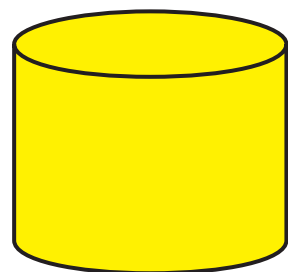


校務に役立つ データベース入門

近江兄弟社高等学校 長谷川友彦



普段の校務の中で、生徒の様々な情報を管理したり、それをもとに書類を作ったりと、複雑で難しい仕事が山のようにありますよね。その度にとっても時間のかかる作業に頭を悩ませているのではないのでしょうか。

そんなとき、データベースを使うと、とっても効率的に仕事が進みます。

でも、データベースってなんだか敷居が高そうに思えてしまいますよね。データベースソフトウェアは、使ってみても何をしたらいいのかよくわからなくてあきらめたという人もいるのではないのでしょうか。

ところが、データベースはコンピュータを使う以前の基本的なことだけを理解すれば、そんなに難しいものではありません。

この小冊子を読んで、データベースがそんなに難しいものではないことを知り、少しでも興味を持っていただければ幸いです。

もくじ

- Section.1 失敗事例に学ぼう.....01
 - どれが正しい？ 一名簿作成時の注意点..... 01
 - ある意味奇跡的なできごとから生まれた教訓—名前に一意性はない..... 02
 - 変更処理は—苦勞—情報の分散は避けるべき..... 02
 - 失敗事例から導かれたデータベースの必要性..... 03
- Section.2 データベースの特徴.....04
 - 「情報を一箇所に集中させ、かつ複数の人が同時にアクセスできる」とは？..... 04
 - 「データの重複がなく、矛盾なくデータを保持できる」とは？..... 05
 - 「データの改ざんなどが起こらず、安全性が確保できる」とは？..... 06
 - データベースとは何か..... 06
 - 関係データベース管理システム (RDBMS)..... 07
 - Microsoft SQL Server 2005..... 07
- Section.3 関係データベースの基礎.....08
 - 表計算とデータベースの違い..... 08
 - データベースは設計が命..... 08
 - 関係データベース..... 09
 - 関係データベースの基本概念..... 09
 - 関係データベースの基本的な関係演算..... 10
 - 選択演算..... 10
 - 射影演算..... 10
 - 結合演算..... 10
- Section.4 データの正規化..... 11
 - 第1正規形..... 11
 - 第2正規形..... 12
 - 第3正規形..... 13
 - 主キーと外部キー..... 13
- Section.5 生徒データベースの設計例..... 14
 - 生徒データベース構築以前の問題点..... 14
 - 生徒データベースの論理モデル..... 15
 - 生徒データベースの概念モデル..... 16
 - 生徒データベースの E-R 図..... 16
 - E-R 図とは..... 16

Section.1 失敗事例に学ぼう

ここでは、データベースとは何かという話をする前に、私がこれまで経験した失敗事例をいくつか紹介したいと考えています。ここで紹介する "失敗事例" は、おそらくどの学校でも少なからず経験されていることばかりではないかと思いますが、そのことを通して、データベースがなぜ必要なのかを考えていただければと思います。

どれが正しい？ 一名簿作成時の注意点

私は教員になって一年目のときには教務部に所属し、クラス替えをして次年度の生徒名簿を作成する仕事をしていました。年度末に学年から上がってきた生徒の氏名を元に、学年、組、出席番号、氏名を割り振った CSV ファイルを作成し、本部事務に提出しました。

本当は学年、組、氏名のみを本部事務に提出すればよかったのですが、私はそれを知らなかったため、表計算ソフトウェアを用いてデータの並べ替えを行い、出席番号を割り振ったものを本部事務に提出しました。ところが、本部事務では、受け取ったデータを元に、独自に並べ替え処理を行い、新しい学年、組、出席番号を割り振っていたのです。この時、利用していたソフトウェアが異なったため、並べ替え規則が異なり、同じ生徒でも私が割り振った出席番号と、本部事務が割り振った出席番号が異なってしまいました。

最終的には、本部事務から高校教務部に完成した名簿データが戻ってくるため、それをもとに高校での様々な業務が始まるわけなのですが、実際の現場では、学年団をはじめ様々な部署が一刻も早く新しい名簿を早く欲しがるものです。本部事務で出席番号を割り振ることを知らなかった私は、本部事務からデータが戻ってくるのを待たずに、先生方の求めに応じて私の作った名簿データを渡してしまったため、一年間異なる出席番号が出回ったままになり、現場に混乱をもたらしてしまいました。

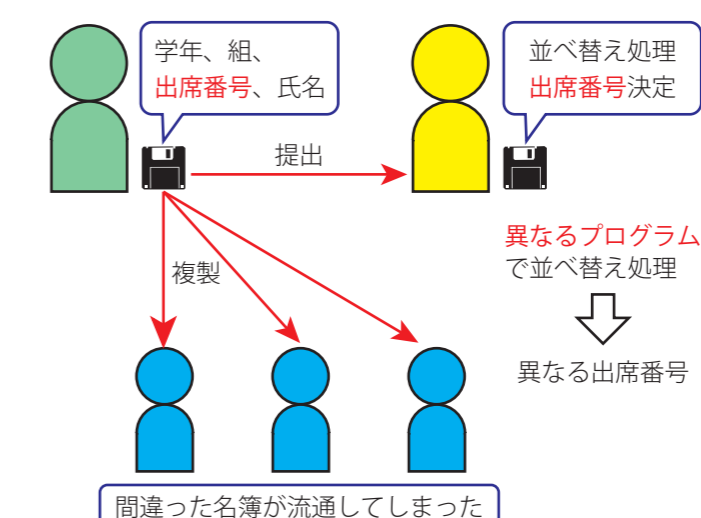


図 1. 並べ替えプログラムの違いによる間違った名簿の流通

ある意味奇跡的なできごとから生まれた教訓—名前に一意性はない

これも私が教員一年目の時に経験したできごとなのですが、生徒指導要録を作成しているときのことでした。生徒指導要録を作成する際に用いたデータは、データベース処理ソフトウェアに保存された成績データから、生徒の「氏名」をキーにして成績データを拾い上げて、一つ一つデータを埋め込んでいくという処理を行っていました。

このとき、どうしても3人の生徒について、特定の科目の成績が二重に拾い上げられ、双方に異なる成績が入るといった現象が発生しました。

この原因は、2つの学年にまたがって3組もの同姓同名の生徒がいるというある意味奇跡的なできごとでした。成績データは年度ごとに全学年の生徒のデータを管理していたため、「氏名」をキーにして照合すると、同姓同名の生徒が存在する場合、他の学年の同姓同名の生徒も拾い上げてしまったのです。

これまで、さまざまな処理において、「氏名」をキーにして照合し、さまざまなデータを併合するという行っていました。この経験を通じて「氏名」が一意性をもち、データの照合に用いるにはあまりふさわしいものではないということが分かりました。

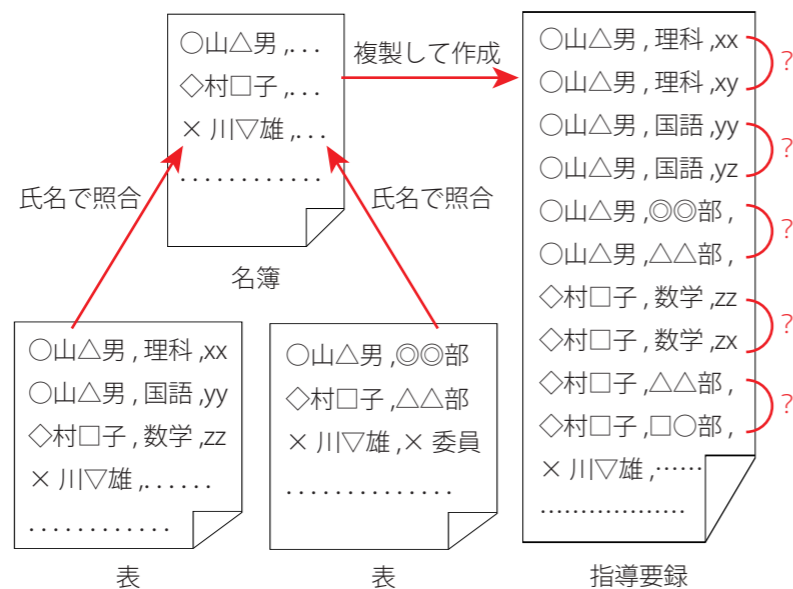


図2. 氏名で照合することによる矛盾の発生

変更処理は一苦労—情報の分散は避けるべき

次年度の準備に慌ただしい年度末や年度始めには、各部署は一刻も早く新年度の生徒名簿のデータを欲しがります。名簿データのコピーはいろいろな部署に出回りますが、一体誰がコピーしたか、誰が名簿を持っているかの全てを把握することはほぼ不可能です。

先に述べた出席番号処理のミスが発生した際、出回った名簿データを一つ一つ修正する必要があったのですが、名簿のコピーをすべて手直ししていくことができず、成績処理において異なる生徒のデータが入力されてしまうなどの重大なミスを招いてしまう結果となりました。

また、年度途中で転入生が発生した際も、それをすべてに反映させることも非常に困難で、各担当者に配布された成績処理用生徒名簿データ内に転入生の履修情報が完全に反映されなかったために、成績入力ができないなどのミスが発生させてしまいました。

データを分散させておくなど必要以上に人の手がかかる管理方法では、データの追加や修正の際に、それを十分に反映させられないミスが生じる可能性があり、データの信頼性はとても低いものになってしまいます。

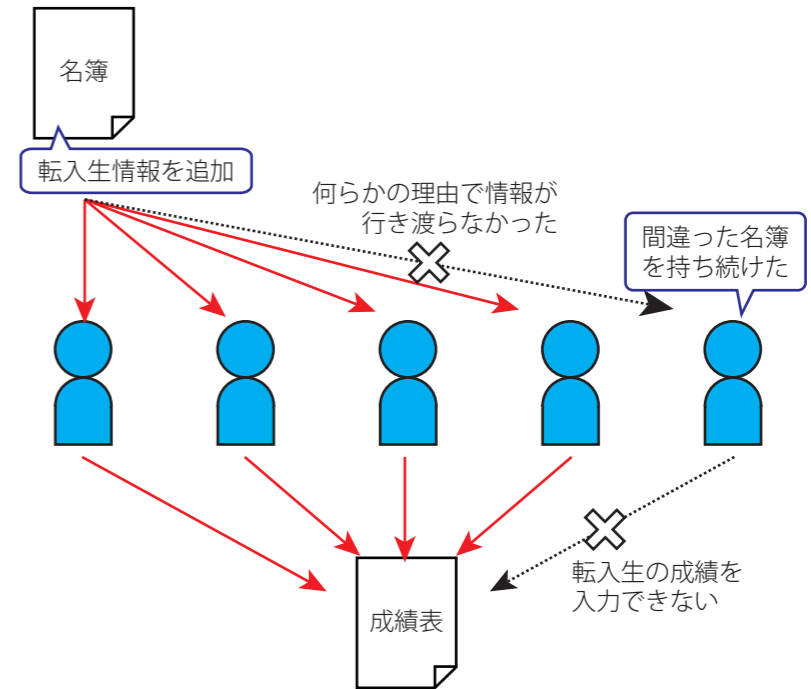


図3. それぞれが持つ名簿を手直るとどこかでミスが起こる

失敗事例から導かれたデータベースの必要性

これらの失敗から、教職員の間で生徒情報を安全かつ集中的・効率的に管理するために生徒の情報をデータベース化して管理する必要性の認識が広がりました。

データベースは、情報を安全かつ集中的・効率的に管理するために、おおよそ次のような条件を満たしている必要があります。

- ◆ 情報を一箇所に集中させ、かつ複数の人が同時にアクセスできる
- ◆ データの重複がなく、矛盾なくデータを保持できる
- ◆ データの改ざんなどが起こらず、安全性が確保できる

よく、Microsoft Excelなどの表計算ソフトウェアではだめなのかという声を聞きますが、表計算ソフトウェアでは、複数の人と同時に情報共有することができませんし、データを矛盾なく保持するのも、VBAなどの高度な技術を用いれば可能かもしれませんが、それだと一部の人のみしか扱えないものになってしまいますので、上述の条件を満たすデータベースを用いるのが適当です。

Section.2 データベースの特徴

前項では、失敗事例を通してなぜデータベースが必要であるかという点について述べました。ここでは、情報をデータベース化して管理することの利点を通して、データベースの必要性を考えます。また、そもそもデータベースとはどのようなものであるかということについてもまとめたいと思います。

「情報を一箇所に集中させ、かつ複数の人が同時にアクセスできる」とは？

図4の左図のように、それぞれの部署が情報を持っていると、どの情報が最新のもので、どの情報が正しい情報のなのかを把握することは、組織の規模が大きくなればなるほど困難になっていきます。

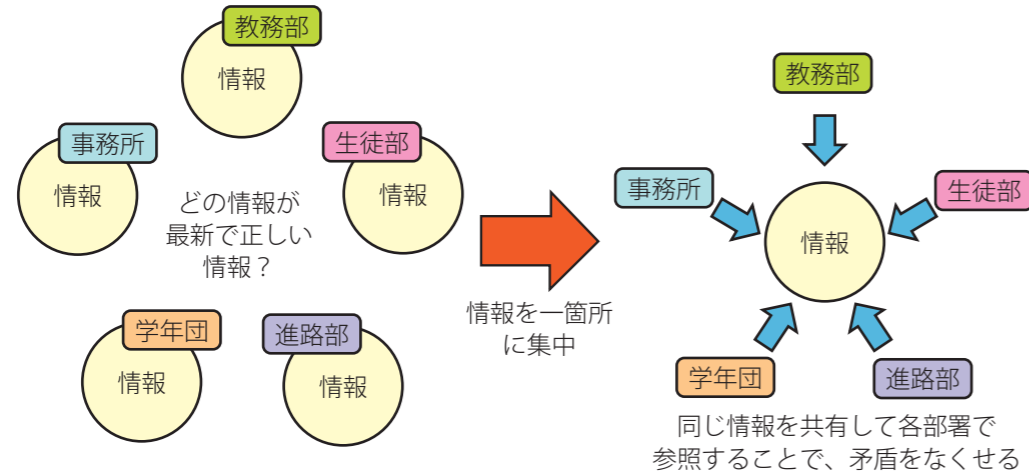


図4. 情報の集中により矛盾を減らせる

そこで、図4の右図のように、情報を一箇所に集中し共有することで、全ての部署が同じ情報を参照することになり、常に最新で正しい情報が一つに定まることになります。

ただし、全ての情報に全ての人アクセスし、書き換えができるということになると、誤ってデータを消去してしまったり、書き換えてしまったりといったことが起こります。そこで、それぞれの情報には、更新や閲覧に関する権限をユーザごとに設定することで、このようなミス回避することができます。

情報が一箇所に集中しているという意味は、物理的に一つのコンピュータやサーバに情報が保存されていて、それを全員が参照しにいくということだけではありません。情報が矛盾なく格納されていなければ、本当の意味で情報を一箇所に集中したことはありません。それが、次に述べる「データの正規化」です。

「データの重複がなく、矛盾なくデータを保持できる」とは？

人間の目には同じように見える情報も、コンピュータは一字一句同じものでなければ異なる情報として扱います。図5のように、一つの表でデータを管理した場合、同じデータを入力したつもりであっても、実は異なるデータが入力されている場合があり、中には人間の目ではなかなかチェックできないものもあります。

生徒番号	組	氏名	科目コード	選択科目	担当	教科コード	教科名	点数
001	1組	伊藤 太郎	101	現代文	富田	1	国語	80
001	1組	伊藤 太郎	201	化学	鈴木	2	理科	96
002	2組	佐藤 花子	102	古典	山本	1	国語	50
002	2組	佐藤 花子	202	物理	浜田	2	理科	48
003	2組	山田 一郎	101	現代文	富田	1	国語	83
003	2組	山田 一郎	202	物理	浜田	2	理科	90

全角の空白、半角の空白 名前の後ろに変な 「富」の字が
の違いが発生している 空白が入っている 微妙に異なる

図5. 正規化しない表に起こる矛盾

この表で、例えば担当者「富田」の成績表を検索しようとするすると、1行目の「富田」は検索されないため、正しくデータを取り出すことができません。上の表で「物理」を選び出そうとしても、人間の目には空白文字は見えないため、「物理」と「物理 」は同じように見えてしましますが、コンピュータには異なるデータとして取り扱われるため、正しくデータを取り出すことができません。同様に、「山田 一郎」さんと「山田 一郎」くんはコンピュータには別人のように扱われてしまうのです。

このような矛盾をなくし、なるだけデータの重複がないようにデータを保持できるようにする措置が「データの正規化」です。上の表を正規化したものが下の図6になります。

【成績表】			【生徒表】		
生徒番号	科目コード	点数	生徒番号	組	氏名
001	101	80	001	1組	伊藤 太郎
001	201	96	002	2組	佐藤 花子
002	102	50	003	2組	山田 一郎
002	202	48			
003	101	83			
003	202	90			

【科目表】				【教科表】	
科目コード	選択科目	担当	教科コード	教科コード	教科名
101	現代文	富田	1	1	国語
102	古典	山本	1	2	理科
201	化学	鈴木	2		
202	物理	浜田	2		

図6. 正規化した表

「データの改ざんなどが起こらず、安全性が確保できる」とは？

前頁でデータを正規化することで情報が一箇所に集中されることで、矛盾がなくなることを述べました。しかし、誰もがすべてのデータに自由にアクセスできるようになってしまうと、誤ってデータを消去したり、データを書き換えてしまったりということが起こってしまいます。そこで、ユーザごとに何ができて何ができないかといった権限を設定する必要になります。

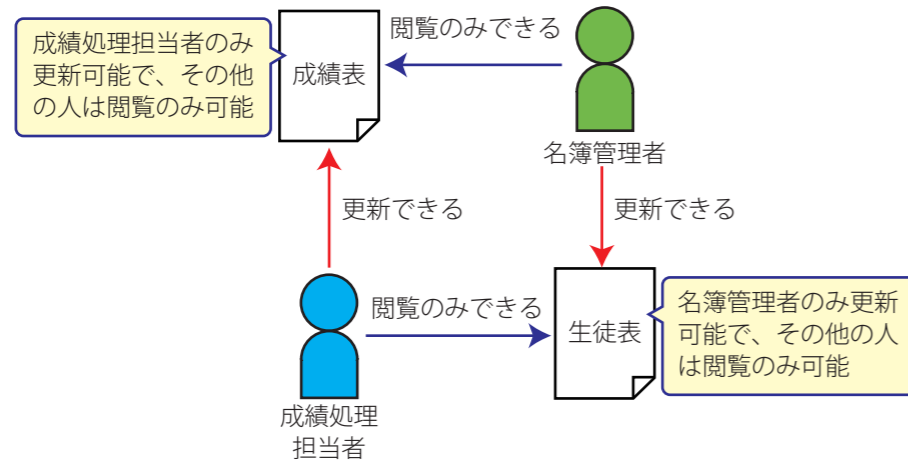


図7. 権限の設定の例

データベースとは何か

データベースは、広義には「データを集めて管理し、必要な時に取り出して再利用することが可能なもの」と定義されており、住所録や辞書なども含め、電子化されていないものもデータベースといえます。コンピュータ上でデータベースを実現し、データの作成や操作、管理などを行うシステムのことをデータベース管理システム（DBMS）といいます。

DBMS で実現されるデータベースは、次のような機能をもつべきだとされています。

- ◆データの保管：一貫性を持ち、矛盾なくデータを保管すること
- ◆データの操作：目的に応じて必要な部分だけを取り出すことができる
- ◆同時実行制御：同時に複数のユーザからのアクセスに対応する
- ◆セキュリティ：ユーザによってデータへのアクセス権限を細かく設定できる
- ◆データの完全性：不正なデータが登録されたり、更新されることがないようにすること

このようなことから、学事情報を安全かつ効率的に管理・運用するには、DBMS を実現するのがよいでしょう。

関係データベース管理システム（RDBMS）

データベースには、管理方法によっていくつかの種類がありますが、その中でも表形式でデータを整理・管理するものを関係データベースといいます。

関係データベースをコンピュータ上で実現するためのシステムを、関係データベース管理システム（RDBMS）といいます。RDBMS を利用すると、設計したデータベースを実際に構築し、利用することができるようになります。

RDBMS は、次のような機能を持っているものをいいます。

- ◆データを管理する機能
データベースを定義したり、データの整合性を確保することができる
障害の復旧などの機能を持っている
- ◆データを操作する機能
各種プログラムからアクセスすることができる
データベースを操作する言語を提供する
(多くの場合、SQL という言語が使われている)
- ◆セキュリティ機能
ユーザごとにデータへのアクセス権限を設定できる
- ◆同時実行制御機能
複数のユーザから同時にアクセスすることができる

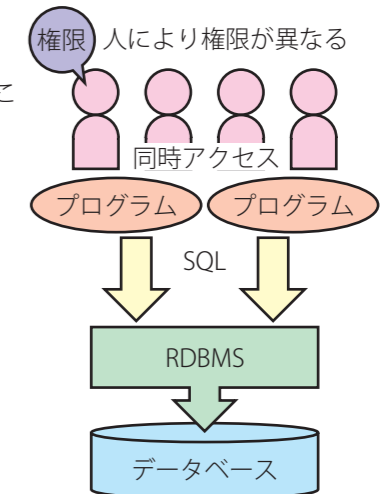


図22.RDBMSの主な機能

RDBMS を使えば、複数の人が異なるソフトウェアやプログラムからでも同じデータベースに接続することが可能で、しかもユーザごとにアクセス権限を設定することができます。

▶ Microsoft SQL Server 2005

ちなみに私の勤務校では、Microsoft SQL Server 2005 という RDBMS を使用しています。

Microsoft SQL Server 2005 は、Microsoft 社製の RDBMS です。Microsoft SQL Server を使えば、Windows ネットワーク上で容易にデータベースを構築できます。また、Windows server の Active Directory によりセキュリティを一本化できます。

Microsoft SQL Server 2005 Express Edition は、Microsoft より無償で提供されているエディションで、GUI で直感的に操作できる管理ツールである SQL Server 2005 Management Studio Express を使うこともできます。

Windows Server 2003 の Active Directory が構成されている環境が好ましいですが、SQL Server で独自のユーザ管理も行えますので、そのような環境のない方にもお試しいただけると幸いです。

Microsoft SQL Server 2005 Express Edition は、「<http://www.microsoft.com/japan/sql/>」よりダウンロードが可能です。

この小冊子を読み終えたら、是非一度触ってみてください。案外簡単にデータベースの管理ができることがわかると思います。また、次の稿では Microsoft SQL Server 2005 を用いたデータベースの構築と管理の実践的な方法について書ければと思っています。

Section.3 関係データベースの基礎

生徒情報管理などを行う際、どのソフトウェアを用いるのかといったことがよく議論されますが、データベースにおいてもっとも大事なことはデータをどのように格納するかという設計です。ここでは、関係データベースの基本的な概念、主な演算方法を通して関係データベースの基礎について解説します。

表計算とデータベースの違い

生徒情報を管理するデータベースを作成する際、それにどのソフトウェアを使うのかということがよく議論されます。私の学校でも、「桐にするのか？ Excelにするのか？」なんていうことがことあるごとに話題になります。

多くの先生方にとっては、表計算ソフトウェアの方がとっつきやすく、使い勝手がよいように思えます。表計算ソフトウェアでも、データベースのようなものをつくることも可能です。

しかし、表計算ソフトウェアでデータベースのようなものをつくっても、管理の仕方が限られてしまい、柔軟に必要なデータだけを取り出したり、別の表のデータと結合をして新しい表を作ったりすることはできません。

データベースでは、様々な表から柔軟にデータを取り出し、必要な表を簡単に作ることが可能です。それによって多角的にデータを分析したり、評価したりすることが可能になります。

また、データベースは、別のソフトウェアからもアクセスすることも可能であり、使い慣れた表計算ソフトウェアで必要なデータを取り出すといったこともできます。これまでに、表計算ソフトウェアを使って作られた帳票などの資産を活用したいという場合、データベースを構築して表計算ソフトウェアからデータベースにアクセスするようになります。

データベースは設計が命

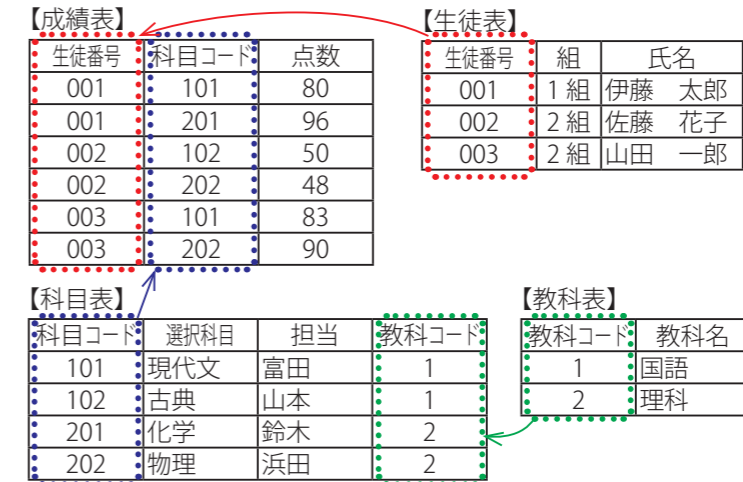
データベースを作成する上で、どのようなソフトウェアを使うかということよりも大切なことがあります。それは、どのような項目を用意し、データをどのように管理するのかということです。つまり、データベースの設計を行うことです。

利用するソフトウェアによって、インターフェイスの使い勝手の良し悪しや、作成できる書類の得手不得手があるのは事実です。しかし、データベースの設計を、使い慣れたソフトウェア上で構築すればよいだけのことで、まず何よりも設計を行うことが大切です。

データベースは、きちんと設計がなされているかが何よりも大切です。設計さえきちんとできていれば、データベースはとても使い勝手のよいものになってくれます。

関係データベース

表形式でデータを格納し、同値のデータをもとに複数の表を結合していく構造をしている。



↓ 同値のデータをもとに表を結合していく

生徒番号	組	氏名	科目コード	選択科目	担当	教科コード	教科名	点数
001	1組	伊藤 太郎	101	現代文	富田	1	国語	80
001	1組	伊藤 太郎	201	化学	鈴木	2	理科	96
002	2組	佐藤 花子	102	古典	山本	1	国語	50
002	2組	佐藤 花子	202	物理	浜田	2	理科	48
003	2組	山田 一郎	101	現代文	富田	1	国語	83
003	2組	山田 一郎	202	物理	浜田	2	理科	90

図9. 関係データベースの概念

▶ 関係データベースの基本概念

関係データベースは、行と列からなる2次元の表形式（テーブル）でデータを保管していきます。

生徒番号	組	氏名	科目コード	選択科目	担当	教科コード	教科名	点数
001	1組	伊藤 太郎	101	現代文	富田	1	国語	80
001	1組	伊藤 太郎	201	化学	鈴木	2	理科	96
002	2組	佐藤 花子	102	古典	山本	1	国語	50
002	2組	佐藤 花子	202	物理	浜田	2	理科	48
003	2組	山田 一郎	101	現代文	富田	1	国語	83
003	2組	山田 一郎	202	物理	浜田	2	理科	90

図10. 関係データベースの行と列

行：横に見た一つ一つのデータそのもの → **ロウ**、**レコード**とも呼ばれる

列：縦に見た一つ一つの項目にあたる → **カラム**、**フィールド**、**項目**などと呼ばれる

関係データベースの基本的な関係演算

関係データベースでは、表に演算を施すことで、様々な形の結果を取り出すことができます。

▶ 選択演算

表から必要な行だけを抜き出します。

例えば、図 10 の表から「選択科目 = 現代文」で選択することで、下図のような結果を得ることができます。

生徒番号	組	氏名	科目コード	選択科目	担当	教科コード	教科名	点数
001	1組	伊藤 太郎	101	現代文	富田	1	国語	80
003	2組	山田 一郎	101	現代文	富田	1	国語	83

図 11. 選択演算を施した結果

▶ 射影演算

表から必要な列だけを抜き出します。

例えば、図 10 の表から、「生徒番号」、「組」、「氏名」で射影することで、下図のような結果を得ることができます。

生徒番号	組	氏名
001	1組	伊藤 太郎
001	1組	伊藤 太郎
002	2組	佐藤 花子
002	2組	佐藤 花子
003	2組	山田 一郎
003	2組	山田 一郎

図 12. 射影演算を施した結果

▶ 結合演算

2つの表の共通する項目（キー）によって関連づけ、新しい表形式の結果を取得します。

例えば、図 9 の「教科表」と「科目表」から「教科コード」をキーとして結合すると、下図のような表形式の結果になります。

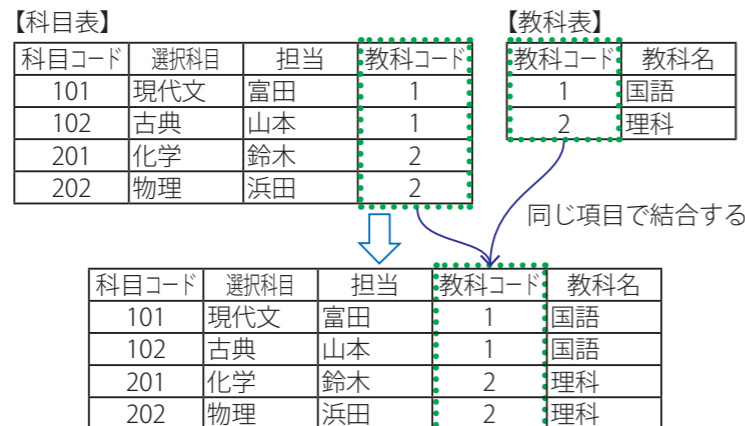


図 13. 結合演算を施した結果

Section.4 データの正規化

結合演算の際の関係をもとに、データの冗長性や不整合を減らしていくことをデータの正規化といいます。データの正規化を行えば、データの更新や削除なども正確に行うことができるようになります。ここでは、データの正規化の方法を通して、関係データベースを設計する方法の1つであるデータの正規化について触れます。

第1正規形

表が、第1正規形を満たしているとは、次の条件を満たしていることをいいます。

◆ 表が単純に定義されている

項目が複合項目になっている

【非正規形】

科目コード	教科名	科目名	受講生		担当者
			生徒番号	氏名	
101	国語	現代文	001	伊藤 太郎	富田
			002	佐藤 花子	
			003	山田 一郎	
102	国語	古典	001	伊藤 太郎	山本
			002	佐藤 花子	
201	理科	化学	001	伊藤 太郎	鈴木
202	理科	物理	002	佐藤 花子	浜田
			003	山田 一郎	

↓

【第1正規形】

科目コード	教科名	科目名	生徒番号	氏名	担当者
101	国語	現代文	001	伊藤 太郎	富田
101	国語	現代文	002	佐藤 花子	富田
101	国語	現代文	003	山田 一郎	富田
102	国語	古典	001	伊藤 太郎	山本
102	国語	古典	002	佐藤 花子	山本
201	理科	化学	001	伊藤 太郎	鈴木
202	理科	物理	002	佐藤 花子	浜田
202	理科	物理	003	山田 一郎	浜田

図 14. 非正規形から第1正規形へ

よく、表計算ソフトウェアを使って表を作るとき、見た目のわかりやすさを重視するあまり、非正規形のような表をつくってしまいがちです。情報を管理するためには少なくとも第1正規形にはしておく必要があります。

非正規形から第1正規形にするには、項目を1行で表現できるようにし、同じデータもすべて表記し、1行だけを取り出してもデータとして成り立つようにします。

第2 正規形

表が第2 正規形を満たしているとは、次の条件を満たしていることをいいます。

- ◆ 第1 正規形であること
- ◆ ある項目の値が決まれば他の項目の値がすべて決定するように表を構成する (これを完全関数従属という)

【第1 正規形】

生徒番号	組	氏名	科目コード	選択科目	担当	教科コード	教科名	点数
001	1組	伊藤 太郎	101	現代文	富田	1	国語	80
001	1組	伊藤 太郎	201	化学	鈴木	2	理科	96
002	2組	佐藤 花子	102	古典	山本	1	国語	50
002	2組	佐藤 花子	202	物理	浜田	2	理科	48
003	2組	山田 一郎	101	現代文	富田	1	国語	83
003	2組	山田 一郎	202	物理	浜田	2	理科	90

【第2 正規形】

【成績表】

生徒番号	科目コード	点数
001	101	80
001	201	96
002	102	50
002	202	48
003	101	83
003	202	90

【生徒表】

生徒番号	組	氏名
001	1組	伊藤 太郎
002	2組	佐藤 花子
003	2組	山田 一郎

【科目表】

科目コード	選択科目	担当	教科コード	教科名
101	現代文	富田	1	国語
102	古典	山本	1	国語
201	化学	鈴木	2	理科
202	物理	浜田	2	理科

図 15. 第1 正規形から第2 正規形へ

第1 正規形の表には、ある項目の値が決まれば、他の項目の値がすべて一つに決定するような項目が存在します。そのような項目を主キー (太枠で囲んだ項目) とした表に分割します。

主キーは、行を一意に決定するためのカギとなる項目のことで、その値は表の中で重複と未定義は許されません。主キーは、「成績表」のように、2つの項目で構成しても構いません。

第3 正規形

表が第3 正規形を満たしているとは、次の条件を満たしていることをいいます。

- ◆ 第2 正規形であること
- ◆ 完全関数従属であり、推移的に他の項目の値が決まるような項目 (推移的関数従属) が存在しないように表を構成する

【第2 正規形】

【科目表】

科目コード	選択科目	担当	教科コード	教科名
101	現代文	富田	1	国語
102	古典	山本	1	国語
201	化学	鈴木	2	理科
202	物理	浜田	2	理科

【科目表】

科目コード	選択科目	担当	教科コード
101	現代文	富田	1
102	古典	山本	1
201	化学	鈴木	2
202	物理	浜田	2

【教科表】

教科コード	教科名
1	国語
2	理科

外部キー

主キー

図 16. 第2 正規形から第3 正規形へ

第2 正規形の「科目表」では、科目コードが決まれば教科コードも教科名も決定します。これを完全関数従属といいますが、同じ表で教科コードが決まれば、教科名も自動的に決定します。このように、主キーから決定した項目にも推移的に決定するような関係 (推移的関数従属) が存在する場合、推移的関数従属を分割して新たに表を作ります。

▶ 主キーと外部キー

- ◆ 主キー：他の項目の値をすべて決定するような項目
- ◆ 外部キー：他の表の主キーと同じ値で表を結合することができるような項目

表を正規化した際には、必ず主キーを設定するようにしましょう。

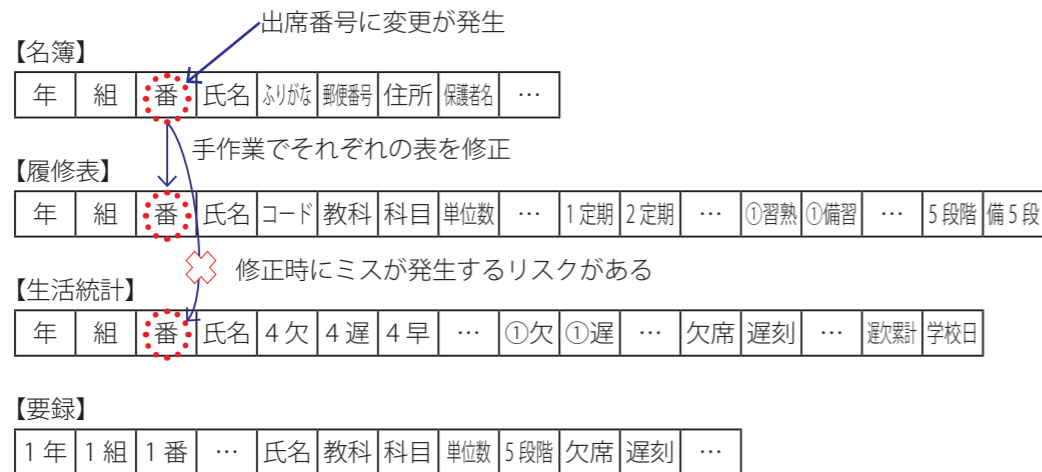
もし、表の中に主キーとなるような項目がない場合は、新たにキーとなるような項目を作るようにしましょう。単純な連番を格納するだけの項目でも構いません。

Section.5 生徒データベースの設計例

ここまで、関係データベースの基本的な概念からデータの正規化についての解説を通して、関係データベースとはどのようなものであるかという概略を解説してきました。ここでは、実際に私の学校で行った生徒データベースの構築方法を例に、データベースの設計方法を解説します。

生徒データベース構築以前の問題点

私の学校では生徒データベースを構築する以前、下図のような表の構成で生徒の情報を管理していました。



それぞれの項目は履修表、生活統計からコピーして手作業で作成

図 17. 生徒データベース構築以前のテーブル構成

生徒データベース構築以前は、それぞれのテーブルが独立して存在していました。データに変更やミスが見つかったときには、それぞれ個別に修正が必要になり、その修正時にミスや見落としがどうしても発生してしまいました。

指導要録を作成する際の要録表を作成する際は、それぞれの表からデータを拾い集め、データをコピーすることで作成していました。その際、「氏名」をキーにしてそれぞれの表を照合し、データをコピーする処理を行っていたため、冒頭で紹介した失敗事例のように、同姓同名の生徒の成績を拾ってしまったり、表によって生徒の名前に使われていた空白文字が全角と半角に違いがあったりなどでデータをうまく反映させることができませんでした。

生徒データベースの構築の必要性を強く感じたのはこのような処理を行っていたときでした。

生徒データベースの論理モデル

実際にデータベースに実装するためのテーブルの構造や列の型、制約などを定義したものを論理モデルといいます。

私の学校で実際に構築したデータベースのテーブルの構成を示したものが下図です。本当はもっとスマートな構造にできたかもしれませんが、過去に作成した帳票の再利用等を考えて、このような形になっています。

【生徒】

生徒ID	XX年	XX組	XX番	...	氏名	ふりがな	郵便番号	住所	保護者名	...
------	-----	-----	-----	-----	----	------	------	----	------	-----

※ XX年、XX組、XX番は、05年、06年のように年度での学籍を示しています。

※生徒IDは、生徒の入学時に一意に定められるコード番号で、生徒証の番号になっています。

【履修】

生徒ID	講座コード	年度	1定期	2定期	...	①習熟	①備習	...	5段階	備5段
------	-------	----	-----	-----	-----	-----	-----	-----	-----	-----

【講座】

講座コード	年度	教科コード	科目コード	担当者	単位数	教室	種別
-------	----	-------	-------	-----	-----	----	----

※講座コードは、この表をもとに、実施年度や教科、科目名を一つにまとめたコードです。

【教科】

教科コード	教科名
-------	-----

【科目名】

科目コード	教科コード	科目コード	科目名
-------	-------	-------	-----

【生活】

生徒ID	①部活動	...	①生徒会	...	①学級活動	...	①所見	...	①資格	...
------	------	-----	------	-----	-------	-----	-----	-----	-----	-----

※「①部活動」「①生徒会」などの「①」は1年生時を示しています。2年生時は「②」。

【月別生活統計】

生徒ID	年度	4指	4欠	4遅	4早	4特活欠	...
------	----	----	----	----	----	------	-----

※「4欠」「4遅」などは、4月の欠席、4月の遅刻などを示し、月ごとの値を格納します。

図 21. 生徒データベースの論理モデル

図 21 の太枠部分は主キーを示しています。ここでは、感じだけをつかんでいただきたいと思いますので、それぞれの項目の意味定義についての説明は省略させていただきます。

なお、論理モデルでは文字列型や数値型といったデータの型や、データの取りうる値の範囲などの制約も同時に決定しますが、ここでは省略しています。

生徒データベースの概念モデル

データベースの設計は、実世界から何をデータベースの管理対象とするかを抽出し、実体間の関係性を記述した概念モデルを作成する概念モデリングと、概念モデルをもとに実際にデータベースへ実装するためのテーブル構造を記述した論理モデルを作成する論理モデリングへと進んでいきます。本稿では分かりやすいように、先に論理モデルから示させていただきます。

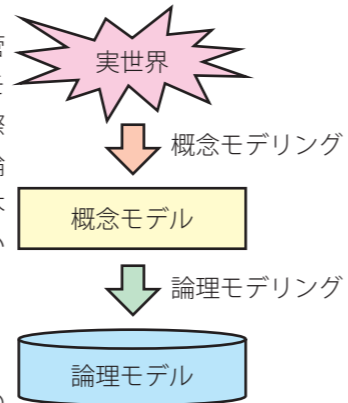


図 18. データベースモデリング

▶ 生徒データベースの E-R 図

前頁で紹介した表の概念モデルを E-R 図を用いて表したものが下の図です。(ただし、実際には現在開発中で未実装な箇所もあります。)

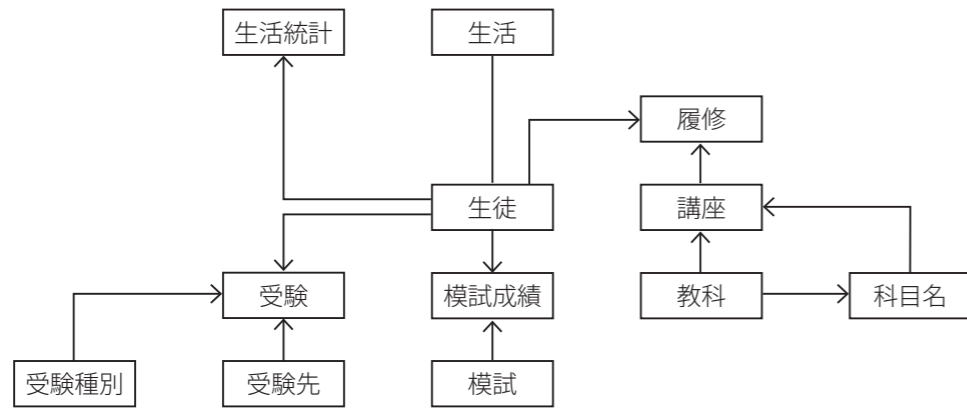


図 19. 生徒データベースの E-R 図

また、この図をもとにして、更にデータベースの拡張を検討することもできます。現状では、まだ受験、模試成績を中心としたデータベースの構築はこれからの課題となっています。

この図で線をたどってつなぐことができる実体同士は、互いに結合することが可能です。たとえば履修と受験を接続すると、卒業した生徒の成績と進路先を対応づけることで、在校生の進路指導に活かすことができたり、履修と模試成績を接続することで、教科における指導がどの程度の実力として現れるかといった関係を分析したりすることができるようになります。

▶ E-R 図とは

上の図のように、実体 (Entity) と関係 (Relation) を図で表したものを E-R 図といい、データベースのテーブルの関係性を示したものを概念モデルといいます。

データベースの関係性は、1 対 1、1 対多、多対多を右の

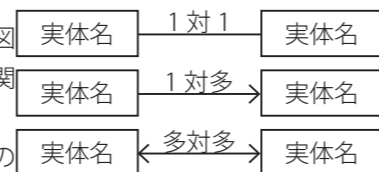


図 20.E-R 図の関係性

近江兄弟社高等学校
長谷川 友彦
hasegawa@ob-sch.ac.jp