13	A000001	0004	¥150	25	¥5	S0004
8	00/02/13	A000004	0004	¥150	5	¥5	S0002
9	00/02/14	A000001	0003	¥200	10	¥5	S0003
10	00/02/14	A000002	0003	¥200	20	¥5	S0003
11	00/02/14	A000002	0002	¥80	10	¥5	S0002
12	00/02/14	A000001	0001	¥100	5	¥5	S0001
13	00/02/14	A000003	0001	¥100	20	¥5	S0002