

ビュー

目次

1. はじめに	1
2. ビューの作成と利用	1
3. ビューの上書きと削除	5

1. はじめに

複雑な SELECT 文を作成したときに、その SQL 文を保存しておきたくすることがあります。そのようなときに利用するのがビューです。ビューは、任意の SELECT 文の結果を仮想的な表として保存する機能です。ビューを利用すると、以後は複雑な SELECT 文を毎回作成する必要はなく、ビューに対する単純な SELECT 文で同様の結果を得ることが出来ます。

2. ビューの作成と利用

1 ビュー作成の基本

ビューを作成する構文は以下の通りです。

(構文) ビュー作成

```
CREATE VIEW <ビュー名>  
AS SELECT <列名> FROM <表名> [WHERE、JOIN 等];
```

ビューもデータベースオブジェクトですので CREATE 文で作成します。構文を見ると SELECT 文に対して名前を付けてビューを作成しているのが直感的に分かると思います。

SELECT 文には、WHERE 句や JOIN 句、GROUP BY 句など SELECT 文で使用出来るものはほとんどが使用可能です。関数や式も使用可能です。

それではビューを作成してみましょう。今回は、「集合演算」にて学んだ 3 つのテーブルを内部結合させる SELECT 文をビューとして作成してみます。

ビューを作成する前に、まずは、SELECT 文を確認してみましょう。「集合演算」にて使用した SELECT 文は以下のようなものでした。

(サンプルコード) 3 テーブルの内部結合

```
SELECT SI.shop_id, SI.shop_name, SI.item_id, I.item_name, I.sel_price,  
       SI.quantity, StI.stock  
FROM ShopItem SI INNER JOIN Item I ON SI.item_id = I.item_id  
INNER JOIN StockItem StI ON SI.item_id = StI.item_id  
WHERE StI.house_id = 'S001';
```

(実行結果)

SHOP_ID	SHOP_NAME	ITEM_ID	ITEM_NAME	SEL_PRICE	QUANTITY	STOCK
000D	福岡	0001	シャツ	1000	100	0
000A	東京	0001	シャツ	1000	30	0
000B	仙台	0002	ホッチキス	500	30	120
000A	東京	0002	ホッチキス	500	50	120
000C	大阪	0003	セーター	4000	20	200
000B	仙台	0003	セーター	4000	120	200
000A	東京	0003	セーター	4000	15	200
000C	大阪	0004	包丁	3000	50	3
000B	仙台	0004	包丁	3000	20	3
000C	大阪	0006	フォーク		90	99
000B	仙台	0006	フォーク		10	99
000C	大阪	0007	スプーン	790	70	999
000B	仙台	0007	スプーン	790	40	999

13 行選択されました

この SELECT 文をビューとして作成する CREATE 文は以下の通りです。

(サンプルコード) 3 テーブルの内部結合をビュー化

```
CREATE VIEW ItemView AS
SELECT SI.shop_id, SI.shop_name, SI.item_id, I.item_name, I.sel_price,
       SI.quantity, StI.stock
FROM ShopItem SI INNER JOIN Item I ON SI.item_id = I.item_id
INNER JOIN StockItem StI ON SI.item_id = StI.item_id
WHERE StI.house_id = 'S001';
```

これでビューが作成されました。それではこちらのビューに対して SELECT 文を発行してみましょう。SELECT 時にビューはテーブルと同じように扱えます。

(サンプルコード) ビューの中身を表示

```
SELECT * FROM ItemView;
```

(実行結果)

SHOP_ID	SHOP_NAME	ITEM_ID	ITEM_NAME	SEL_PRICE	QUANTITY	STOCK

000D	福岡	0001	シャツ	1000	100	0
000A	東京	0001	シャツ	1000	30	0
000B	仙台	0002	ホッチキス	500	30	120
000A	東京	0002	ホッチキス	500	50	120
000C	大阪	0003	セーター	4000	20	200
000B	仙台	0003	セーター	4000	120	200
000A	東京	0003	セーター	4000	15	200
000C	大阪	0004	包丁	3000	50	3
000B	仙台	0004	包丁	3000	20	3
000C	大阪	0006	フォーク		90	99
000B	仙台	0006	フォーク		10	99
000C	大阪	0007	スプーン	790	70	999
000B	仙台	0007	スプーン	790	40	999
13 行選択されました						

このようにビューを用いることで非常にシンプルな SELECT 文で同じ結果を得られるようになりました。

2 ビュー利用時の注意点

大変便利なビューですが、いくつか注意しなければいけない点があります。

●関数や式を使ったビュー

関数や式を使った SELECT 文に対してビューを設定する場合、単純な SELECT 文に「CREATE VIEW ビュー名」をつけただけではエラーになってしまいます。このような場合には、関数や式を使っている列に対して別名を付けます。

●ビューに対する更新

ビューに対する INSERT、DELETE、UPDATE といった更新系 SQL（データを更新する SQL）の使用はかなり厳しい制限付きではありますが、可能な場合があります。

標準 SQL では、「ビュー定義の SELECT 文において、いくつかの条件を満たしている場合、ビューに対する更新が可能」と定められています。代表的な条件を上げると、次の通りです。

- ① SELECT 句に DISTINCT が含まれていない
- ② FROM 句に含まれるテーブルが 1 つだけである
- ③ GROUP BY 句を使用していない
- ④ HAVING 句を使用していない

以上の条件が満たされている場合は、ビューに対して更新が可能です。

3. ビューの上書きと削除

1 ビューの上書き

「CREATE VIEW」を実行するとき、作成しようとする名前のビューが既に存在する場合はエラーになります。

このような場合には「CREATE OR REPLACE VIEW～」のように OR REPLACE をつけることにより、上書きすることが出来ます。

(構文) ビュー作成

```
CREATE OR REPLACE VIEW <ビュー名>  
AS SELECT <列名> FROM <表名> [WHERE、JOIN 等] ;
```

2 ビューの削除

ビューの作成が CREATE 文であることから予想出来ますが、ビューの削除は DROP 文を使用します。

(構文) ビュー削除

```
DROP VIEW <ビュー名> ;
```