「リレーショナル」の意味

これまで学習してきたこと

●データの格納と取り出しについて

・４大命令でテーブルにデータの取得や追加・変更・削除ができる

・WHERE句で処理対象の行を絞り込める

・ORDER BYやDISTINCTなどで検索結果に対して処理を施せる

●データの取得と同時に計算や処理をさせる

・式や関数を用いて、計算や集計をさせることができる

これらの内容だけだとExcelなどの表計算ソフトでも管理が可能です。

本格的にデータベースを活用したシステムを構築する場合、今週以降に学習する数々の

機能を駆使して、**高度なデータ操作**を実現することが可能になります。

# 複数テーブルへのデータ格納

データベースでは管理したい情報により表(テーブル)を作成します。例えば、店舗の情報を管理したい場合は店舗表を作成します。そして店舗に関する情報を表に挿入（追加）していくことで、店舗表を確認すれば店舗の情報を管理することが出来ます。

従業員の情報も同じで、従業員表を作成して従業員の情報を挿入することでデータの管理をすることが出来ます。ここで注目してほしい列がWORK\_STOREです。

WORK\_STORE列は従業員の勤務店舗を管理していますが、店舗に関する情報は店舗表で管理するためここでは「001」や「003」などの数字が格納されています。この数字は店舗表の店舗番号を表しています。

ハンズオン　従業員表にある店舗に関する情報を確認する

1. 店舗表の情報を取得する

SELECT \* FROM STORE ;

1. 従業表から勤務店舗の情報を取得する

SELECT EMP\_NO, ENAME, WORK\_STORE

FROM EMPLOYEE ;

# 外部キーとリレーションシップ

店舗表の店舗番号は**主キー列**のため重複したデータは格納しないので店舗番号が決まれば、どの行のデータかが確定します。そして、従業員表のWORK\_STOREには**店舗表の店舗番号**が格納されています。つまり、WORK\_STOREの値を確認すれば従業員の勤務している店舗の情報がわかるようになっています。

今回の「店舗表」と「従業員表」のように、ある２つのテーブルに情報として関連がある場合、その関連を**リレーションシップ**といいます。また、WORK\_STORE列のように他のテーブルの値を格納することで、リレーションシップを担う列のことを**外部キー(foreign key)**といいます。

●外部キー列の役割

外部キー列は、他のテーブルにある列の値を格納することによって「その行が他のテーブルのどの行と関連しているか」を明らかにします。

# デメリットの克服

複数にテーブルを分けてデータを格納したほうが、データを安全、確実、高速に取り扱うことが出来ます。一方で、WORK\_STOREのように勤務店舗の名前ではなく、店舗番号が管理されているため人間にとっては理解しにくいデメリットが生じます。

そこでデータベースには、管理に適した形態の複数のテーブルから、**人間が理解しやすい形態の1つの結果表**を得る**結合(join)**という機能を備えています。

結合のほかにも、私たちが普段利用するデータベース製品は、複数のテーブルに分けて格納されたデータを関連づけて管理、利用するための様々な機能を有しています。

このようなデータベースを特に**リレーショナルデータベース**(RDB)といい、その中枢を担うデータベース管理システム（DBMS）をRDBMSと言います。

・従業員表 ・店舗表

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| EMP\_NO | ENAME | WORK\_STORE |  | STORE\_NO | SNAME |
| 00001  00002  00003  … | 小林友治  小林圭輔  友保隼平  … | 003  001  001  … | 結合  (join) | 001  002  003  005 | 中崎町店  森ノ宮店  生駒店  阿倍野店 |

テーブルの結合

# 結合の基本的な使い方

・内部結合の基本構文

SELECT 列名

FROM **メインとなる表 JOIN 連結したい表 ON ２つの表の結合条件** ;

ハンズオン　従業員表と店舗表を結合する

1. 従業員表の勤務店舗列と店舗表の店舗番号列を結合して１つの結果表を表示する。

SELECT EMP\_NO, ENAME, SNAME

FROM EMPLOYEE

JOIN STORE ON WORK\_STORE = STORE\_NO ;

従業員表のみだと、勤務している店舗名は店舗表と見比べないと確認できませんでしたが

結合を使うことで従業員名と勤務している店舗名を一緒に表示させることが出来ます。

# 結合の動作イメージ

・間違った結合イメージ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| EMP\_NO | ENAME | WORK\_STORE | STORE\_NO | SNAME |
| 00001  00002  00003  … | 小林友治  小林圭輔  友保隼平  … | 003  001  001  … | 001  002  003  005 | 中崎町店  森ノ宮店  生駒店  阿倍野店 |

・正しい結合イメージ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| EMP\_NO | ENAME | WORK\_STORE |  | STORE\_NO | SNAME |
| 00001  00002  00003  … | 小林友治  小林圭輔  友保隼平  … | 003  001  001  … |  | 001  002  003  005 | 中崎町店  森ノ宮店  生駒店  阿倍野店 |

結合は関連する２つのテーブルは対等な関係ではありません。あくまでもFROM句で指定した表が主役であり、JOIN句で指定したテーブルをON句で指定した条件に基づいてデータを繋いでいきます。

# テーブル名の指定

2つのテーブルを結合すると、同じ名称の列が存在する可能性が出てきます。

店舗表と従業員表にはADDRESS列があるため、２つのテーブルを結合するとADDRESS列が複数存在することになります。この時、以下のSQL文ではどちらのテーブルからADDRESS列を参照すればわからなくなります。

SELECT ADDRESS

FROM EMPLOYEE JOIN STORE ON WORK\_STORE = STORE\_NO

・従業員表 ・店舗表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ENAME | ADDRESS  結合  (join) |  | SNAME | ADDRESS |
| 小林友治  小林圭輔  友保隼平  … | 京都府京都市下～  大阪市中央区大～  大阪市中央区大～  … |  | 生駒店  中崎町店  中崎町店  … | 生駒市菜畑～  大阪市北区中～  大阪市北区中～  … |

※どちらのADDRESSを使ってよいのかわからない

このような場合、列名のみを記載せず「**表名.列名**」のように、どの表に属する列であるかを明示的に指定する必要があります。先ほどのハンズオンのように結合をしても列名が重複しない場合は表名の記載は省略できますが、SQL文の可読性を考えると**結合を伴うときは表名から記載する**ほうが良いでしょう。また、表名が長く複雑な場合はASで**別名を付けておくと列指定や結合条件の記述が簡潔**になります。

ハンズオン　テーブル名を指定して結合を行う

1. 従業員の勤務している店舗の名前と住所を表示する。

SELECT E.ENAME, S.SNAME, S.ADDRESS

FROM EMPLOYEE AS E

JOIN STORE AS S ON E.WORK\_STORE = S.STORE\_NO ;

# イコール以外の結合条件式

結合の条件には等価記号(＝)を使用するとことがほとんどですが、原理的には条件を指定すればよいので＝以外の演算子を用いた条件式を記述することが出来ます。

このような結合を**非等価結合**といい、DBMSにかかる負荷は等価結合より大きくなります。

ハンズオン　非等価結合で表を結合する

1. ランク表(RANKPOINT)の情報を表示して内容を確認する。

SELECT \* FROM RANKPOINT ;

1. 従業員表とランク表の非等価結合を行い、ランクを確認する。

SELECT E.ENAME, E.RANK\_POINT, R.RANK\_NAME

FROM EMPLOYEE AS E

JOIN RANKPOINT AS R ON ( E.RANK\_POINT >= R.LO\_POINT AND

E.RANK\_POINT <= R.HI\_POINT ) ;

# ３テーブル以上の結合

３つ以上のテーブルを結合する場合は、継続してJOIN ～ ON ～を繰り返し記述することで結合するとこが出来ます。この時、テーブルは一度に結合されるわけでは無く、先頭から順に結合処理が行われます。

SELECT 列名

FROM メインとなる表 JOIN 連結表１ ON メインと連結表１の結合条件

JOIN 連結表２ ON 結合された表と連結表２の結合条件

・メインとなる表 ・連結表１

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 列名１ | 列名２  結合 |  | 列名３ | 列名４ |
| … | … |  | … | … |

・連結表２

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 列名１ | 列名２ | 列名３ | 列名４ |  | 列名５ | 列名６ |
| … | … | … | … | 結合 | … | … |