# MySQL Views

# VIEWS

- Views are stored queries that when invoked produce a result set
- A view acts as a virtual table
- A view contains rows and columns, just like a real table. The fields in a view are fields from one or more real tables in the database.
- You can add SQL functions, WHERE, and JOIN statements to a view and present the data as if the data were coming from one single table.
- CREATE VIEW Syntax:

```
CREATE
   [OR REPLACE]
   VIEW view_name [(column_list)]
   AS select_statement
   [WITH [CASCADED | LOCAL] CHECK OPTION]
```

# VIEWS

**Why views ?**

- Views can have column names and expressions.
- You can use any clauses in views.
- Views can be used in INSERT/UPDATE/DELETE.
- Views can contain expressions in the select list.
- Views can be views of views.
- A view doesn’t take any extra space on disk.
- Views can help us to protect databases:
  - We can hide information of tables
  - We can give different grants on views

# VIEWS

**Updatable views**

- Some views are **updatable**. That is, you can use them in statements such as UPDATE, DELETE, or INSERT to update the contents of the underlying table.

- For a view to be updatable, there must be a one-to-one relationship between the rows in the view and the rows in the underlying table.

# VIEWS

A view is not updatable if it contains any of the following:

- Aggregate functions (SUM(), MIN(), MAX(), COUNT(), and so forth)
- DISTINCT
- GROUP BY
- HAVING
- UNION or UNION ALL
- Subquery in the select list
- Certain joins
- Reference to nonupdatable view in the FROM clause
- Subquery in the WHERE clause that refers to a table in the FROM clause
- Refers only to literal values (in this case, there is no underlying table to update)

# VIEWS

With respect to insertability (being updatable with INSERT statements), an updatable view is insertable if it also satisfies these additional requirements for the view columns:

- There must be no duplicate view column names.
- The view must contain all columns in the base table that do not have a default value.
- The view columns must be simple column references. They must not be expressions, such as these:

```
3.14159
col1 + 3
UPPER(col2)
col3 / col4
(subquery)
```

# VIEWS

- The **WITH CHECK OPTION** clause can be given for an updatable view to prevent inserts to rows for which the WHERE clause in the *select_statement* is not true. It also prevents updates to rows for which the WHERE clause is true but the update would cause it to be not true (in other words, it prevents visible rows from being updated to nonvisible rows).

- In a **WITH CHECK OPTION** clause for an updatable view, the LOCAL and CASCADED keywords determine the scope of check testing when the view is defined in terms of another view. When neither keyword is given, the default is CASCADED.

# VIEWS

**EXAMPLES**:

```
CREATE OR REPLACE VIEW myView_1 AS
SELECT id,name,surname,position,captain,team,height
FROM player;

CREATE OR REPLACE VIEW myView_2 AS
SELECT id,name,surname,salary*12 as annual_salary FROM
player;

update myView_2 set salary = 5000000 where id = 1;
-- Error Code: 1348. Column 'annual_salary' is not updatable
```

# VIEWS

**EXAMPLES**:

```
create view upd_view as select * from player where
team = 6;


insert into upd_view (id,name,team) values
(30,'Macià',2); -- it's inserted into table, but we
can not see on the view



create view upd_view2 as select * from player where
team = 6 with check option;

insert into upd_view2 (id,name,team) values
(31,'Macià2',3); -- Error Code: 1369. CHECK OPTION
failed 'liga.upd_view2'
```

END