



VIEWS

METIS



What are SQL views?

SQL object that provides snapshot of information from one or more table.
Essentially, a view is a named query and/or join that provides a virtual table of data.

What is possible with a view?

Filtering, sorting, math/string/date functions just like a table.
Best practice is to NOT join views.

Notes: A view does not create or store more data, merely a metadata object!
A view helps store SELECT logic for quick access or sharing with other users.

Creating a View



Question:

How much markup does the company charge for each item in inventory?

Inventory Table, Top 5

id	name	vendor_unit_price
1	tiger t-shirt	4.2500
2	giraffe-print bag	24.9900
3	elephant tie	13.1900
4	zebra-striped pants	16.8800
6	leopard-print scarf	8.5500

Item Details Table, Top 5

item_id	name	sales_price
1	tiger t-shirt	9.9900
2	giraffe-print bag	49.9900
3	elephant tie	35.4900
4	zebra-striped pants	30.9900
5	peacock feather hat	34.9900

Creating a View



Question:

How much markup does the company charge for each item in inventory?

SELECT
same as
before

```
CREATE VIEW v_markup
AS
SELECT inv.id, inv.name,
       (d.sales_price - inv.vendor_unit_price) AS markup
FROM   inventory AS inv
JOIN   item_details AS d
ON     inv.id = d.item_id;
```

TIP: Name the view with a leading 'v' so that it is not confused with a table.

Creating a View



Question:

How much markup does the company charge for each item in inventory?

```
SELECT * FROM v_markup;
```

id	name	markup
1	tiger t-shirt	5.7400
2	giraffe-print bag	25.0000
3	elephant tie	22.3000
4	zebra-striped pants	14.1100
6	leopard-print scarf	5.9400
8	gazelle lamp	41.1400
9	bedding set, tiger icons	38.0000

Creating a View



Question:

Which items does the company mark up \$20 or more (ordered by the highest markup)?

```
SELECT UPPER(name), markup
FROM v_markup
WHERE markup > 20
ORDER BY markup DESC;
```

UNAME	markup
GAZELLE LAMP	41.1400
BEDDING SET, TIGER ICONS	38.0000
GIRAFFE-PRINT BAG	25.0000
ELEPHANT TIE	22.3000

Note: Views allow functions, filtering, and ordering just like tables.


Altering or Dropping Views



Altering a view object

**Must know full
SELECT
statement to
alter view**

```
ALTER VIEW v_markup
AS
SELECT inv.id, inv.name, inv.vendor_unit_price
       (d.sales_price - inv.vendor_unit_price) AS markup
FROM inventory AS inv
JOIN item_details AS d
ON inv.id = d.item_id;
```



Deleting a view object

```
DROP VIEW v_markup;
```