

計算機科学実験及演習 4(データベース)課題 4

1029-33-0786

松井 玲

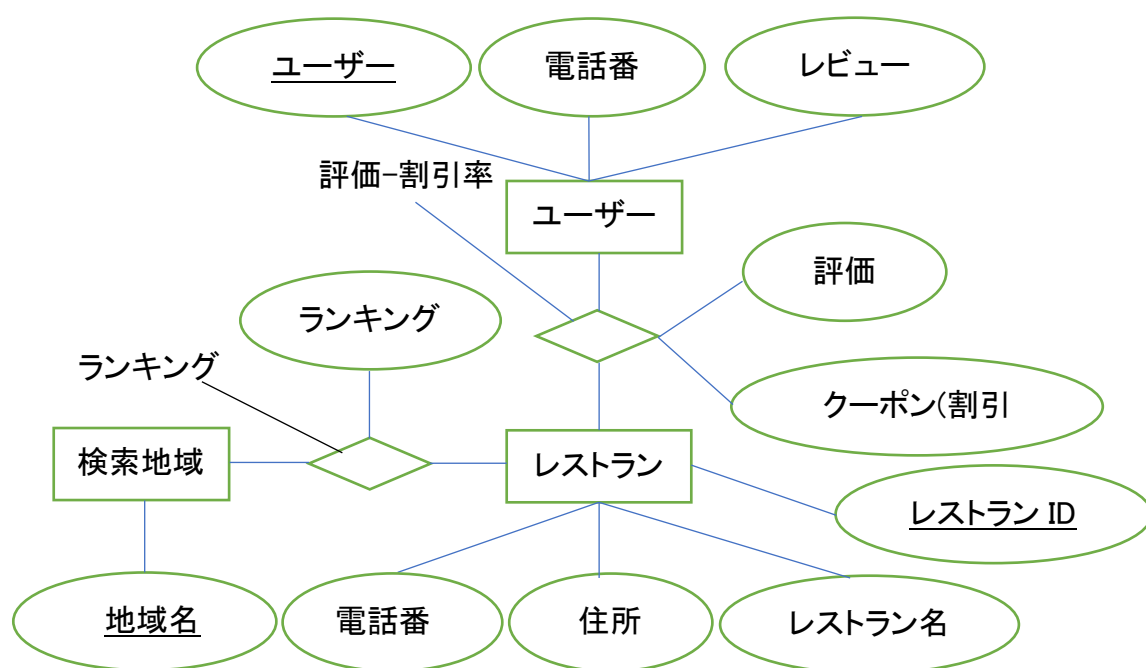
1. 設計した ER 図

ユーザーは名前と電話番号を持ち、ユーザー名は登録時に他ユーザーとの重複を許さないなので、キー属性となる。

ユーザーは各レストランに対し評価(点数)を行い、さらにレストランはユーザーに対しクーポン(割引率)をもつ関連集合が存在する。評価点は 0 から 100 の間の整数とする。

レストランは ID、名前、住所、そして電話番号を持ち、ID によって一意に定まる。

レストランは料理に応じて地域別にまとめられ、検索範囲の地域ごとのランキングを関連集合にもつ。ランキングは自然数である。



2. 設計した関係スキーマ

関係: (ユーザー(ユーザー名、電話番号、レビュー数), Σ (ユーザー))

Σ (ユーザー)は次の一貫性制約を含む

$\sigma 1$: 属性(ユーザー名)に含まれる値に重複はない

$\sigma 2$: 属性(電話番号)に含まれる値に重複はない

$\sigma 3$: $\text{dom}(\text{ユーザー名}) = \text{String}$

$\sigma 4$: $\text{dom}(\text{電話番号}) = \text{String}$

$\sigma 5$: $\text{dom}(\text{レビュー数}) = \text{Integer}$

関数従属性集合 $F = \{\text{ユーザー名} \rightarrow \text{電話番号}, \text{レビュー数}$

$\text{電話番号} \rightarrow \text{ユーザー名}, \text{レビュー数}\}$

ユーザー名は各ユーザーに一つで、重複はない。また、電話番号も同様の性質を持つ。

関係:(レストラン(レストラン ID、レストラン名、電話番号、住所), Σ (レストラン))

Σ (レストラン)は次の一貫性制約を含む

$\sigma 1$: 属性集合{レストラン ID}が主キーである。

$\sigma 2$: $\text{dom}(\text{レストラン ID}) = \text{Integer } (1 \leq \text{レストラン ID})$

$\sigma 3$: $\text{dom}(\text{レストラン名}) = \text{String}$

$\sigma 4$: $\text{dom}(\text{電話番号}) = \text{String}$

$\sigma 5$: $\text{dom}(\text{住所}) = \text{String}$

関数従属性集合 $F = \{\text{レストラン ID} \rightarrow \text{レストラン名、電話番号、住所}$

$\text{レストラン名、住所} \rightarrow \text{レストラン ID、電話番号}$

$\text{電話番号} \rightarrow \text{レストラン ID、レストラン名、住所}\}$

レストラン ID は各ユーザーに一つで、重複はない。また電話番号、レストラン名と住所の組も同様の性質を持つ。

関係:(評価-割引率(ユーザー名、レストラン ID、評価、割引率), Σ (評価-割引率))

Σ (評価-割引率)は次の一貫性制約を含む

$\sigma 1$: 属性集合{ユーザー名、レストラン ID}が主キーである。

$\sigma 2$: $\text{dom}(\text{ユーザー名}) = \text{String}$

$\sigma 3$: $\text{dom}(\text{評価}) = \text{Integer } (0 \leq \text{評価} \leq 100)$

$\sigma 4$: $\text{dom}(\text{割引率}) = \text{Integer } (0 \leq \text{割引率} \leq 100)$

$\sigma 5$: $\text{dom}(\text{レストラン ID}) = \text{Integer } (1 \leq \text{レストラン ID})$

関数従属性集合 $F = \{\text{ユーザー名、レストラン ID} \rightarrow \text{評価、割引率}\}$

ユーザー名とレストラン名の組が決まれば、そのユーザーがレストランに対して書いた評価も一つに定まる。

関係:(ランキング(レストラン ID、地域名、ランキング), Σ (ランキング))

Σ (ランキング)は次の一貫性制約を含む

$\sigma 1$: 属性集合{レストラン ID、地域名}が主キーである。

$\sigma 2$: $\text{dom}(\text{レストラン ID}) = \text{Integer } (1 \leq \text{レストラン ID})$

$\sigma 3$: $\text{dom}(\text{地域名}) = \text{String}$

$\sigma 4: \text{dom}(\text{ランキング}) = \text{Integer} (1 \leq \text{ランキング})$

関数従属性集合 $F = \{\text{レストラン ID}, \text{地域名} \rightarrow \text{ランキング}\}$

レストラン ID と検索範囲の地域名が決まれば、その地域内でのランキングも一つに定まる。

これらの関係スキーマにおいて多値従属性は存在しない。

3. 関係スキーマの正規化

3-1. 関係: (ユーザー (ユーザー名、電話番号、レビュー数), Σ (ユーザー))

Σ (ユーザー) は次の一貫性制約を含む

$\sigma 1$: 属性 (ユーザー名) に含まれる値に重複はない

$\sigma 2$: 属性 (電話番号) に含まれる値に重複はない

$\sigma 3: \text{dom}(\text{ユーザー名}) = \text{String}$

$\sigma 4: \text{dom}(\text{電話番号}) = \text{String}$

$\sigma 5: \text{dom}(\text{レビュー数}) = \text{Integer}$

関数従属性集合 $F = \{\text{ユーザー名} \rightarrow \text{電話番号}, \text{レビュー数}$

$\text{電話番号} \rightarrow \text{ユーザー名}, \text{レビュー数}\}$

この関係の候補キーは属性 (ユーザー名), (電話番号) である。

これは部分関数従属を持たず、データは値であるため第 2 正規形である。

決定項はともに候補キーであるのでボイスコッド正規形である。

3-2. 関係: (レストラン (レストラン ID、レストラン名、電話番号、住所), Σ (レストラン))

Σ (レストラン) は次の一貫性制約を含む

$\sigma 1$: 属性集合 {レストラン ID} が主キーである。

$\sigma 2: \text{dom}(\text{レストラン ID}) = \text{Integer} (1 \leq \text{レストラン ID})$

$\sigma 3: \text{dom}(\text{レストラン名}) = \text{String}$

$\sigma 4: \text{dom}(\text{電話番号}) = \text{String}$

$\sigma 5: \text{dom}(\text{住所}) = \text{String}$

関数従属性集合 $F = \{\text{レストラン ID} \rightarrow \text{レストラン名}, \text{電話番号}, \text{住所}$

$\text{レストラン名}, \text{住所} \rightarrow \text{レストラン ID}, \text{電話番号}$

$\text{電話番号} \rightarrow \text{レストラン ID}, \text{レストラン名}, \text{住所}\}$

この関係の候補キーは属性{レストラン ID}、{電話番号}、{レストラン名、住所}である。
よってこの関係は部分関数従属を持たず、決定項は候補キーであるのでボイスコッド
正規形である。

3-3. 関係: (評価-割引率(ユーザー名、レストラン ID、評価、割引率), Σ (評価-割引率))

Σ (評価-割引率)は次の一貫性制約を含む

$\sigma 1$: 属性集合{ユーザー名、レストラン ID}が主キーである。

$\sigma 2$: $\text{dom}(\text{ユーザー名}) = \text{String}$

$\sigma 3$: $\text{dom}(\text{評価}) = \text{Integer } (0 \leq \text{評価} \leq 100)$

$\sigma 4$: $\text{dom}(\text{割引率}) = \text{Integer } (0 \leq \text{割引率} \leq 100)$

$\sigma 5$: $\text{dom}(\text{レストラン ID}) = \text{Integer } (1 \leq \text{レストラン ID})$

関数従属性集合 $F = \{\text{ユーザー名、レストラン ID} \rightarrow \text{評価、割引率}\}$

この関係の決定項は主キーであるので、これはボイスコッド正規形を満たしている。

3-4. 関係: (ランキング(レストラン ID、地域名、ランキング), Σ (ランキング))

Σ (ランキング)は次の一貫性制約を含む

$\sigma 1$: 属性集合{レストラン ID、地域名}が主キーである。

$\sigma 2$: $\text{dom}(\text{レストラン ID}) = \text{Integer } (1 \leq \text{レストラン ID})$

$\sigma 3$: $\text{dom}(\text{地域名}) = \text{String}$

$\sigma 4$: $\text{dom}(\text{ランキング}) = \text{Integer } (1 \leq \text{ランキング})$

関数従属性集合 $F = \{\text{レストラン ID、地域名} \rightarrow \text{ランキング}\}$

この決定項は主キーであるのでボイスコッド正規形を満たしている。

4. 正規化の手法

3NF や BCNF でない設計から BCNF に変換するためには、以下のステップとアプローチが考えられる。

1. 候補キーの確認:

- BCNF に変換するためには、まずテーブル内の候補キーを確認する。候補キーは、一意性制約を満たす属性または属性の組み合わせである。

2. 関数従属性の特定:

- テーブル内の関数従属性を特定する。

3. 部分関数従属性の解決:

- テーブル内に部分関数従属性が存在する場合、それらを解決する。
これを行うためには、新しいテーブルを作成し、部分関数従属性の属性を移動する。

4. 関数従属性の分解:

- BCNF に変換するために、関数従属性を分解する、つまり関数従属性を満たす新しいテーブルを作成する。
- 次の条件を満たす属性集合 $X, Y, Z (\subseteq V)$ を選ぶ
 1. $XYZ = V$
 2. $\Sigma \models \nu X \rightarrow Y$
 3. 各 $A \in Z$ に対して $\Sigma \not\models \nu X \rightarrow A$
- 属性集合を $(XY), (XZ)$ に分割し、 R 中に $W \subseteq W'$ なる二つの関係スキーマが存在すれば W を削除する

この手法を用い続ければ BCNF に変換することができる。

5. 課題 4 について

- ① 表の定義において、キーの指定により保持できる関数従属性や正規形について考察しなさい。表を正規化していく(例えば分解法や合成法)中で、注目するキーや関数従属性で結果がどのように変化するか(どの正規形になるか、どの関数従属性が保持されるか)を考察しなさい。

まず、分解法は上記 4 で述べた正規化の手法であり、BCNF まで正規化することができるが、テーブルの分解の過程で関数従属性が保持されない場合がある。 $X \rightarrow Y$ が存在するにもかかわらず X と Y のテーブルを分割してしまう場合があるからである。一方で合成法は各関数従属性の両辺を関係スキーマとし、元の関係スキーマのキーを含む関係があれば終了する。この手法は関数従属性を保持できるが第3正規形までの合成しか保証できない。

- ② 課題3で設計した関係スキーマに基づいて関係表を定義しなさい。定義するための SQL 文を示しなさい。なお、各関数従属性が保持されることを文章で説明しなさい。

関係:(ユーザー(ユーザー名、電話番号、レビュー数), Σ (ユーザー))

関数従属性集合 $F=\{\text{ユーザー名} \rightarrow \text{電話番号、レビュー数}$

$\text{電話番号} \rightarrow \text{ユーザー名、レビュー数}\}$

```
CREATE TABLE Users (  
    UserName TEXT NOT NULL,  
    PhoneNumber TEXT NOT NULL,  
    ReviewCount INTEGER,  
    PRIMARY KEY (UserName, PhoneNumber),  
    UNIQUE (UserName),  
    UNIQUE (PhoneNumber)  
);
```

図1: Users テーブルの定義文

関係:(レストラン(レストラン ID、レストラン名、電話番号、住所), Σ (レストラン))

関数従属性集合 $F=\{\text{レストラン ID} \rightarrow \text{レストラン名、電話番号、住所}$

$\text{レストラン名、住所} \rightarrow \text{レストラン ID、電話番号}$

$\text{電話番号} \rightarrow \text{レストラン ID、レストラン名、住所}\}$

```
CREATE TABLE Restaurants (  
    RestaurantID INTEGER PRIMARY KEY,  
    RestaurantName TEXT,  
    PhoneNumber TEXT,  
    Address TEXT  
);
```

図2: Restaurants テーブルの定義文

関係:(評価-割引率(ユーザー名、レストラン ID、評価、割引率), Σ (評価-割引率))

関数従属性集合 $F=\{\text{ユーザー名、レストラン ID} \rightarrow \text{評価、割引率}\}$

```
CREATE TABLE RatingsDiscounts (  
    UserName TEXT,  
    RestaurantID INTEGER,  
    Rating INTEGER,
```

```
DiscountRate INTEGER,
PRIMARY KEY (UserName, RestaurantID),
FOREIGN KEY (UserName) REFERENCES Users(UserName),
FOREIGN KEY (RestaurantID) REFERENCES Restaurants(RestaurantID)
);
```

図3: RatingDiscounts テーブルの定義文

関係: (ランキング(レストラン ID、地域名、ランキング), Σ (ランキング))

関数従属性集合 $F = \{ \text{レストラン ID, 地域名} \rightarrow \text{ランキング} \}$

```
CREATE TABLE Rankings (
    RestaurantID INTEGER,
    RegionName TEXT,
    Ranking INTEGER,
    PRIMARY KEY (RestaurantID, RegionName),
    FOREIGN KEY (RestaurantID) REFERENCES Restaurants(RestaurantID)
);
```

図4: Rankings テーブルの定義文

これらの SQL 文で関係表を定義することができる。

また、関係従属性集合に含まれる関数従属性は分割されていないため、保存されている。

- ③ データを作成して、上記の表に挿入しなさい。データを挿入するための SQL 文を示しなさい。また、データを挿入した表の出力(先頭の一部で結構)を示しなさい。

以下にデータ挿入文と挿入結果(の一部)を示す

```
INSERT INTO Users (UserName, PhoneNumber, ReviewCount)
VALUES
    ('田中 太郎', '555-1111', 5),
    ('山本 さちこ', '555-2222', 8),
    ('佐藤 一郎', '555-3333', 12),
    ('鈴木 由美', '555-4444', 7),
    ('高橋 健太', '555-5555', 3),
    ('渡辺 朋子', '555-6666', 10),
    ('斎藤 裕太', '555-7777', 6),
    ('伊藤 美穂', '555-8888', 9),
    ('中村 雄大', '555-9999', 15),
```


図 5: Users テーブルにデータ挿入文

Reset Filters Records: 32 Search 32 records...

	UserName 🔑 abc 🚩	PhoneNum... 🔑 abc 🚩	ReviewCount # 🚩
	Search column...	Search column...	Search column...
1	田中 太郎	555-1111	5
2	山本 さちこ	555-2222	8
3	佐藤 一郎	555-3333	12
4	鈴木 由美	555-4444	7
5	高橋 健太	555-5555	3
6	渡辺 朋子	555-6666	10
7	斎藤 裕太	555-7777	6
8	伊藤 美穂	555-8888	9

図 6: Users テーブルの挿入結果

```
INSERT INTO Restaurants (RestaurantName, PhoneNumber, Address)
VALUES
  ('居酒屋 たんか', '03-1111-2222', '東京都新宿区 1-1-1'),
  ('寿司 まぐろや', '06-2222-3333', '大阪府大阪市中央区 2-2-2'),
  ('焼肉 こだわり家', '075-3333-4444', '京都府京都市東山区 3-3-3'),
  ('ラーメン 一八', '092-4444-5555', '福岡県福岡市博多区 4-4-4'),
  ('カフェ ゆり', '052-5555-6666', '愛知県名古屋市中村区 5-5-5'),
  ('蕎麦 かずのこ', '011-6666-7777', '北海道札幌市中央区 6-6-6'),
  ('うどん やまと', '082-7777-8888', '広島県広島市中区 7-7-7'),
```

図 7: Restaurants テーブルのデータ挿入文

Reset Filters

Records: 32

Search 32 records...

	UserName 🔑 abc 🚩	PhoneNum... 🔑 abc 🚩	ReviewCount # 🚩
	Search column...	Search column...	Search column...
1	田中 太郎	555-1111	5
2	山本 さちこ	555-2222	8
3	佐藤 一郎	555-3333	12
4	鈴木 由美	555-4444	7
5	高橋 健太	555-5555	3
6	渡辺 朋子	555-6666	10
7	斎藤 裕太	555-7777	6
8	伊藤 美穂	555-8888	9

図 8: Restaurants テーブルの挿入結果

```
INSERT INTO RatingsDiscounts (UserName, RestaurantID, Rating, DiscountRate)
VALUES
  ('田中 太郎', 1, 90, 10),
  ('山本 さちこ', 2, 85, 15),
  ('佐藤 一郎', 3, 78, 12),
  ('鈴木 由美', 4, 92, 8),
```

図 9: RatingDiscounts テーブルのデータ挿入文

Reset Filters

Records: 32

Search 32 records...

	UserName 🔑 abc ⇅	PhoneNum... 🔑 abc ⇅	ReviewCount # ⇅
	Search column...	Search column...	Search column...
1	田中 太郎	555-1111	5
2	山本 さちこ	555-2222	8
3	佐藤 一郎	555-3333	12
4	鈴木 由美	555-4444	7
5	高橋 健太	555-5555	3
6	渡辺 朋子	555-6666	10
7	斎藤 裕太	555-7777	6
8	伊藤 美穂	555-8888	9

図 10: RatingDiscounts テーブルの挿入結果

```
INSERT INTO Rankings (RestaurantID, RegionName, Ranking) VALUES
(1, '関東', 1),
(11, '関東', 3),
(2, '関西', 2),
(12, '関西', 1),
(3, '関西', 3
```

図 11: Rankings テーブルのデータ挿入文

Reset Filters Records: 32 Search 32 records...

	UserName 🔑 abc 🚩	PhoneNum... 🔑 abc 🚩	ReviewCount # 🚩
	Search column...	Search column...	Search column...
1	田中 太郎	555-1111	5
2	山本 さちこ	555-2222	8
3	佐藤 一郎	555-3333	12
4	鈴木 由美	555-4444	7
5	高橋 健太	555-5555	3
6	渡辺 朋子	555-6666	10
7	斎藤 裕太	555-7777	6
8	伊藤 美穂	555-8888	9

図 12: Rankings テーブルの挿入結果