

会計計算を Stored Procedure で行う（薬剤）続き

前回 Stored Procedure を使用して会計計算をする場合に、薬剤費の計算は単純ではないという話でしたが、薬剤費を求める過程の一つの段階として、薬剤を点数に直して複数行で表示する Stored Procedure を作成しました。

そして、その Stored Procedure をテーブル代わりにしてパラメーターを与えて複数行のフィールドの合計をデータベース上で求めることが出来ました。ということは一つのカルテでたくさんの薬を処方した場合に、一応データベース上ではそのカルテでの薬剤費の合計を求めることが出来たということです。

次にアプリケーション上でその Stored Procedure を使用して点数の合計をデータベース上で求めたように表示できるかということ、これもまた簡単ではありません。

というのは、アプリケーション上で Stored Procedure オブジェクトを置いて、入力パラメーターを指定したとしますと、得られる出力パラメーターは、複数行のフィールドの値のうち、最初のレコードの値だけであって、値の総計ではないからです。これはもともと作成した Stored Procedure が複数行を返すようになっているのですから当然の結果です。

さてそれではそれをふまえてアプリケーション上で一時的な SQL を作成して複数行で表示されたフィールド値の合計を得るか、あるいは最初からデータベース上で総計を出力する Stored Procedure をつくるかどちらかにするということが必要になります。

構造上も操作上も簡単なのは後者ですから、データベース上で「ある Stored Procedure により抽出される複数行レコードの合計値を求めるもう一つ別の Stored Procedure を作る」ということにします。

< データベース上の作業 >

一つの Stored Procedure（複数行を返す）をテーブル代わりに呼び出して複数行の合計を1行として出力する Stored Procedure を作成することは簡単です。

ある Stored Procedure から呼び出された Stored Procedure をネストされたといいます。

新たに作る Stored Procedure は < GET_YAKUZAI_TEN > という Stored Procedure をネストします。

ところで、ネストされた Stored Procedure は入力パラメーターを取る必要がありました。

その Stored Procedure を呼び出す Stored Procedure もまた入力パラメーターを指定する必要があります。

なぜなら、親 Stored Procedure が入力パラメーターを受け取る用意が出来ていないと、ネストされた Stored Procedure もまたどこからも入力パラメーターを受け取る機会がなくなり、開いたときにエラーとなるからです。そこで親 Stored Procedure とネストされた Stored Procedure のそれぞれにどのように必要な入力パラメーターを与えるかということが問題になります。

ネストされる予定の Stored Procedure を再確認します。

```

CREATE PROCEDURE GET_YAKUZAI_TEN(KARTE_NO INTEGER)
RETURNS (naifukuyaku_ten DOUBLE PRECISION,nissuu INTEGER,tani_ten
INTEGER,ten INTEGER)
AS
DECLARE VARIABLE TANNI_YAKKA DOUBLE PRECISION;
DECLARE VARIABLE HINITI INTEGER;
BEGIN
FOR SELECT SUM(YAKKA*RYOU),MAX(DAYS)
FROM KARTE_DRUG
WHERE KARTE_NO=:karte_no
GROUP BY DAYS,TEKIYOU_CODE
INTO TANNI_YAKKA,HINITI
DO
BEGIN
nissuu=HINITI;
naifukuyaku_ten=TANNI_YAKKA;
IF (naifukuyaku_ten<=15) THEN
tani_ten=1;
ELSE
tani_ten=CAST(TANNI_YAKKA/10-1 AS INTEGER);
ten=nissuu*tani_ten;
SUSPEND;
END
END;

```

上の Stored Procedure は KARTE_NO を入力パラメーターにしています。
 入力パラメーターである KARTE_NO はまず、Stored Procedure 名に続いて宣言されて
 いますが、同じようにネストされた Stored Procedure に与えるための入力パラメータ
 ーもおなじように親 Stored Procedure 名に続いて宣言します。

次に実際のコード上で BEGIN FOR 節以下では WHERE KARTE_NO=:karte_no というよう
 に
 SQL 文のようにしてパラメーターを受け取っていました。それと同じようにネストする
 ときも WHERE 文でパラメーターを与えればよいかというと、そうでなく
 Stored Procedure 名のあとにすぐ括弧内に入力パラメーターを指定します。
 つまり

```

-----
CREATE PROCEDURE GET_YAKUZAITEN_GOIUKEI(KARTE_NO INTEGER)
RETURNS (tensuu INTEGER)
AS
BEGIN

```

```

SELECT SUM(TEN) FROM GET_YAKUZAI_TEN
WHERE KARTE_NO=:karte_no
INTO tensuu;
END

```

とするのではなく、正しくは以下ようになります。

```

CREATE PROCEDURE GET_YAKUZAITEN_GOIUKEI(KARTE_NO INTEGER)
RETURNS (tensuu INTEGER)
AS
BEGIN
SELECT SUM(TEN) FROM GET_YAKUZAI_TEN(:KARTE_NO)
INTO tensuu;
END

```

つまり Stored Procedure をデータベース上で呼び出した方法に準じて

```
GET_YAKUZAI_TEN(:KARTE_NO)
```

という形で入力パラメーターにします。

このようにして親 Stored Procedure にパラメーターを与えたと、親 Stored Procedure を開いたときにネストされた Stored Procedure にもパラメーターが渡され、ネストした Stored Procedure → 呼び出した親 Stored Procedure という順に処理が実行され、求める結果を得ることができます。

以上で <GET_YAKUZAI_TEN> でえられる複数の薬剤の合計点数を親 Stored Procedure から呼び出すことによって一個の結果として得ることができるようになりました。

これでカルテに入力された複数の薬剤を取りまとめて薬剤費合計を算出する準備が整いました。

次に実際のアプリケーション上で特定のカルテ番号の薬剤費合計を算出・表示させてみます。

アプリケーションでパラメーター付きの Stored Procedure を使用するには
順序として

- 入力パラメーターを与える

- Stored Procedure を開く

- 出力パラメーターを得る

という手順が必要でした。

ボタンをクリックしたらラベル上に Stored Procedure の結果を表示するという一連の手続きは以下ようになります。

```

procedure TKarte.KaikeiButtonClick(Sender: TObject);
begin

```

```
SP_GET_YAKUZAITEM_GOJUKEI.Params[0].Value:=StrToInt(KarteForm.DBEditKarte  
No.Text);  
SP_GET_YAKUZAITEM_GOJUKEI.ExecProc;  
Label15.Caption:=IntToStr(SP_GET_YAKUZAITEM_GOJUKEI.Params[1].Value);  
end;
```

このプロシージャ-をカルテ番号 <10131> のカルテ上で実行します、Label15 のラベルに 2250 と表示されます。

これは

1174+112+14+350 の合計になります。

以上のように薬剂の場合は複数の薬剂を服用法・投与日数でグループ化したうえで単位点数を計算し、投与日数をかけてグループごとに算定し、最終的にその合計値が求める薬剂費ということになります。

同じように注射欄に入力した注射の点数の計算も同様の原理に従って行います。ただし、注射の場合は投与日数という考え方はありませんので注射の投与法が問題になるだけです。