問020035解説

◆解答

設問１　ａ　エ　　ｂ　エ　　ｃ　イ

設問２　ア

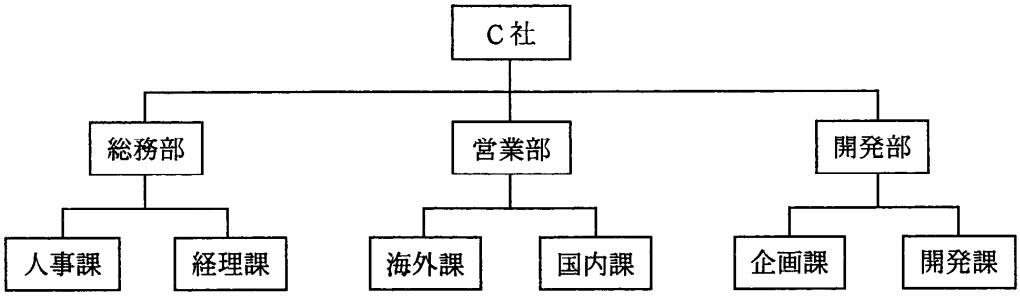
設問３　エ

設問４　ア

◆解説

組織編成の変更に伴う従業員データベースの構築に関する問題である。

変更後の部課組織



変更後の組織編成

① 変更前の組織は部組織のみで、課の組織がなかった。

② 変更後の組織は、各部に２つの課が新設された。

③ 新設された課組織のコード、課名称にＡ、Ｂの２案を考えた。

Ａ案：部署コードと部署名からなる部署表を作成し、部署コードの項目を従業員表に付加した。

Ｂ案：部コードと部名から成る部表と課コード、課名、部コードから成る課表を作成し、課コードの項目を従業員表に付加した。

組織変更前の従業員データベース

部表

部コード	部名

従業員表

従業員番号	氏名	部コード	内線	入社年月日	住所	自宅電話	年齢

変更前の従業員データベースの表構成

変更後の従業員データベース（A案）

部署表

部署コード	部署名
D001	総務部人事課

従業員表

従業員番号	氏名	部署コード	内線	入社年月日	住所	自宅電話	年齢
2005012	情報太郎	D001	211	20020401	東京都…	03-123…	31

A案の表構成とデータの格納例

変更後の従業員データベース（B案）

部表

部コード	部名
D001	総務部

課表

課コード	課名	部コード
S001	人事課	D001

従業員表

従業員番号	氏名	課コード	内線	入社年月日	住所	自宅電話	年齢
2005012	情報太郎	S001	211	20020401	東京都…	03-123…	31

B案の表構成とデータの格納例

正規化とは

正規化はデータの冗長性を少なくして、関連性の強いデータ項目群にまとめ、一事実一カ所になるようにすることである。正規化によって、各データ項目の意味や項目間の関係が明確になり、冗長性がなくなり、意味のあるレコードになり、レコード間の重複が最小限に押さえられることになる。

正規化によって、データ構造は標準化し、柔軟性のあるデータ構造になり、様々なアプリケーションに利用可能になる。正規化されたデータ構造が物理設計の基礎となる。

正規化の目的

① データ項目の意味を正確に定義する。
② データ項目同士の関係を正確に定義する。
③ データの冗長性を取り除く。

④　データの重複を最小限にする。
⑤　データに関係する要件を標準化する。
⑥　データの更新、追加、削除作業の効率化を図る。
⑦　データの整合性を保つ。
⑧　データの属性間の関係を最も少なくなるようにする。

データの整合性

複数の箇所に格納されているデータ間に、論理的な矛盾が発生しないようにすることである。あるマスタレコードは更新され、別のマスタレコードが更新されない事態が発生すると、２つのマスタレコード間の論理的な矛盾が発生することになる。

正規形と外部キー

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。

②　第１正規形

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。

③　第２正規形

第１正規形の表の属性の値には、主キーが決まると一意に決まる値と決まらない値が存在する。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。

④　第３正規形

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

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。

正規化をしない、冗長性があるテーブルでの問題点

①　事前登録不能（登録）

新しい商品を注文の前に登録できない。テーブルにデータを登録するにはキーである注文番号、商品ＩＤが必要で、一度も注文されたことのない商品には注文番号が割り当てられていないため登録ができない。注文と同時でなければ商品が登録できないことになる。

②　重複更新（更新）

商品の名前や単価が変更されたとき、複数ある同一商品の情報をすべて更新しなければ

ならない。同じ商品の情報が注文ごとに繰り返し登録されているため、１つの商品情報を更新するために複数行を残らず更新する必要がある。

③　関係喪失（削除）

一度しか注文されたことのない商品を含む注文が削除されると、商品情報が失われてしまう。注文を削除するという行為により、商品が削除されてしまうことがある。

GROUP BY を用いたＳＱＬの基本文

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に指定できるのは、列名でSELECT文節に表れているものである。SELECT文節にグループ化する列名と集計関数がある場合、GROUP BYの指定がないとエラーになる。また、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SELECFT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、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。

WHERE、GROUP BY、HAVINGを用いたＳＱＬ文の検索例

受注表から２件以上の注文を受けている顧客の最新の受注月日を求める。

```
SELECT 顧客番号,MAX(受注日) FROM 受注表
    GROUP BY 顧客番号 HAVING COUNT(*) >= 2
```

最新の受注日を求めるためにMAX関数を利用する。HAVING文節は、GROUP BYによって、グループ化されたデータに検索条件を設定し、データを絞り込むことができる。

WHEREとGROUP BY を指定すると、検索条件が与えられた後にグループ化が行われる。

GROUP BYとHAVINGを使用すると、グループ化が行われた後に検索条件が与えられる。

GROUP BYと集合関数を使用して値を求め、その結果を条件に従って評価する場合、条件式にHAVINGを使用する。WHEREを使用するとエラーになる。

COUNT(*)は、行数(レコード数)のカウントである。重複を許す場合である。COUNT(DISTINCT,列名)の場合は重複を除いた行数である。

ユーザビューの特徴

①　データの多用性

ビューを利用して表をいろいろな見方でアクセスできる。

② データの独立性

元の表に新たな列が追加されても影響を受けない。

③ データの安全性

他の人に見せたくないデータが隠せる。

④ データ更新の制約

２つ以上の表を元にして作られたり、GROUP BYを使用して定義されたビューでアクセスするデータは、元の表のデータを更新することができない。

⑤ 一つの表の部分定義

１つの表から必要な部分のデータだけを取り出してアクセスできるように定義する。

⑥ 複数の表からの定義

条件を指定して、必要なデータだけのビューを作成することができる。

ユーザビューの定義が果たす機能

① １つの表の部分定義

実際に存在する１つの表またはユーザビューから、必要な部分のデータだけを取り出してアクセスできるようにする。余分なデータがないので、ユーザにとって使いやすくなる。

② 複数の表からの定義

複数の表に対して条件を指定して、必要なデータだけのユーザビューを作成できる。表名の短縮名を定義し、列名と組み合わせたり、列名を各自に分かりやすいものに変えて使用することができる。

③ ユーザビューによる安全保護

効果的な安全保護が可能で、データに不整合を持ち込むようなデータの追加を許さないようにすることができる。

ビュー定義

① 機能

スキーマ中に新しいビュー表を定義する。ビュー表は記憶媒体上に実在しない仮想の表であり、実表から導出する。ビュー表を用いると、データの機密保護や表の別の見方が提示できる。

② 構文

```
CREATE VIEW ビュー表名
        属性名、属性名、…、
        AS SELECT 属性名、属性名、…、
        FROM 表名
        WHERE  導出条件
```

ビュー表名は、新たに作成するビューに対してつけた名前である。同一スキーマの中の実表名、ビュー表名は一意性をもった名前である。属性名は、ビュー表に対する属性定義を記述する。属性名を省略すると、SELECT文で指定した属性名とデータ型を用いる。属性名を指定する場合、ビュー表を構成するすべての属性に対して指定する。ビュー表で指定

した属性名に対して実表の属性のデータ型が継承される。属性名は、実表の属性名と同一でもよいし、異なった名前にしてもよい。属性名はビュー表中の属性を識別するための一意性をもった名前とする。FROM句の表名は、実表名、ビュー表名のどちらでも指定できる。

表定義に関係するＳＱＬ文の機能と構文

① CREATE DATABASE文

機能：データベースを作成する。

構文：CREATE DATABASE データベース名

② CREATE TABLE文

機能：表を作成する。

構文：CREATE TABLE 表名 (列名１ データ型、列名２ データ型、…)

③ CREATE VIEW文

機能：ビューを作成する。

構文：CREATE VIEW ビュー名 (列名、列名、…) AS SELECT文

④ ALTER TABLE文

機能：表の構造を変更する。

構文：追加の場合　ALTER TABLE 表名 ADD 列名 データ型 …

修正の場合　ALTER TABLE 表名 ALTER(MODIFY) 列名 データ型 …

削除の場合　ALTER TABLE 表名 DROP 列名 …

日付に関する条件指定のＳＱＬ文

受注月日が09/01/1991か09/10/1991の顧客番号と受注月日を求める。

```
SELECT 顧客番号, 受注日 FROM 受注表
    WHERE 受注日 = '09/01/1991' OR 受注日 = '09/10/1991'
```

日付に関する条件指示の日付は文字列の形式で指示する。同一の列に対する条件が複数あり、それらが等号による指示の時、IN述語を用いることができる。列名 IN(値の並び)

```
SELECT 顧客番号,受注日 FROM 受注表
    WHERE 受注日 IN('09/01/1991','09/10/1991')
```

日付の期間に関する条件指定のＳＱＬ文

受注月日が８月の顧客番号と受注月日を求める。

```
SELECT 顧客番号,受注日 FROM 受注表
    WHERE 受注日 >= '08/01/1991' AND 受注日 <= '08/31/1991'
```

同一列に対する条件が２つあり、それらが範囲を表すときはBETWEEN述語を用いる。

列名 BETWEEN 値 AND 値の形式で指示する。

```
SELECT 顧客番号,受注日 FROM 受注表
    WHERE 受注日 BETWEEN '08/01/1991'AND '08/31/1991'
```

受注表の構成

受注表

伝票番号	受注日	従業員番号	顧客コード	受注額	納品日

図６　受注表の構成

設問１

ａは、Ａ案の部署表は、部署名として部の名称と課の名称を連結したものを採用している。部の名称を変更する場合には、２つの課の前についている部の名称を変更しなければならない。複数行の変更が必要となる。ａの答えは部名で、求める答えはエとなる。

ｂは、適切に正規化されない場合の問題点で、複数行を修正する必要が生じるのは重複更新の問題である。求める答えはエとなる。

適切に正規化をしていない冗長性があるテーブルでの問題点の一つに、商品の名前や単価が変更されたとき、複数ある同一商品の情報をすべて更新しなければならないことが生じる。同じ商品の情報が注文ごとに繰り返し登録されているため、１つの商品情報を更新するために複数行を残らず更新する必要が生じる。

設問２

```
SELECT 課表.課コード, 課表.課名, AVG(従業員表.年齢)
  FROM 課表, 従業員表
  WHERE 課表.課コード = 従業員表.課コード GROUP BY 課表.課コード、課表.課名
```

課毎の平均年齢を算出するため、課コード別にグループ化して平均年齢を算出する。GROUP BY の後につく項目はSELECT文の後の項目と同じとなるため、GROUP BY 課表.課コード, 課表.課名, となる。WHERE 句の後の下線の部分となり、求める答えはアとなる。

設問３

個人情報保護の観点からビューを作成することにした。表示する内容は従業員番号、氏名、課コード、内線とする。

ビューの構文は次のように定義する。

```
CREATE VIEW ビュー表名
        属性名、属性名、…、
        AS SELECT 属性名、属性名、…、
        FROM 表名
        WHERE  導出条件
```

ビューの表名が従業員公開表であり、属性名は従業員番号、氏名、課コード、内線となる。

ビュー表の定義文は

CREATE VIEW 従業員公開表 AS SELECT 従業員番号、氏名、課コード、内線
FROM 従業員表

答えは下線の部分となり、求める答えはエとなる。

設問４

```
SELECT SUM（受注表. 受注額）
  FROM 受注表, 従業員公開表
  WHERE 従業員公開表. 課コード = 'S101' AND
        受注表.従業員番号 = 従業員公開表.従業員番号 AND
        受注表.受注日 BETWEEN '20110701' AND '20110930'
```

受注表の受注額を使用して、営業部海外課の課コード'S101'で受注表の従業員番号と従業員公開表の従業員番号の一致するレコードで、受注日の期間が２０１１年７月１日から２０１１年９月３０日までの受注額を集計する。

下線の部分が答えで、求める答えはアとなる。