

ダウンロード版

データベース問題 (8)

学籍番号 氏名 学年 学科

学籍番号、氏名は全員記入をお願いします。

1. 下記の2つの表に関する問題.

表名 SALES				表名 PRODUCT	
PID	Price	Seller		PID	PNAME
101	1200	x		101	A
102	1010	x		102	B
101	1150	y			
103	800	y			

(1) SALES と PRODUCT を自然結合すると、結果は何になるか？

(2) 表 SALES のリレーションスキーマを「 $R(A_1, \dots, A_n)$ 」の形で書きなさい

2. 下記の表に関する問題.

表名 R		
A	B	C
アイス	食べ物	白
アメ	お菓子	赤
アメ	お菓子	黄
アメ	お菓子	紫
アリ	昆虫	黒

(1) この表を $R[A, B]$ と $R[A, C]$ で置き換えるという分解を行うと、分解結果はどうなるか？

(2) 自然結合 $R[A, B] * R[A, C]$ はどうなるか？

（問 1 の解答例）

(1)

PID	Price	Seller	PNAME
101	1200	x	A
102	1010	x	B
101	1150	y	A
103	800	y	

(2)

SALES(PID, Price, Seller)

（問 2 の解答例）

(1)

A	B	A	C
アイス	食べ物	アイス	白
アメ	お菓子	アメ	赤
アリ	昆虫	アメ	黄
		アメ	紫
		アリ	黒

(2) $R[A, B] * R[A, C]$ は R に等しくなる

テキストの範囲 (ページ 77 ~ 82)

■ リレーシオンの分解(decomposition)

あるリレーションが、リレーシヨンスキーマ $R(A_1, A_2, \dots, A_n)$ のインスタンスであるとする。このリレーシヨンの、 m 個のリレーシオン $X_1, X_2, \dots, X_m$ への分解は次のように定義される。

$X_1, X_2, \dots, X_m$ を R の全属性集合 $\{A_1, A_2, \dots, A_n\}$ の部分集合で、

$$X_1 \cup X_2 \cup \dots \cup X_m = \{A_1, A_2, \dots, A_n\}$$

なる条件を満たすとき (X_i と X_j は共通集合をもってよい)、このリレーシオンを m 個の射影 $R[X_1], R[X_2], \dots, R[X_m]$ で置き換えることを、このリレーシオンの分解という。各 $R[X_i]$ を R の分解成分という。

■ リレーシオンの情報無損失分解

あるリレーシオンが、その分解成分の自然結合をとると元の復元できるとき、その分解は情報無損失分解(information lossless decomposition) であるという。

■ 更新時異状

更新時異状は、修正時異状、挿入時異状、削除時異状の 3 種類がある。データベース設計時の配慮によって、更新時異状のかなりの部分を防ぐことができる。

■ 修正時異状

修正時異状は、リレーシオンのタプルの属性値の修正において発生する異状のこと。

実世界におけるある変化を、あるリレーシオンのタプルの属性値の修正において反映させたいというときに、リレーシオンの設計によっては、リレーシオンの中の複数のタプルを同時に更新せねばならないことがある。これは、つい、**複数のタプルを同時に更新することを忘れてしまい、データベースを壊してしまう** 可能性を生む。

■ 挿入時異状

挿入時異状は、リレーシオンへのタプルの挿入において発生する異状のこと。

非空制約あるいはキー制約があると、その制約が付与されている属性の属性値が空値であるようなタプルは挿入できない。そのことが原因で、実世界におけるある変化を、あるリレーシオンへの**タプルの挿入によってデータベースに反映させたいと思っても、それが出来ない** ことがある。

■ 削除時異状

削除時異状は、リレーシオンからのタプルの削除において発生する異状のこと。

実世界におけるある変化を、あるリレーシオンからのタプルの削除によってデータベースに反映させようとしたとき、そのタプルの削除によって、**他の有用な情報が失われる** ことがある。

(参考)

■ DISTINCT を含む SQL の例 . . . 分解成分を求めたい場合に役立つ

table
'results'

name	teacher_name	student_name	score
Database	K	KK	85
Database	K	AA	75
Database	K	LL	90
Programming	A	KK	85
Programming	A	LL	75

```
SELECT DISTINCT name, teacher
FROM results;
```



重複した値を持つ行は 1 つの行になる
Duplicate Records are Eliminated
from the Query Result

name	teacher_name
Database	K
Programming	A

Query result is one new table

■ テーブルの分解の例

テーブルの分解
(Table Decomposition)

■ 分解前

results(name, teacher_name, student_name, score)

name	teacher_name	student_name	score
Database	K	KK	85
Database	K	AA	75
Database	K	LL	90
Programming	A	KK	85
Programming	A	LL	75

■ 分解後

A(name, teacher_name), B(name, student_name, score)

name	teacher_name
Database	K
Programming	A

name	student_name	score
Database	KK	85
Database	AA	75
Database	LL	90
Programming	KK	85
Programming	LL	75

テーブルの分解と結合

A(name, teacher_name), B(name, student_name, score)

name	teacher_name
Database	K
Programming	A

name	student_name	score
Database	KK	85
Database	AA	75
Database	LL	90
Programming	KK	85
Programming	LL	75

SELECT A.name, A.teacher_name, B.student_name, B.score
FROM A, B
WHERE A.name = B.name



結合問い合わせにより元の
テーブルに戻るので、この
分解は無損失分解である。
The original table 'results' is
constructed.
The decomposition is
LOSSLESS.

name	teacher_name	student_name	score
Database	K	KK	85
Database	K	AA	75
Database	K	LL	90
Programming	A	KK	85
Programming	A	LL	75

テーブルの分解
(Table Decomposition)

■ 分解前

results(name, teacher_name, student_name, score)

■ 分解後

A(name, teacher_name), B(name, student_name, score)



results の全ての属性が、分解後のテーブルに含まれていること。
All attributes in 'results' must be included in the decomposed tables.