

301仮想表をつくる(1)

仮想表をつくる(1)

view

リレーショナル・データベースでは、view（ビュー）と呼ばれる仮想的なテーブルがよく使われる。「仮想的」というのは、viewは、データベースの中では普通のテーブルの様な「実体」を持っておらず、それを作り出す定義のみが、システムに記憶されているだけだからである。それにも関わらず、SQLの中で、viewはほとんどテーブルと区別なく扱われる。例えば、select文のfrom句には、テーブル名とview名との混在が可能である。

viewをつくる

まず、実際にviewをつくる操作を見てみよう。

viewをつくる命令の一般形は、次のようなものである。

```
create view view名 [( 項目名リスト ... )] as select 項目名リスト from テーブル／view 名リスト where 選択条件
```

要するに、create view は、selectの出力が、再び、テーブルとなることを利用して、select文で、新しい仮想的なテーブルviewを定義しているのである。

次の例では、テーブル「従業員」をもとに、「v_従業員横浜市在住」という名前のviewがつくられている。

例 1: テーブルから新しいviewをつくる

```
create view v_従業員横浜市在住1
as
select *
from 従業員
where 住所 like '%横浜市%'
```

viewとテーブルの違い

もう一度、テーブルとviewの違いを考えてみよう。今つくられた「v_従業員横浜市在住1」は、viewである。以前に、以下の様にinsertで行を挿入した、「従業員横浜市在住1」は、テーブルである。

```
insert into 従業員横浜市在住1
select *
from 従業員
where 住所 like '%横浜市%'
```

この二つのテーブルとviewは、ともに、テーブル「従業員」から、同じselect文でつくられており、内容的には一致している。だから、次の二つの検索は、同じ結果をもたらす。

```
select *  
from 従業員横浜市在住1    /* テーブルから検索 */  
where 年齢 < 30
```

```
select *  
from v_従業員横浜市在住1    /* view から検索 */  
where 年齢 < 30
```

テーブル「従業員横浜市在住」の場合、もとのテーブル「従業員」と全く同じ データのコピーを、テーブルの中にかかえることになる。これは、ある意味で資源の ムダ遣いである。これに対して、view 「v_従業員横浜市在住」の場合には、それ自身としては、データをかかえていない。ただ、システムは、このviewの定義を知っているのである。先ほどのviewからの検索の例では、システムは、このviewの定義に基づいて、この select文を、次のようなselect文に置き換えて、実行する。

```
select *  
from 従業員  
where 住所 like '%横浜市%'  
and 年齢 < 30
```