

Table of Contents

1. Creating a Simple View.....	1
2. Using the View	2
3. A more complex view.....	2

We will look at views in more detail soon, but the assignment for this unit is easier with a view. A view is a select query that is given a name and the view definition is stored in the database. You can then use the view as a data source in a From clause. For the assignment you can just think of the view as a table expression and you can

- select columns from the view
- use filters against the columns in the view
- join the view to base tables using the same syntax you always use

The view we will use in the assignment and also in the subqueries demo contains a join of several tables and the purpose of the view is to hide that complexity from the queries we want to run.

1. Creating a Simple View

Demo 01: This is a very simple view that just exposes three columns for the products table for some of the rows. Note that the view is defined within a database.

```
drop view if exists a_prd.Prod_HW_APL;

Create view a_prd.Prod_HW_APL as (
Select
prod_id , prod_name, prod_list_price
From a_prd.products
Where catg_id in ('APL', 'HW')
);
```

Demo 02: -- select data from the view

```
Select *
From a_prd.Prod_HW_APL;
```

prod_id	prod_name	prod_list_price
1000	Hand Mixer	125.00
1070	Iron	25.50
1071	Iron	25.50
1072	Iron	25.50
1080	Cornpopper	25.00
1090	Gas grill	149.99
1100	Blender	49.99
1110	Pancake griddle	49.99
1120	Washer	549.99
1125	Dryer	500.00
1126	WasherDryer	850.00
1130	Mini Freezer	149.99
1160	Mixer Deluxe	149.99
4569	Mini Dryer	349.95
4575	Electric can opener	49.95

2. Using the View

Demo 03: -- -- select data from the view with a filter

```
Select *
From a_prd.Prod_HW_APL
Where prod_list_price > 100;
```

prod_id	prod_name	prod_list_price
1000	Hand Mixer	125.00
1090	Gas grill	149.99
1120	Washer	549.99
1125	Dryer	500.00
1126	WasherDryer	850.00
1130	Mini Freezer	149.99
1160	Mixer Deluxe	149.99
4569	Mini Dryer	349.95

Demo 04: -- -- select data from the view with a join to a base table

```
Select PR.prod_id, quantity_ordered * quoted_price as ExtPrice , ord_id
From a_prd.Prod_HW_APL PR
Join a_oe.order_details OD on PR.prod_id = OD.prod_id
Where ord_id between 110 and 115;
```

prod_id	ExtPrice	ord_id
1090	149.99	110
1130	149.99	110
1110	99.98	112
1080	22.50	113
1130	625.00	114
1000	200.00	115
1120	1900.00	115
1080	25.00	115
1100	180.00	115

3. A more complex view

The next view is more complex, including several joins and filters and renaming the column names

Demo 05: a more complex view

```
drop view if exists a_oe.cust_orders;

Create view a_oe.cust_orders as (
Select
    OH.ord_id as Invoice
  , OH.ord_date as OrderDate
  , OH.cust_id as CustID
  , PR.catg_id as Category
  , OD.prod_id as ItemPurchased
From a_oe.order_headers OH
Join a_oe.order_details OD on OH.ord_id= OD.ord_id
```

```

Join a_prd.products      PR on OD.prod_id = PR.prod_id
Where OD.quoted_price > 0
and OD.quantity_ordered > 0
);

```

Demo 06: When you use the second view, refer to the column names defined in the view. Note that if an order has more than one detail line, the view returns more than one row

```

Select Invoice, ItemPurchased
From a_oe.cust_orders
Where month(OrderDate) = 6;
+-----+-----+
| Invoice | ItemPurchased |
+-----+-----+
|      301 |           1100 |
|      302 |           1140 |
|      302 |           1040 |
|      303 |           1000 |
|      306 |           1120 |
|      306 |           1125 |
|      307 |           1120 |
|      307 |           1125 |
|      312 |           1040 |
|      312 |           1050 |
|      312 |           1060 |
|      312 |           1060 |
|      313 |           1000 |
|      324 |           4576 |
|      378 |           1120 |
|      378 |           1125 |
|      390 |           1010 |
|      395 |           1010 |
|      540 |           1080 |
|      540 |           1110 |
|      540 |           1152 |
+-----+-----+
21 rows in set (0.09 sec)

```