DB

VIEWS

🎯 OBJECTIVES FOR TODAY 🎯

- VIEWS

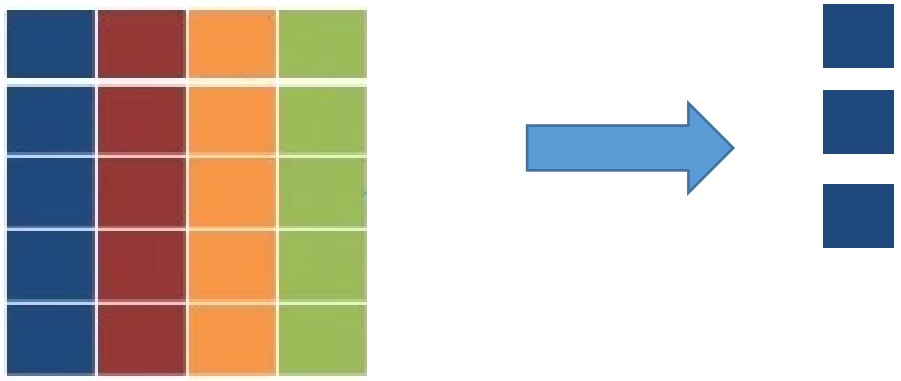
https://www.w3schools.com/sql/sql_view.asp

https://www.datacamp.com/community/tutorials/views-in-sql

The following SQL selects all customers from Brazil:

```
SELECT CustomerName
FROM Customers
WHERE Country = "Brazil";
```

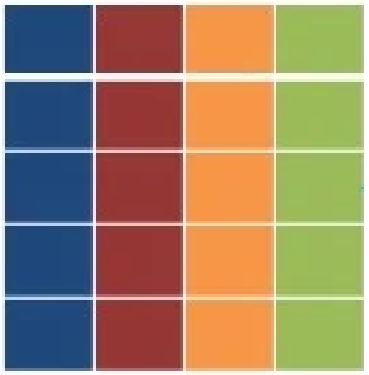
Customer



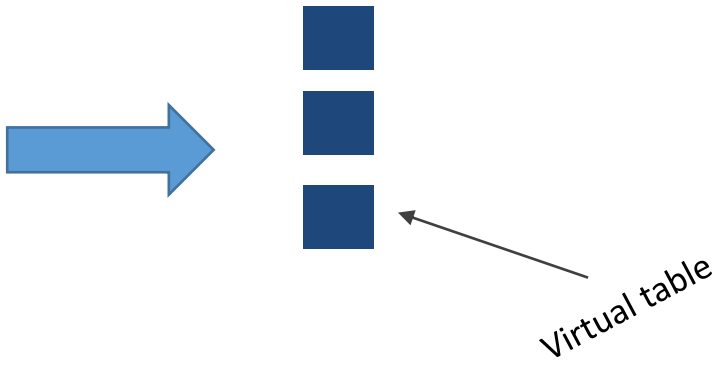
The following SQL **creates a view** that selects all customers from Brazil:

```
CREATE VIEW BrazilCustomers AS
SELECT CustomerName
FROM Customers
WHERE Country = "Brazil";
```

Customer

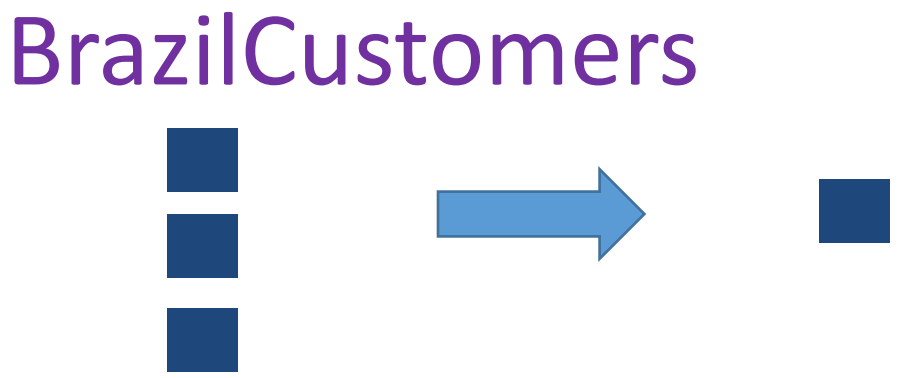


BrazilCustomers



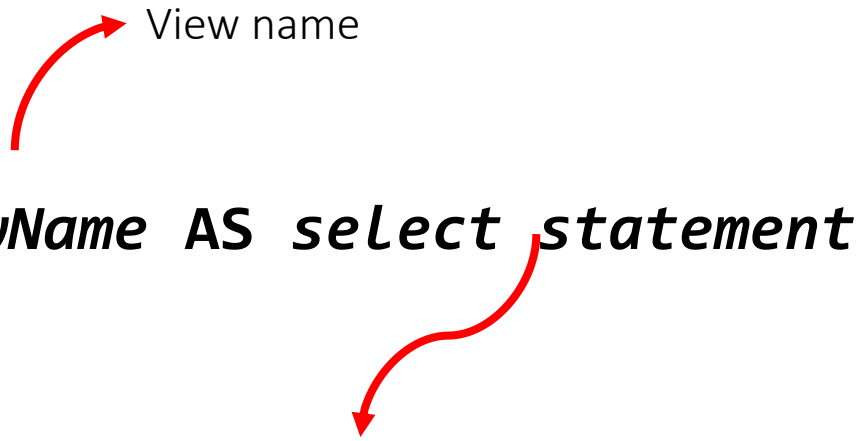
We can then query the view as follows:

```
SELECT * FROM [BrazilCustomers]
WHERE BrazilCustomers.CustomerName == “ronan” ;
```



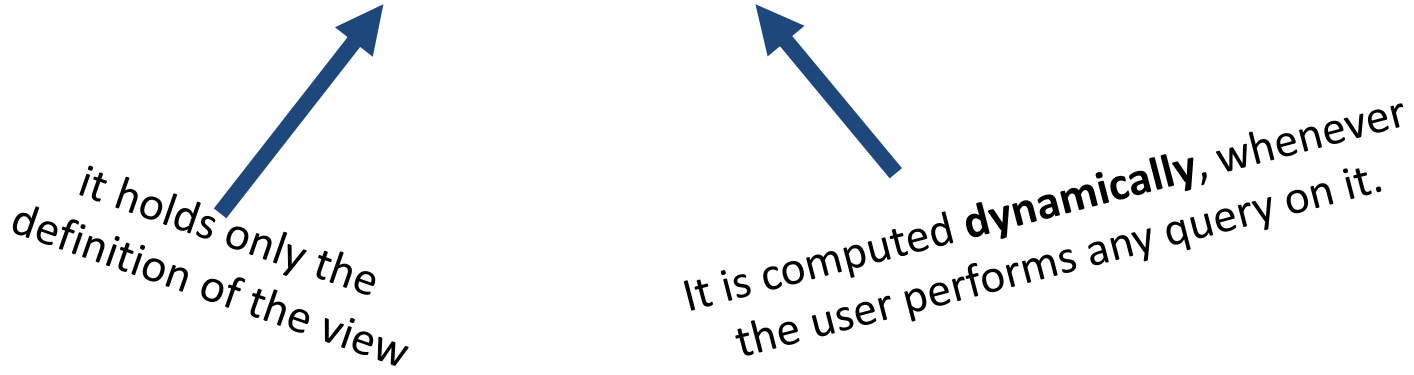
What is a view ?

A view is a **virtual** **table** based on the result of an SQL statement.



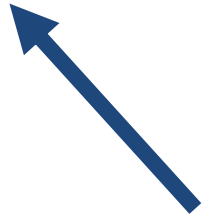
What is a view ?

- ✓ You can select data from multiple tables
- ✓ It does not hold the data



Why do we need to create views ?

- ✓ Simplify the **complex** SQL queries.
- ✓ Provide **restriction** to users from accessing sensitive data.



For example, a user has permission to access particular columns of data rather than the whole table.

Example of views

Employee

EmployeeID	Ename	DeptID	Salary
1001	John	2	4000
1002	Anna	1	3500
1003	James	1	2500
1004	David	2	5000
1005	Mark	2	3000
1006	Steve	3	4500
1007	Alice	3	3500

```
CREATE VIEW emp_view AS
SELECT EmployeeID, Ename
FROM Employee
WHERE DeptID=2;
```

Creating View by filtering records using WHERE clause

emp_view

EmployeeID	Ename	DeptID	Salary
1001	John	2	4000
1004	David	2	5000
1005	Mark	2	3000

Example of views

Employee

EmployeeID	Ename	DeptID	Salary
1001	John	2	4000
1002	Anna	1	3500
1003	James	1	2500
1004	David	2	5000
1005	Mark	2	3000
1006	Steve	3	4500
1007	Alice	3	3500

```
CREATE VIEW emp_view AS
SELECT DeptID, AVG(Salary)
FROM Employee
GROUP BY DeptID;
```

Create View of grouped records on Employee table

emp_view

DeptID	AVG(Salary)
1	3000.00
2	4000.00
3	4250.00

Operations on VIEWS

CREATE

```
CREATE VIEW viewName AS
SELECT *
FROM Customers
```

REMOVE

```
DROP VIEW viewName
```

Good and bad points on view

- ✓ Hide the complexity
- ✓ Allow to protect some data

- ✓ Can take time to compute
- ✓ Can create dependencies

EXERCICES !