第７章　集合演算

　　７－１　テーブルの足し算と引き算

集合演算とは・・・レコードの集合を使い、レコード同士を足したり消したりすること。

このような計算を集合演算しと呼ぶ。

本章ではこちらのものを使います。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0001 | Tシャツ | 衣服 | 2000 | 500 | 2009-09-20 |
| 0002 | 穴あけパンチ | 事務用品 | 500 | 320 | 2009-09-11 |
| 0003 | カッターシャツ | 衣服 | 3000 | 2800 |  |
| 0004 | 包丁 | キッチン用品 | 3000 | 2800 | 2009-09-20 |
| 0005 | 圧力鍋 | キッチン用品 | 6800 | 5000 | 2009-01-15 |
| 0006 | フォーク | キッチン用品 | 500 |  | 2009-09-20 |
| 0007 | おろしがね | キッチン用品 | 880 | 790 | 2008-04-28 |
| 0008 | ボールペン | 事務用品 | 100 |  | 2009-11-11 |
|  |  |  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0001 | Tシャツ | 衣服 | 1000 | 500 | 2009-09-20 |
| 0002 | 穴あけパンチ | 事務用品 | 500 | 320 | 2009-09-11 |
| 0003 | カッターシャツ | 衣服 | 4000 | 2800 |  |
| 0009 | 手袋 | 衣服 | 800 | 500 |  |
| 0010 | やかん | キッチン用品 | 2000 | 1700 | 2009-09-20 |
|  |  |  |  |  |  |

|  |  |
| --- | --- |
| 0001 | Tシャツ |
| 0002 | 穴あけパンチ |
| 0003 | カッターシャツ |
| 0004 | 包丁 |
| 0005 | 圧力鍋 |
| 0006 | フォーク |
| 0007 | おろしがね |
| 0008 | ボールペン |
| 0009 | 手袋 |
| 0010 | やかん |

この二つの表を使い集合演算しを勉強します。

　　　　　テーブルの足し算

UNION・・・レコードの足し算を行うコマンド。

例

SELECT shohin\_id,shohin\_mei

FROM shohin

UNION

SELECT shohin\_id,shohin\_mei

FROM shohin2;

これで二つの表が合体されました。

ID　0001.0002.0003は重複しているが、UNION句では

重複して表示されずに重複分は排除される。

注意事項

1. 演算対象となるレコードの列数が同じであること。

・・・片方が２列、片方が３列などの表どうしは合体できません。

1. 足し算の対象となるレコードの列のデータ型が一致してること。

　・・・INTEGER型やCHAR型などが一致してない場合エラーになる。

1. SELECT文はどんなものを指定してもいいが、ORDER BYだけは最後に一つだけ。

・・・WHEREなどは両方つけても問題ないが、ORDER BYは並び順なので１つだけね。

　　　　重複行を残すときの方法

　ALLオプション・・・UNIONは基本重複行は排除されるがそれを排除せずに表示するコマンド。

|  |  |
| --- | --- |
| 0001 | Tシャツ |
| 0001 | Tシャツ |
| 0002 | 穴あけパンチ |
| 0002 | 穴あけパンチ |
| 0003 | カッターシャツ |
| 0003 | カッターシャツ |
| 0004 | 包丁 |
| 0005 | 圧力鍋 |
| 0006 | フォーク |
| 0007 | おろしがね |
| 0008 | ボールペン |
| 0009 | 手袋 |
| 0010 | やかん |

例

SELECT shohin\_id,shohin\_mei

FROM shohin

UNION ALL

SELECT shohin\_id,shohin\_mei

FROM shohin2

ORDER BY shohin\_id;

このように0001.0002.0003がUNIONでは重複していた

ので自動削除されていたがALLをつけることによって

重複されている分もすべて表示された。

テーブルの共通部分の選択

　INTERSECT・・・重複された分のみを表示する方法

例

|  |  |
| --- | --- |
| 0001 | Tシャツ |
| 0002 | 穴あけパンチ |
| 0003 | カッターシャツ |

SELECT shohin\_id,shohin\_mei

FROM shohin

INTERSECT

SELECT shohin\_id,shohin\_mei

FROM shohin2 これで先ほどUNIONで重複されていたレコードのみが

ORDER BY shohin\_id; 表示された。

レコードの引き算

EXCEPT・・・レコード同士の引き算をするコマンド。

|  |  |
| --- | --- |
| 0004 | 包丁 |
| 0005 | 圧力鍋 |
| 0006 | フォーク |
| 0007 | おろしがね |
| 0008 | ボールペン |

例

SELECT shohin\_id,shohin\_mei

FROM shohin

EXCEPT

SELECT shohin\_id,shohin\_mei

FROM shohin2

ORDER BY shohin\_id;

これで上の商品テーブルから下の商品２テーブルにあるものがひかれたものが表示された

（商品テーブルにあって商品２テーブルにはないものの表示）

|  |  |
| --- | --- |
| 0009 | 手袋 |
| 0010 | やかん |

SELECT shohin\_id,shohin\_mei

FROM shohin2

EXCEPT

SELECT shohin\_id,shohin\_mei　　ちなみに上と下のテーブルを逆にすると商品２にしかないもの

FROM shohin 　　　　　が表示される。

７－２　結合（テーブルを列方向に連結する）

結合・・・別のテーブルから列を持ってきてから列を増やすという操作（結合　JOIN）

複合キー・・・列の組み合わせで主キーを構成すること。

正規化・・・おなじIDなのに中身が違うなど矛盾が生まれるものを正しい値に戻すこと

主な原因が重複していること。

第一正規化・・・繰り返し項目を別テーブルに分離

第二正規化・・・部分関数従属の関係にあるで＾他を分離する。

部分関数従属・・・主キー以外の項目が、主キーの一部の要素だけで決まるもの。

例・

購入個数は注文番号と商品コードが決まらないとわからない

　　　「注文番号１１１１の商品２２２２は一個」

　商品名や商品単価は商品コードのみわかれば自動的に決定する。

　　　第三正規化・・・推移的関数従属の分離

推移的関数従属・・・

マスター系・・・名簿、商品名などのデータ、辞書のようなもの

トランザクション系・・・システム運用によってどんどん増えていくデータ



|  |  |  |  |
| --- | --- | --- | --- |
| 000A | 東京 | 0001 | 30 |
| 000A | 東京 | 0002 | 50 |
| 000A | 東京 | 0003 | 15 |
| 000B | 名古屋 | 0002 | 30 |
| 000B | 名古屋 | 0003 | 120 |
| 000B | 名古屋 | 0004 | 20 |
| 000B | 名古屋 | 0006 | 10 |
| 000B | 名古屋 | 0007 | 40 |
| 000C | 大阪 | 0003 | 20 |
| 000C | 大阪 | 0004 | 50 |
| 000C | 大阪 | 0006 | 90 |
| 000C | 大阪 | 0007 | 70 |
| 000D | 福岡 | 0001 | 100 |
|  |  |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 0001 | Tシャツ | 衣服 | 2000 | 500 | 2009-09-20 |
| 0002 | 穴あけパンチ | 事務用品 | 500 | 320 | 2009-09-11 |
| 0003 | カッターシャツ | 衣服 | 3000 | 2800 |  |
| 0004 | 包丁 | キッチン用品 | 3000 | 2800 | 2009-09-20 |
| 0005 | 圧力鍋 | キッチン用品 | 6800 | 5000 | 2009-01-15 |
| 0006 | フォーク | キッチン用品 | 500 |  | 2009-09-20 |
| 0007 | おろしがね | キッチン用品 | 880 | 790 | 2008-04-28 |
| 0008 | ボールペン | 事務用品 | 100 |  | 2009-11-11 |

**今回はこの二つの表をつかいます**

内部結合

INNER　JOIN・・・Aに属する列を橋渡しの橋を使ってBに属する列を一緒の結果にまとめること

例

SELECT TS.tenpo\_id,TS.tenpo\_mei,TS.shohin\_id,S.shohin\_mei,S.hanbai\_tanka

FROM tenposhohin AS TS INNER JOIN shohin AS S

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 000A | 東京 | 0001 | Tシャツ | 2000 |
| 000A | 東京 | 0002 | 穴あけパンチ | 500 |
| 000A | 東京 | 0003 | カッターシャツ | 3000 |
| 000B | 名古屋 | 0002 | 穴あけパンチ | 500 |
| 000B | 名古屋 | 0003 | カッターシャツ | 3000 |
| 000B | 名古屋 | 0004 | 包丁 | 3000 |
| 000B | 名古屋 | 0006 | フォーク | 500 |
| 000B | 名古屋 | 0007 | おろしがね | 880 |
| 000C | 大阪 | 0003 | カッターシャツ | 3000 |
| 000C | 大阪 | 0004 | 包丁 | 3000 |
| 000C | 大阪 | 0006 | フォーク | 500 |
| 000C | 大阪 | 0007 | おろしがね | 880 |
| 000D | 福岡 | 0001 | Tシャツ | 2000 |

ON TS.shohin\_id =S.shohin\_id;

店舗商品テーブルをTS

商品テーブルをS　と呼ぶとき

INNERで二つをつなぐ

ONを使って店舗商品テーブルのIDと

商品テーブルのIDが同じものだという

橋渡しをして合体をさせている。

ポイント①

　　FROM句に二つのテーブル名を書いてる

　　がINNER JOINがあることで成立してる。

ポイント②

　　ON句の後にあるのが結合キー

　　二つのテーブルの重複してるところを結合キーとして使おう

ポイント③

　　SELECT句で先にテーブル名の列名を書くことによって表示するものをしっかり定義する。

外部結合

OUTER JOIN・・・片方のものの情報がすべて表示され、それに合わせたもう片方のものが表示される。

例

SELECT TS.tenpo\_id,TS.tenpo\_mei,S.shohin\_id,S.shohin\_mei,S.hanbai\_tanka

FROM Tenposhohin AS TS RIGHT OUTER JOIN shohin AS S

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 000D | 福岡 | 0001 | Tシャツ | 2000 |
| 000A | 東京 | 0001 | Tシャツ | 2000 |
| 000B | 名古屋 | 0002 | 穴あけパンチ | 500 |
| 000A | 東京 | 0002 | 穴あけパンチ | 500 |
| 000C | 大阪 | 0003 | カッターシャツ | 3000 |
| 000B | 名古屋 | 0003 | カッターシャツ | 3000 |
| 000A | 東京 | 0003 | カッターシャツ | 3000 |
| 000C | 大阪 | 0004 | 包丁 | 3000 |
| 000B | 名古屋 | 0004 | 包丁 | 3000 |
|  |  | 0005 | 圧力鍋 | 6800 |
| 000C | 大阪 | 0006 | フォーク | 500 |
| 000B | 名古屋 | 0006 | フォーク | 500 |
| 000C | 大阪 | 0007 | おろしがね | 880 |
| 000B | 名古屋 | 0007 | おろしがね | 880 |
|  |  | 0008 | ボールペン | 100 |

ON TS.shohin\_id =S.shohin\_id;

今回の場合はOUTER　JOINの右側にある

Sの商品テーブルの商品IDが左側のTSの商品テーブルにあるものが表示されていく、

SのID５の圧力鍋はないがその場合はNULLとして表示される。

RIGHTとLEFTは変えることができる。

ライトだと商品IDのものがどこにあるかの表示

店舗IDのところに何があるかが表示される。

なので自分の表示したいものをしっかり見極めて決める必要がある。

　　　　３つ以上のテーブルを使った結合

例

SELECT TS.tenpo\_id,TS.tenpo\_mei,TS.shohin\_id,Shohin\_mei,S.hanbai\_tanka,ZS.zaiko\_suryo

FROM Tenposhohin AS TS

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 000A | 東京 | 0001 | Tシャツ | 2000 | 0 |
| 000A | 東京 | 0002 | 穴あけパンチ | 500 | 120 |
| 000A | 東京 | 0003 | カッターシャツ | 3000 | 200 |
| 000B | 名古屋 | 0002 | 穴あけパンチ | 500 | 120 |
| 000B | 名古屋 | 0003 | カッターシャツ | 3000 | 200 |
| 000B | 名古屋 | 0004 | 包丁 | 3000 | 3 |
| 000B | 名古屋 | 0006 | フォーク | 500 | 99 |
| 000B | 名古屋 | 0007 | おろしがね | 880 | 999 |
| 000C | 大阪 | 0003 | カッターシャツ | 3000 | 200 |
| 000C | 大阪 | 0004 | 包丁 | 3000 | 3 |
| 000C | 大阪 | 0006 | フォーク | 500 | 99 |
| 000C | 大阪 | 0007 | おろしがね | 880 | 999 |
| 000D | 福岡 | 0001 | Tシャツ | 2000 | 0 |

INNER JOIN shohin AS S

ON TS.shohin\_id = S.shohin\_id

INNER JOIN Zaikoshohin AS ZS

　　 ON TS.shohin\_id =ZS.shohin\_id

WHERE ZS.souko\_id = 'S001';

３つ以上のテーブルを合体させるときは、

INNER JOINでひとつひとつを結合させていく

今回はTS商品IDとS商品IDが、

TS商品IDとZS商品IDが等しいということ

を証明することによって３つのテーブルを

合体させることが可能となっている

　　　　クロス結合

CROSS JOIN・・・テーブル同士を結合させる集合演算子、すべてのものにすべてを当てはめる。

※ほとんど使うことはない。