

## 問020034解説

### ◆解答

- 設問1 a イ b エ  
設問2 ウ  
設問3 c ウ  
設問4 d エ

### ◆解説

正規化の手順とSQL文に関する問題である。

#### 正規化の手順について

##### ① 非正規形

属性の値が集合値や複合値で表されるレコードは非正規形である。1レコードの1データ項目に2個以上の値が存在する場合である。趣味という属性は社員コードK0001に関しては、水泳、ドライブの2つの値があり、社員コードY0002に関しては、サッカー、読書の2つの値が存在する。趣味の属性値のように1つの属性に対して、2つの属性値を取るような場合は非正規形である。

社員コード	社員名	所属番号	所属	上位部名称	趣味
K0001	片田	1	営業1課	営業部	水泳、ドライブ
Y0002	吉田	2	営業2課	営業部	サッカー、読書

正規形は1レコードの1データ項目は1つの内容で構成される。受注票やレシートを1つのレコードとして考えると、複数の商品を受注したり、購入する場合、レコードの中の受注明細に相当する部分のデータ項目は1属性に複数のデータが存在することになり、非正規形である。

##### ② 第1正規形

社員コード	社員名	所属番号	所属	上位部名称	趣味
K0001	片田	1	営業1課	営業部	水泳
K0001	片田	1	営業1課	営業部	ドライブ
Y0002	吉田	2	営業2課	営業部	サッカー
Y0002	吉田	2	営業2課	営業部	読書

表において、どの属性の値を取っても、繰り返しなどの集合値や複合値を持たない表は第1正規形である。社員表において社員コードK0001の属性「趣味」について、水泳、ドライブの内容をそれぞれ別個のレコードとして分離して表す場合、第1正規形になる。社員コードK0001に関して、属性「趣味」の内容を水泳、ドライブの2つのレコードに、また、社員コー

ドY0002に関して、属性「趣味」の内容をサッカー、読書の2つのレコードに、それぞれ分離し、4つのレコードにした場合、これらのレコードは第1正規形になる。

第1正規形に関して、社員番号K0001の2つのレコードは、趣味を除くその他の属性の値は全て同じ値であり、冗長性が大きい。これらのレコードの特定の属性値を更新する場合、更新時の整合性に問題が生じ、論理的矛盾を発生する場合がある。また、社員番号Y0002の2つのレコードについても同様の問題がある。これらの問題を解消するための正規化が第2正規化である。

### ③ 第2正規形

第1正規形において、全ての非キーの属性値を、主キーが決まると一意に決まる属性値と一意に決まらない属性値に分離し、2つのレコードを設定する場合、これらのレコードは第2正規形である。社員コードと趣味の2つの属性で趣味レコードを設定し、残りの社員コード、社員名、所属番号、所属、上位部番号の5つの属性をまとめて社員レコードを設定すると、これらのレコードは第2正規形を表すことになる。2つのレコードにおいて、重複している属性は社員コードのみであって、社員コードは2つのレコードの主キーであるため、重複が最小になり、主キーは原則として変更されない属性であるから論理矛盾が発生しないと考えられる。

社員レコード

社員コード	社員名	所属番号	所属	上位部名称
K0001	片田	1	営業1課	営業部
Y0002	吉田	2	営業2課	営業部

趣味レコード

社員コード	趣味
K0001	水泳
K0001	ドライブ
Y0002	サッカー
Y0002	読書

第2正規形において、社員レコードの所属番号、所属、上位部名称に重複が発生しており、これらの属性値に変更処理が生じると、整合性に問題が発生する恐れがある。これらの問題を解消する正規化が第3正規化である。

### ④ 第3正規形

社員レコード

社員コード	社員名	所属番号
K0001	片田	1
Y0002	吉田	2

趣味レコード

社員コード	趣味
K0001	水泳
K0001	ドライブ
Y0002	サッカー
Y0002	読書

所属レコード

所属番号	所属	上位部名称
1	営業1課	営業部
2	営業2課	営業部

第2正規形において、非キー以外の属性で主キーとなり得る属性とその主キーが決まると一意に決まる属性を分離して新しいレコードを設定し、元のレコードから新しいレコードの属性を除いた属性の集まりと新しいレコードの主キーの属性を元のレコードの外部キーとして加えてレコードを設定する。これを第3正規形という。

⑤ 外部キー

第2正規形の社員レコードで、主キーの社員コード以外の属性である所属番号が決まると、所属および上位部名称は一意に決まる。そこでこの3つの属性を分離して、所属番号を主キーとする所属番号、所属、上位部名称からなる所属レコードを設定し、残りの2つの属性群の社員コード、社員名と、所属番号の3つの属性で社員レコードを設定する。社員レコード、所属レコード、趣味レコードの3レコード群は第3正規形である。第3正規形で新しくできた社員レコードの属性「所属番号」のような属性を外部キーという。外部キーとは、その属性を主キーとするレコードが、外部キーが属するレコード外に存在する場合で、外部に存在するレコードを参照できる場合である。

⑥ 第3正規形の求め方

- ① 第2正規形において、非キー以外の属性で主キーとなり得る属性と、その主キーが決まると一意に決まる属性を分離して新しいレコードを設定する。
- ② ①で分離された主キー以外の属性を除いた属性群で新しいレコードを設定する。
- ③ ①、②の対象になったレコードを除く第2正規形のレコードと①、②で設定された新しいレコードで構成されるレコード群は第3正規形である。

### GROUP BYの用法

商品番号毎の受注数量の合計を表示する。

```
SELECT 商品番号, SUM(数量) FROM 受注明細表 GROUP BY 商品番号
```

GROUP BY 句はグループ化を伴う操作であり、「どの表から」、「グループ化して」、「式の結果を取り出す」の構文である。特定の列の値に従って行をグループ化する。特定の列の値が同じ行をまとめて、表をいくつかのグループに分ける場合に使用する。GROUP BYを指定すると、SELECT文節で指定できるものは、GROUP BYで指定した列名またはその列名による集計関数しか記述することができない。GROUP BYによってグループ化する場合、グループの中で値が一意に定まらない列名を、SELECT文節で指定することができない。ただし、GROUP BYで指定した列名をSELECT文節で必ずしも指定する必要はない。

複数の列を対象にグループ化する場合は、GROUP BY 列名, …, 列名の形式を用いる。集計関数を複数個使用する場合は、次の構文になる。

```
SELECT 列名, 集計関数(列名), …, 集計関数(列名) FROM 表名 GROUP BY 列名
```

受注明細表は、(伝票番号、商品番号、顧客番号、数量)のレコード様式である。

設問2の問題の場合、1940年よりも前に生まれた会員が含まれる世帯の世帯番号を求める問題であるから、対称のレコードを世帯番号別にグループ化し、そのグループの中のレコードが条件を満たすならば、その世帯番号が真となり、その番号を出力する。従って、そ

のグループ内に条件を満たすレコードが複数個存在しても結果として出力されるレコードは1つとなる。

設問2の解答中のエのようにGROUP BY 世帯番号、会員番号になると、世帯番号別、会員番号別にグループ化し、そのグループ内で条件に対する真偽を評価するため、世帯番号については同一世帯番号が複数個発生する可能性があることになる。

設問3のSELECT文の分析

```
SELECT 世帯表.班, c
FROM 世帯表, 会員表
WHERE 世帯表.世帯番号 = 会員表.世帯番号
GROUP BY 世帯表.班
```

このSELECT文は、世帯表と会員表を使用して、両表の世帯番号の一致するレコードを世帯表の班別にグルーピングする。グルーピングされたレコードは、同じ班に属する世帯番号別の会員番号レコードの集まりになる。その表を利用して、班内の世帯数と会員数を求める。COUNT(DISTINCT 世帯表.世帯番号)で重複を許さない世帯数のもとまり、COUNT(\*)で班に属するレコード件数すなわち会員数を求めることができる。

### COUNT () 関数

COUNT () 関数は、照会結果の行数を返す。WHERE句を指定すると、検索条件を満たす行数を知ることができる。COUNT () 関数に引数として列名を指定すると、その列に値をもつ行だけがカウントされ、COUNT (\*) の場合はNULLかどうかに関係なく、すべての行がカウントされる。

### 子供会表の作成要領

子供会表の5項目は、会員表から抽出されるのは、会員番号、世帯番号、氏名、生年月日であり、世帯表から抽出されるのは電話番号である。両方の表を結合するのは、両表にある項目である世帯番号を使用し、生年月日が20030402～20090401の条件を満足するレコードが抽出できればよいことになる。その条件は次のようになる。

```
WHERE 会員表.世帯番号 = 世帯表.世帯番号 AND
      会員表.生年月日 BETWEEN 20030402 AND 20090401
```

### 設問 1

a は、非正規形の世帯情報から、会員表を作成する正規化の手順の問題で、この場合の会員表は、会員表、世帯表、続柄表、性別表の各項目が重複なく含まれた表になる。1レコードの項目は次のようになる。

(会員番号)、氏名、続柄、性別、生年月日、(世帯番号)、(世帯主番号)、世帯主氏名、住所、電話番号、登録日、班、備考

第1正規化に基づいて世帯情報を取り出して会員表を作成することになる。求める答えはイとなる。

bは、非キー以外の属性で主キーとなり得る属性とその主キーが決まると一意に決まる属性を分離して新しいレコードを設定し、元のレコードから新しいレコードの属性を除いた属性の集まりと新しいレコードの主キーの属性を元のレコードの外部キーとして加えてレコードを設定する。ここでは、続柄、性別をコード化し、それぞれのコードに対して名称を共通的に定義して用いている。これは、第3正規形の考え方である。答えは第3正規化に基づいて続柄表と性別表を作成したとなり、求める答えはエとなる。

## 設問2

アは、条件の生年月日が1940年以降になっている。

イは、条件の生年月日が1939年12月31日以降になっている。

ウは、世帯番号別にグループ化して、生年月日が1940年よりも前に生まれた会員が存在する世帯の世帯番号を抽出する。求める答えはウとなる。

エは、世帯番号、会員番号別にグループ化して、生年月日が1940年よりも前に生まれた会員が存在する世帯の世帯番号を抽出することになる。従って、世帯番号が同じで、会員番号が異なる場合、同じ世帯番号が複数個抽出される。

## 設問3

世帯表と会員表を使用して、両表の世帯番号の一致するレコードを世帯表の班別にグルーピングすると、同じ班に属する世帯番号別の会員番号レコードの集まりになる。その表を利用して、班内の世帯数と会員数を求めることができる。COUNT(DISTINCT 世帯表.世帯番号)で重複を許さない世帯数が求まり、COUNT(\*)で班に属するレコード件数すなわち会員数を求めることができる。求める答えはウとなる。

## 設問4

会員表と世帯表を結合するのは、両表にある項目である世帯番号を使用し、会員表の生年月日が20030402~20090401の条件を満足するレコードが抽出できればよいことになる。その条件は次のようになる。

```
WHERE 会員表.世帯番号 = 世帯表.世帯番号 AND  
      会員表.生年月日 BETWEEN 20030402 AND 20090401
```

求める答えはエとなる。

ア、ウは、生年月日が2003年4月2日または2009年4月1日の会員が対象になる。

イは、会員表の会員番号と世帯表の世帯主番号で結合しているが、ドメインが等しくない。