テーブルの結合(続き)

# 結合相手の行がない場合

結合とは、指定された表同士を外部キーの値を元にデータの紐づけを行います。

その為、基本的には外部キーとなる値は結合相手に存在することが前提になります。

では、結合相手に外部キーとなる値が存在しない場合はどうなるでしょうか？

ハンズオン　結合相手が存在しない場合　**※必須**

1. 勤務店舗に存在しない店舗番号と勤務店舗が未定の従業員を登録する。

INSERT INTO EMPLOYEE

VALUES ('00018','山名文和','1980-07-03','0748112222',

'滋賀県東近江市八日市緑町１０－５',DATE(NOW()),NULL,

0,800,**'004'**,NULL);

INSERT INTO EMPLOYEE

VALUES ('00019','秋山賢太','1983-06-24','0729991000',

'兵庫県川辺郡猪名川町上野字北畑１１－１',DATE(NOW()),NULL,

0,800,**NULL**,NULL);

1. 従業員表の情報を確認する。

SELECT \* FROM EMPLOYEE ;

1. 従業員表と店舗表の結合を行う。

SELECT E.EMP\_NO, E.ENAME, E.WORK\_STORE,

S.STORE\_NO, S.SNAME

FROM EMPLOYEE AS E

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

# 左外部結合と右外部結合

先程のように結合相手が存在しない結合は、メインの表にあった行は表示されない仕組みになっています。通常であれば、結合相手がいないので表示されないのは問題なのですが

「勤務店舗が未定の従業員も含めた情報」が欲しいときは、結合相手が居なくても表示される必要が出てきます。このような場合、DBMSには「**結合相手が見つからなくても必ず出力せよ**」と命令を出す**左外部結合(left outer join)**という結合方法があります。

・左外部結合の基本構文

SELECT 列名

FROM メインとなる表 LEFT OUTER JOIN 結合相手の表

※OUTERは省略可能

なお、SQLには右側の表を全行表示させる**右外部結合(right outer join)**と左右の表を全行表示させる完全外部結合(full outer join)も存在します。

・右外部結合の基本構文

SELECT 列名

FROM 結合相手の表 RIGHT OUTER JOINメインとなる表

※OUTERは省略可能

・完全外部結合については、My SQLは対応していないので使用できません。

ハンズオン　テーブルの外部結合を行う。　**※必須**

1. 新規店舗の情報を登録する。（勤務する従業員は未定）

INSERT INTO STORE

VALUES ('006', '尼崎店', '0664879999', '兵庫県',

'尼崎市昭和通２－１７－１６', '9:00:00', '24:00:00', '金', '未オープン');

1. 従業員表と店舗表の左外部結合を行う。

SELECT E.EMP\_NO, E.ENAME, E.WORK\_STORE,

S.STORE\_NO, S.SNAME

FROM EMPLOYEE AS E

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

1. 従業員表と店舗表の右外部結合を行う。

SELECT E.EMP\_NO, E.ENAME, E.WORK\_STORE,

S.STORE\_NO, S.SNAME

FROM EMPLOYEE AS E

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

1. トランザクションの確定を行う。

COMMIT ;

このように外部結合を行った場合、結合相手がいないデータは全てNULLが出力されるようになります。その為、外部結合を行ったときの関数などの取り扱いには注意が必要です。

また、今回のように結合相手が存在しなくても出力する結合を**外部結合**と呼び、結合相手が

存在する場合のみ出力する結合を**内部結合**と呼びます。

# 同じテーブル同士を結合

結合は異なるテーブル間で行われることが一般的ですが、自分自身と結合させることも可能です。同一テーブル同士を結合することを**自己結合(self join)**といいます。自己結合では同じテーブルを複数記述するため、表に別名を必ずつける必要があります。

・従業員表(部下用) ・従業員表(上司用)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| EMP\_NO | ENAME | MGR\_NO |  | EMP\_NO | ENAME |
| 00001  00002  00003  00004  … | 小林友治  小林圭輔  友保隼平  稲田直樹  … | NULL  00001  00002  00002  … |  | 00001  00002  00003  00004  … | 小林友治  小林圭輔  友保隼平  稲田直樹  … |

ハンズオン　自己結合を行う

１．従業員表の自己結合を行い、上司の名前を表示させる。

SELECT E.EMP\_NO, E.ENAME, M.EMP\_NO, M.ENAME

FROM EMPLOYEE AS E

JOIN EMPLOYEE AS M ON ( E.MGR\_NO = M.EMP\_NO)

ORDER BY E.EMP\_NO ;

# 交差結合(クロス結合)について

交差結合とは、２つの表のデータ(行)の全ての組合せを取得する結合方法です。

この結合方法は表1がn件、表２がm件とした場合、データの件数がn×m件になるので状況により非常にデータ件数が増えるので取り扱いには注意が必要です。

・交差結合の基本構文

SELECT 列名

FROM 表名1 CROSS JOIN 表名２;

また、通常のJOIN結合でも**ON句を忘れて結合条件を指定しなかった場合**

**CROSS JOINと同じ動作になる**ので注意が必要です。

・商品表 ・サイズ表

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| PRODUCT\_NO | PNAME | CATEGORY |  | SIZE |
| 0001  0002  0003  … | マルゲリータ  デラックスピザ  シーフードピザ  … | ピザ  ピザ  ピザ  … |  | S  M  L  LL |

|  |  |  |  |
| --- | --- | --- | --- |
| PRODUCT\_NO | PNAME | CATEGORY | SIZE |
| 0001  0001  0001  0001  0002  0002  0002  0002  0003  … | マルゲリータ  マルゲリータ  マルゲリータ  マルゲリータ  デラックスピザ  デラックスピザ  デラックスピザ  デラックスピザ  シーフードピザ  … | ピザ  ピザ  ピザ  ピザ  ピザ  ピザ  ピザ  ピザ  ピザ  … | S  M  L  LL  S  M  L  LL  S  … |

# 集合演算子とは

学習環境のデータベースには、商品を管理する商品表と試作品を管理する試作品表が存在しています。このような仕組みにしておくと、販売中の商品と開発中の商品を明確に分けることが出来ます。しかし、双方の商品を取得する場合、同様のSELECT文を２回発行しなければなりません。このような**同様のSELECT文の結果を組み合わせたい場合は、集合演算子を活用する**とこで、１つの結果表を表示することが出来ます。

なお、集合演算子にはいくつかの種類がありますがMySQLでは、UNION演算子のみ使用可能です。

# UNION - 和集合を求める

UNION演算子は、最も使われる集合演算子になります。２つのSELECT文をUNIONでつないで記述すると、それぞれの検索結果を足し合わせて**和集合**として結果が返ります。

・UNIONの基本構文

SELECT 列名 FROM 表名１

UNION [ALL]

SELECT 列名 FROM 表名２;

※この時、２つのSELECT句の列の数とデータ型は同じにすること。

・和集合(union)の動作イメージ

|  |  |  |  |
| --- | --- | --- | --- |
| PRODUCT\_NO | PNAME | CATEGORY | PRICE |
| 0001  0002  0003  … | マルゲリータ  デラックスピザ  シーフードピザ  … | ピザ  ピザ  ピザ  … | 1200  1700  1500  … |

和集合(union)

|  |  |  |  |
| --- | --- | --- | --- |
| PROTOTYPE\_NO | PROTONAME | CATEGORY | PRICE |
| 9001  9002 | MAXチーズピザ  夏野菜ピザ | ピザ  ピザ | 3000  3000 |

UNIONは和集合の為、表1と表２で同じデータが存在した場合に１件にまとめられます。もし重複行をそのまま出力したい場合はUNION ALLで２つのSELECT文を繋いでく必要があります。

ハンズオン　複数テーブルのデータの和集合を求める

1. 商品表と試作品表の和集合を求める。

SELECT \* FROM PRODUCT

UNION

SELECT \* FROM PROTOTYPE ;