

問020033解説

◆解答

- 設問1 エカ
設問2 a ア
設問3 b ア
設問4 イ

◆解説

書籍を管理する関係データベースの設計および運用に関する問題である。

現在の書籍と貸出状況を管理している書籍表

書籍表

書籍番号	書籍名	著者	出版社	貸出日	返却予定日	社員番号
14030101	情報処理技術の歴史	情報太郎	情報社	null	null	null
14030102	ITで創る未来	東京都子	駒込出版	2014-09-08	2014-09-12	110028

書籍表のデータ格納例

[書籍表に関する説明]

- (1) 利用者に貸出中の書籍には、貸出日と返却予定日が格納されている。これらの項目がnullの書籍は、貸し出されていないことを表す。
- (2) 社員番号には、その書籍を貸し出している利用者の社員番号が格納されている。人事系のシステムで管理している社員表と結合することで、氏名と連絡先が検索できる。
- (3) 返却されたら、貸出日、返却予定日及び社員番号にはnullを設定する。

書籍表

書籍番号	書籍名	著者	出版社
14030101	情報処理技術の歴史	情報太郎	情報社
14030102	ITで創る未来	東京都子	駒込出版

貸出表

貸出番号	書籍番号	社員番号	貸出日	返却予定日	返却日
140158	14030101	080172	2014-09-05	2014-09-08	2014-09-10
140159	14030102	110028	2014-09-08	2014-09-12	null

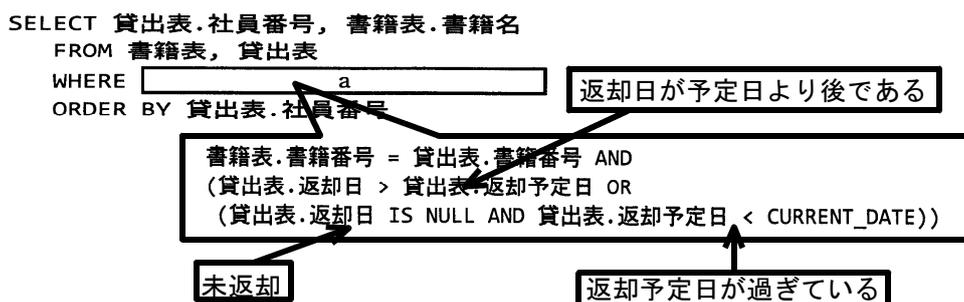
再構築したデータベースの表構成とデータ格納例

書籍表の活用内容から考えると、現時点での貸出状況は把握できるが、過去の貸出実績などを把握することは困難である。従って、書籍ごとの累積貸出状況や利用者ごとの貸出履歴を把握することができない。

〔貸出表に関する説明〕

- (1) 1冊の貸出しに対して一意の貸出番号を付与し、貸出表に情報を記録する。
- (2) 返却日には、返却された日付を格納する。返却日がnullの書籍は、貸出中であることを表す。
- (3) 書籍が返却された後も、貸出表に記録された情報は残す。
- (4) 貸出日、返却予定日及び返却日はDATE型である。

再構築された書籍表、貸出表を用いると、貸出表はトランザクションファイルの役割があるため、過去の貸出実績や利用者別の貸出履歴の情報が把握できるようになる。



この問題は、延滞したことがある利用者の社員番号と、延滞した書籍名を社員番号の昇順に表示したい内容である。WHERE句の内容は、延滞をどのように表現すればよいかを問うている。表現内容は次の2つの内容になる。

- ① 返却日が返却予定日より後である。
- ② まだ返却されていなくて、返却予定日が既に過ぎている。

日付の表現法

① SQL文の検索例 1

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)の形式を用いる。

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

② SQL文の検索例 2

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述語を用いることができる。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'
```

③ SQL文の検索例3

受注月日が09/01/1991か09/10/1991、または8月の顧客番号と受注月日を求めて、顧客番号に続けて「の受注日」を表示する。

```
SELECT 顧客番号||'の受注日', 受注日 FROM 受注表
WHERE 受注日 IN('09/01/1991','09/10/1991') OR
受注日 BETWEEN '08/01/1991' AND '08/31/1991'
```

文字列と文字列を連結するための連結演算子 (||) を含んでいる。

④ SQL文の検索例4

顧客ごとの最も古い受注月日が本日より何カ月と何日前かを求める。

```
SELECT 顧客番号, CURRENT DATE - MIN(受注日) FROM 受注表 GROUP BY 顧客番号
```

顧客ごとにグループ化して日付計算をする例である。CURRENT DATEはシステム変数である。計算結果はYYYYMMDDで表現される。

社員番号の昇順に表示したいという問題に対しては、ORDER BYの使い方が問題となる。

ORDER BYの用法

① SQLの基本文1

日付順に受注情報を表示する。

```
SELECT * FROM 受注表 ORDER BY 受注日
```

受注表

伝票番号	顧客番号	受注日

「どの表から」、「結果を昇順で出力する」の構文である。ORDER BY文節は分類順序を表す。文節にASCを指示すると昇順、DESCを指示すると降順を表し、省略するとASCとなる。

ORDER BY 列名で分類の特定列を指定する。特定の複数列で分類することも可能である。複数列で分類する場合、ORDER BY 列名, ..., 列名の形式になる。最初の列名が分類の第1キ

一となり、順次第2キー、第3キー、…となる。更に、個々の分類キーの昇順、降順を指定する場合、それぞれの列名の後にASC、DESCを指定し、ORDER BY 列名 ASC, 列名 DESC, 列名 ASC, …などの形式で表す。受注表は、(伝票番号、顧客番号、受注日)のレコード様式である。

並べ替えの基準となる列が表の最初の列の場合には、列名を記述する代わりに「ORDER BY 1」で表すこともできる。

② SQL文の検索例2

単価が1000以下の商品を単価の高いもの順に表示する。

```
SELECT * FROM 商品表 WHERE 単価 <= 1000 ORDER BY 単価 DESC
```

GROUP BYの用法

① SQLの基本文1

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。

② SQL文の検索例2

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。GRO

UP 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。

③ SQL文の検索例3

受注明細表からPで始まる商品を受注している顧客数、平均売上数および合計売上数を求める。

```
SELECT 商品番号, COUNT(*), AVG(数量), SUM(数量)
```

```
FROM 受注明細表 WHERE 商品番号 LIKE 'P%' GROUP BY 商品番号
```

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。

伝票番号	商品番号	顧客番号	数量

④ SQL文の検索例4

受注明細表から顧客番号と伝票番号毎の受注件数を求めて顧客番号と伝票番号順に表示する。

```
SELECT 顧客番号, 伝票番号, COUNT(*) FROM 受注明細表
```

```
GROUP BY 顧客番号, 伝票番号 ORDER BY 顧客番号, 伝票番号
```

インデックス

データベースからデータを検索する方法には、順次アクセス方式と直接アクセス方式がある。順次アクセス方式は、目的の行が見つかるまで、すべての行を走査していく方法である。直接アクセス法はインデックスを使用し高速に検索する。インデックスはバイナリツリーを使用し、二分木のルート部から枝の下方に向かって目的のデータに高速に到達する仕組みである。

設問1

解説欄に整理した書籍表の形式や書籍表の活用内容から考えると、現時点での貸出状況は把握できるが、過去の貸出実績などを把握することは困難である。従って、エの書籍ごとの累積貸出状況やカの利用者ごとの貸出履歴を把握することができない。

求める答えはエ、カとなる。

設問2

返却予定日を過ぎてから返却したという表現は次のようになる。

貸出表.返却日 > 貸出表.返却予定日

現在、まだ返却されていない表現は次のようになる。

貸出表.返却日 IS NULL AND 貸出表.返却予定日 < CURRENE_DATE

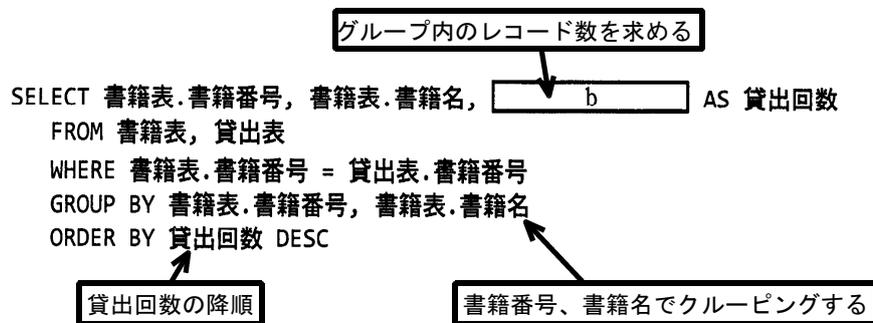
この2つのうちいずれかが成立すればよいから、答えは次のようになる。

書籍表.書籍番号 = 貸出表.書籍番号 AND

(貸出表.返却日 > 貸出表.返却予定日 OR (貸出表.返却日 IS NULL
AND 貸出表.返却予定日 < CURRENE_DATE))

求める答えはアとなる。

設問3



グループ内のレコード数の求め方は、COUNT(*)となる。求める答えはアとなる。

設問4

書籍名を入力して、貸出中かどうかを調べるには、書籍名から書籍表を使用して書籍番号を求める。この場合は書籍表を線形探索する。求めた書籍番号を使用して貸出表から対象の書籍番号の返却日がNULLかどうかを調べる方法を用いる。検索対象のレコード件数が多いのは貸出表であり、貸出表の書籍番号にインデックスが付加されて検索の高速化が行われると検索機能が向上する。最も効果が期待できるのは、貸出表の書籍番号にインデックスが設定されている場合である。求める答えはイとなる。