

データベースシステム(14)(15)

SQL演習 (問い合わせ)

SQL問い合わせ

SQLによる問い合わせの基本的な記述方法

```
SELECT <SELECTリスト>
FROM <FROMリスト>
WHERE <選択条件>
```

<FROMリスト>
問い合わせの対象たるテーブル名のリスト

<選択条件>
(AND, OR, NOTで組み合わせられた)真偽を返す演算

<SELECTリスト>
FROMリストのなかにあるテーブルの属性のリスト

SQL問い合わせの(概念的)評価手順

```
SELECT [DISTINCT] <SELECTリスト>
FROM <FROMリスト>
WHERE <選択条件>
```

- (1) <FROMリスト>のテーブルの直積(cross-product)を計算
- (2) <選択条件>を各行に適用し, 偽ならばその行を削除
- (3) <SELECTリスト>にない属性を削除
- (4) もしDISTINCTが指定されているなら重複行を除く

SQL問い合わせの(概念的)評価手順

「103番のボートを予約した船乗りの名前を挙げよ」

```
SELECT S.sname
FROM Sailors S, Reserves R
WHERE S.sid=R.sid AND R.bid=103
```

sid	sname	rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

sid	bid	day
22	101	10/10/96
58	103	11/12/96

(sid)	sname	rating	age	(sid)	bid	day
22	dustin	7	45.0	22	101	10/10/96
22	dustin	7	45.0	58	103	11/12/96
31	lubber	8	55.5	22	101	10/10/96
31	lubber	8	55.5	58	103	11/12/96
58	rusty	10	35.0	22	101	10/10/96
58	rusty	10	35.0	58	103	11/12/96

テーブルに対するSQL問い合わせは,
スキーマ定義さえあれば
テーブルの中身を実際に知らなくても記述できる.

Suppliers(sid integer, sname char(16), address char(20))
Parts(pid integer, pname char(16), color char(16))
Catalog(sid integer, pid integer, cost real)

Suppliersは部品の供給者
(sidはID, snameは供給者名, addressは住所)

Partsは部品
(pidは部品番号, pnameは部品名, colorは色)

Catalogは部品のカタログ
(sidで示される供給者がpidで示される部品をcostの値段で販売)

Suppliers(sid integer, sname char(16), address char(20))
Parts(pid integer, pname char(16), color char(16))
Catalog(sid integer, pid integer, cost real)

(1) addressが"hiroshima"であるSuppliersのsnameを求めよ

```
SELECT S.sname
FROM Suppliers S
WHERE S.address='hiroshima'
```

(2) 赤(red)のPartsを供給するSuppliersのsnameを求めよ

```
SELECT DISTINCT S.sname
FROM Suppliers S, Parts P, Catalog C
WHERE S.sid = C.sid and C.pid = P.pid
and P.color='red'
```

データベースシステム (担当: 森本康彦@広島大 2009/2/12-14)

7

Suppliers(sid integer, sname char(16), address char(20))
Parts(pid integer, pname char(16), color char(16))
Catalog(sid integer, pid integer, cost real)

(3) 赤または緑(green)のPartsを供給するSuppliersのsidを求めよ

```
SELECT DISTINCT C.sid
FROM Parts P, Catalog C
WHERE C.pid = P.pid
and ( P.color='red' or P.color='green' )
```

(4) 赤と緑のPartsを供給するSuppliersのsidを求めよ

```
SELECT C.sid
FROM Parts P, Catalog C
WHERE C.pid = P.pid and P.color='red'
INTERSECT
SELECT C.sid
FROM Parts P, Catalog C
WHERE C.pid = P.pid and P.color='green'
```

8

練習問題

Suppliers(sid integer, sname char(16), address char(20))
Parts(pid integer, pname char(16), color char(16))
Catalog(sid integer, pid integer, cost real)

(5) sidが29のSuppliersのsnameを求めよ

(6) costが100より低いPartsのpidを求めよ

データベースシステム (担当: 森本康彦@広島大 2009/2/12-14)

9

練習問題

Suppliers(sid integer, sname char(16), address char(20))
Parts(pid integer, pname char(16), color char(16))
Catalog(sid integer, pid integer, cost real)

(7) costが100より低いPartsのpnameを求めよ

(8) costが100より低いPartsを供給するSuppliersのsnameを求めよ

データベースシステム (担当: 森本康彦@広島大 2009/2/12-14)

10

- (1) Windowsにログインしてください。
- (2) コマンドプロンプトを起動し、作業フォルダに移動してください。
- (3) データベースに接続してください。
(データベース名は「<自分のログイン名>.db」)

→ sqlite3 ueto.db

```
SQLite version 3.6.1
Enter ".help" for instructions
Enter SQL statements terminated with a ";"
sqlite>
```

これが接続状態のプロンプト
 接続状態で「SQL文」、「SQLiteのコマンド」を実行できます。

データベースシステム (担当: 森本康彦@広島大 2009/2/12-14)

11

Windowsコマンド(「コマンドプロンプト」で使用)

```
dir      現在のフォルダ内のファイルのリスト
         を表示する
cd       現在のどのフォルダにいるかを表示
cd <d>   <d>で示されるフォルダへの移動 例: cd db
         (<d>が「..」のときは一つ上のフォルダ
         への移動となる 例: cd ..)
<d>:     <d>ドライブへ移動 例: z:
del <i>   <i>で示されるファイルを削除 例: del foo.sql
mkdir <d> <d>で示される名前のフォルダを作成
rmdir <d> <d>で示される名前のフォルダを削除
```

データベースシステム (担当: 森本康彦@広島大 2009/2/12-14)

12

SQLiteコマンド(データベースに接続した状態で使用)

```

.read <file>      <file>に書かれたSQL文を実行
.header ON        問い合わせ結果表の列名を表示
.mode tab         問い合わせ結果表をTSV形式で表示
.mode csv         問い合わせ結果表をCSV形式で表示
.tables           データベース内の全テーブル名のリスト
.schema <table>   テーブルのスキーマ
.output <file>    問い合わせ結果表をファイルに出力
.output stdout    問い合わせ結果表をコマンドプロンプトに表示
.import <file> <table> ファイルのデータをテーブルに読み込む
.quit            SQLiteを終了

```

データベースシステム (担当: 森本康彦@広島大 2009/2/12-14) 13

SQLの練習問題

sid	sname	rating	age
22	Dustin	7	45.0
29	Brutus	1	33.0
31	Lubber	8	55.5
32	Andy	8	25.5
58	Rusty	10	35.0
64	Horatio	7	35.0
71	Zorba	10	16.0
74	Horatio	9	35.0
85	Art	3	25.5
95	Bob	3	63.5

sid	bid	day
22	101	10/10/98
22	102	10/10/98
22	103	10/8/98
22	104	10/7/98
31	102	11/10/98
31	103	11/6/98
31	104	11/12/98
64	101	9/5/98
64	102	9/8/98
74	103	9/8/98

bid	bname	color
101	Interlake	blue
102	Interlake	red
103	Clipper	green
104	Marine	red

Sailors Reserves Boats

データベースシステム (担当: 森本康彦@広島大 2009/2/12-14) 14

SQLのDML
(関係代数以外のテーブル操作)

データベースシステム (担当: 森本康彦@広島大 2009/2/12-14) 15

集計演算

```

COUNT (*)
COUNT ( [DISTINCT] A )
SUM ( [DISTINCT] A )
AVG ( [DISTINCT] A )
MAX ( A )
MIN ( A )

```

SELECT COUNT (*) FROM Sailors S

SELECT COUNT (DISTINCT S.sname) FROM Sailors S

SELECT AVG (S.age) FROM Sailors S WHERE S.rating=10

SELECT S.sname FROM Sailors S WHERE S.rating= (SELECT MAX(S2.rating) FROM Sailors S2)

データベースシステム (担当: 森本康彦@広島大 2009/2/12-14) 16

「最も年齢の高い船乗りの名前と年齢を求めよ」

```

SELECT S.sname, MAX (S.age)
FROM Sailors S

```

集計演算がSELECTリストに含まれる場合、後述の「Group by」を伴う場合を除き、SELECTリストの要素は集計演算でなくてはならない

これは行の値 これは集合(グループ)の値

```

SELECT S.sname, S.age
FROM Sailors S
WHERE S.age =(SELECT MAX (S2.age) FROM Sailors S2)

```

sname	age
Bob	63.5

データベースシステム (担当: 森本康彦@広島大 2009/2/12-14) 17

「GROUP BY」と「HAVING」

集計演算をグループ毎に適用したいことがある

「それぞれのratingで最も若い船乗りの年齢を求めよ」

```

SELECT MIN (S.age)
FROM Sailors S
WHERE S.rating = i      For i = 1, 2, ..., 10:

```

ratingの値、それぞれに対し問い合わせるとよい?

一般的にratingの値がいくつあるか、さらにどんな値のratingがあるか問い合わせ以前にはわからない!

データベースシステム (担当: 森本康彦@広島大 2009/2/12-14) 18

「GROUP BY」と「HAVING」

「それぞれのratingで最も若い船乗りの年齢を求めよ」

```
SELECT MIN (S.age)
FROM Sailors S
WHERE S.rating = i For i= 1, 2, ... , 10:
```

```
SELECT S.rating, MIN (S.age)
FROM Sailors S
GROUP BY S.rating
```

rating	min
10	16
9	40
8	25.5
7	35.0
3	25.5
1	33

MINは複数行からなるグループの集計値だが、各グループ内でGROUP BYの値は同じ(なので、S.rating値は各行の値だがグループの値でもある)

「GROUP BY」と「HAVING」

「それぞれのratingの人数とその平均年齢を求めよ」

```
SELECT Count(*), AVG (S.age)
FROM Sailors S
WHERE S.rating = i For i= 1, 2, ... , 10:
```

```
SELECT S.rating, Count(*), AVG (S.age)
FROM Sailors S
GROUP BY S.rating
```

rating	cnt	avg
10	2	25.5
9	1	35.0
8	2	40.5
7	2	40.0
3	2	44.5
1	1	33.0

「GROUP BY」と「HAVING」 つづき

```
SELECT [DISTINCT] <SELECTリスト>
FROM <FROMリスト>
WHERE <選択条件>
GROUP BY <グループリスト>
HAVING <グループ選択条件>
```

<SELECTリスト>は(i)「属性名」、(ii)「集計演算」のリスト

<グループリスト>は

- ・<SELECTリスト>に含まれる属性の部分集合
- ・<グループリスト>属性の各値に対して集計が行われる

<グループ選択条件>はグループに対する選択条件 (<グループリスト>にある属性あるいは、グループの集計値に対する条件のみ)

「それぞれのratingで最も若い船乗りの年齢を求めよ」

```
SELECT S.rating, MIN (S.age)
FROM Sailors S
GROUP BY S.rating
```

rating	min
10	16
9	40
8	25.5
7	35.0
3	25.5
1	33

「5より高いそれぞれのratingで最も若い...」

```
SELECT S.rating, MIN (S.age)
FROM Sailors S
GROUP BY S.rating
HAVING S.rating > 5
```

rating	min
10	16
9	40
8	25.5
7	35.0

「それぞれのratingの人数とその平均年齢を求めよ」

```
SELECT S.rating, Count(*), AVG (S.age)
FROM Sailors S
GROUP BY S.rating
```

rating	cnt	avg
10	2	25.5
9	1	35.0
8	2	40.5
7	2	40.0
3	2	44.5
1	1	33.0

「2人以上のそれぞれのratingの人数とその平均年齢」

```
SELECT S.rating, Count(*), AVG (S.age)
FROM Sailors S
GROUP BY S.rating
HAVING Count(*) >= 2
```

rating	cnt	avg
10	2	25.5
8	2	40.5
7	2	40.0
3	2	44.5

「それぞれのratingの人数とその平均年齢を求めよ」

```
SELECT S.rating, Count(*), AVG (S.age)
FROM Sailors S
GROUP BY S.rating
```

rating	cnt	avg
10	2	25.5
9	1	35.0
8	2	40.5
7	2	40.0
3	2	44.5
1	1	33.0

「平均年齢が40以上のratingを求めよ」

```
SELECT S.rating
FROM Sailors S
GROUP BY S.rating
HAVING AVG(S.age) >= 40
```

rating
8
7
3

GROUP BYの問い合わせの(概念的)評価手順

- ・<FROMリスト>のリレーションに対し直積(cross-product)を計算
- ・<選択条件>を各タプルに適用し、偽ならば削除
- ・<SELECTリスト>にない属性を削除
- ・この時点で残ったタプルを
 - ・<グループリスト>属性の値ごとにグループ分け
 - ・<グループ選択条件>がある場合、その条件を各グループに適用し、偽となるグループを削除
 - ・もしDISTINCTが指定されているなら重複行を除く

「18歳以上である船乗りのなかで各ratingにおけるもっとも若い船乗りの年齢を求めよ。ただし、そのような船乗りが一人以下しかないratingは除く。」

```
SELECT S.rating, MIN (S.age)
FROM Sailors S
WHERE S.age >= 18
GROUP BY S.rating
HAVING COUNT (*) > 1
```

Answer relation

rating	age
3	25.5
7	35
8	25.5

rating	age	sid	sname	rating	age
1	33.0	22	Dustin	7	45.0
3	25.5	29	Brutus	1	33.0
3	63.5	31	Lubber	8	55.5
7	35.0	32	Andy	8	25.5
7	45.0	58	Rusty	10	35.0
8	55.5	64	Horatio	7	35.0
8	25.5	71	Zorba	10	16.0
9	35.0	74	Horatio	9	35.0
10	35.0	85	Art	3	25.5
		95	Bob	3	63.5

「赤いボートそれぞれに対し、そのボートに対する予約の数を求めよ」

```
SELECT B.bid, COUNT (*)
FROM Boats B, Reserves R
WHERE R.bid=B.bid AND B.color='red'
GROUP BY B.bid
```

bid	count
102	3
104	2

```
SELECT B.bid, COUNT (*)
FROM Boats B, Reserves R
WHERE R.bid=B.bid
GROUP BY B.bid
HAVING B.color='red'
```

HAVINGの<グループ選択条件>にはGROUP BYの<グループリスト>に出てくる属性,あるいは,集計演算しが書けない

「ORDER BY」

問い合わせにより得られる結果表を並び替える

```
SELECT [DISTINCT] <SELECTリスト>
FROM <FROMリスト>
[WHERE <選択条件>]
[GROUP BY <グループリスト>]
[HAVING <グループ選択条件>]
[ORDER BY <ORDERリスト>]
```

```
SELECT S.sname, S.rating
FROM Sailors S
ORDER BY S.rating ASC
```

ASCは省略可, DESCは逆順

```
SELECT S.sname, S.rating
FROM Sailors S
ORDER BY S.rating, S.sname DESC
```

sname	rating	sname	rating
Brutus	1	Brutus	1
Bob	3	Art	3
Art	3	Bob	3
Horatio	7	Dustin	7
Dustin	7	Horatio	7
Andy	8	Andy	8
Lubber	8	Lubber	8
Horatio	9	Horatio	9
Zorba	10	Rusty	10
Rusty	10	Zorba	10

クイズ3 A4レポート用紙にまとめて提出

例題のSailors,Reserves,Boatsのテーブルに対する以下の問い合わせのSQL文とその答えを書け

- (1) ボートを予約した人のなかで、赤いボートを予約しなかった人の名前
- (2) 各ボートに対し、その予約者の平均年齢 (結果表はbidと平均年齢)
- (3) 自分より年齢が若いのにランクが上な人がいる人 (結果表はsid, 自分より若くランクが上の人のsid)