Views

A view is a SELECT statement that's stored as part of the database. That means it can be used not only by SQL programmers but by users and application programs that have access to the database. This provides some distinct advantages over using tables directly. You can think of a view as a virtual table that consists only of the rows and columns specified in it CREATE VIEW statement.

- A view is deleted with the **`DROP VIEW`** statement. Or ALTER VIEW
  - DROP VIEW *view_name*;
  - ALTER VIEW *view_name*;

- A view can be updated with the **`CREATE OR REPLACE VIEW`** statement.
  - CREATE OR REPLACE VIEW *view_name* AS
    SELECT *column1*, *column2*, ...
    FROM *table_name*
    WHERE *condition*;

```
-- Create VIEW statement for a view name VendorMin
-- From this VIEW show all vendors from the state of California (CA)

CREATE VIEW VendorMin AS
SELECT VendorName, VendorState, VendorPhone
FROM Vendors;

SELECT * FROM VendorMin
WHERE VendorState = 'CA'
ORDER BY VendorName;
```

```
-- Create VIEW statement that creates a view of vendors that have invoices, name it
VendorShortList.
-- From this VIEW show all vendors from the state of California (CA)

CREATE VIEW VendorShortList AS
SELECT VendorName, VendorContactLName, VendorContactFName,VendorPhone
FROM Vendors
WHERE VendorID IN(SELECT VendorID FROM Invoices);

SELECT * FROM VendorShortList
```

```
-- Create VIEW statement that uses a JOIN. Name the VIEW VendorAccountDescription
--Return three columns Vendors; VendorName, DefaultAccountNo. GLAccounts,
AccountDescription
--the result set should have one row for each vendor, with the account number and account
description for that vendor's default account number.

CREATE VIEW VendorAccountDescription AS
SELECT V.VendorName, V.DefaultAccountNo, GL.AccountDescription
FROM Vendors AS V
JOIN GLAccounts AS GL
ON V.DefaultAccountNo = GL.AccountNo
SELECT * FROM VendorAccountDescription
```

An Updateable VIEW is one that can be used in **INSERT, UPDATE, or DELETE** statement to modify the contents of a base table that the view refers to. If a view is not updatable, it is called a **Read-only-VIEW.**

```
-- A CREATE VIEW Statement that creates an updatable view

CREATE VIEW InvoiceCredit
AS
SELECT InvoiceNumber, InvoiceDate,InvoiceTotal, PaymentTotal, CreditTotal
FROM Invoices
WHERE InvoiceTotal - PaymentTotal -CreditTotal >0;

-- An UPDATE statement that updates a view

UPDATE InvoiceCredit
SET CreditTotal = CreditTotal + 200
WHERE InvoiceTotal -PaymentTotal - CreditTotal >=200

------ 5 rows affected ----------------------------
```

```
--Write a CREATE VIEW statement that defines a view named InvoiceBasic the returns three
columns
--VendorName, InvoiceNumber, and Invoicetotal.
--Write a SELECT statement that returns all of the columns in the view, sorted by
VendorName,
--where the first letter of the vendor name is N,O or P

CREATE VIEW InvoiceBasic
AS
SELECT VendorName, InvoiceNumber, InvoiceTotal
FROM  Vendors JOIN Invoices
  ON Vendors.VendorID = Invoices.VendorID;


SELECT *
FROM InvoiceBasic
WHERE VendorName LIKE '[N-P]%'
ORDER BY VendorName;
```

```
--Create an updatable view named VendorAddress that returns the VendorID
--both address columns, and the city, state, and zipcode columns for each vendor.
--Write a SELECT statement to examine the result set where VendorID=4


CREATE VIEW VendorAddress
AS
SELECT VendorID, VendorAddress1, VendorAddress2, VendorCity, VendorState, VendorZipCode
FROM Vendors;

SELECT *
FROM VendorAddress
WHERE VendorID = 4;
```