Database Design & Applications

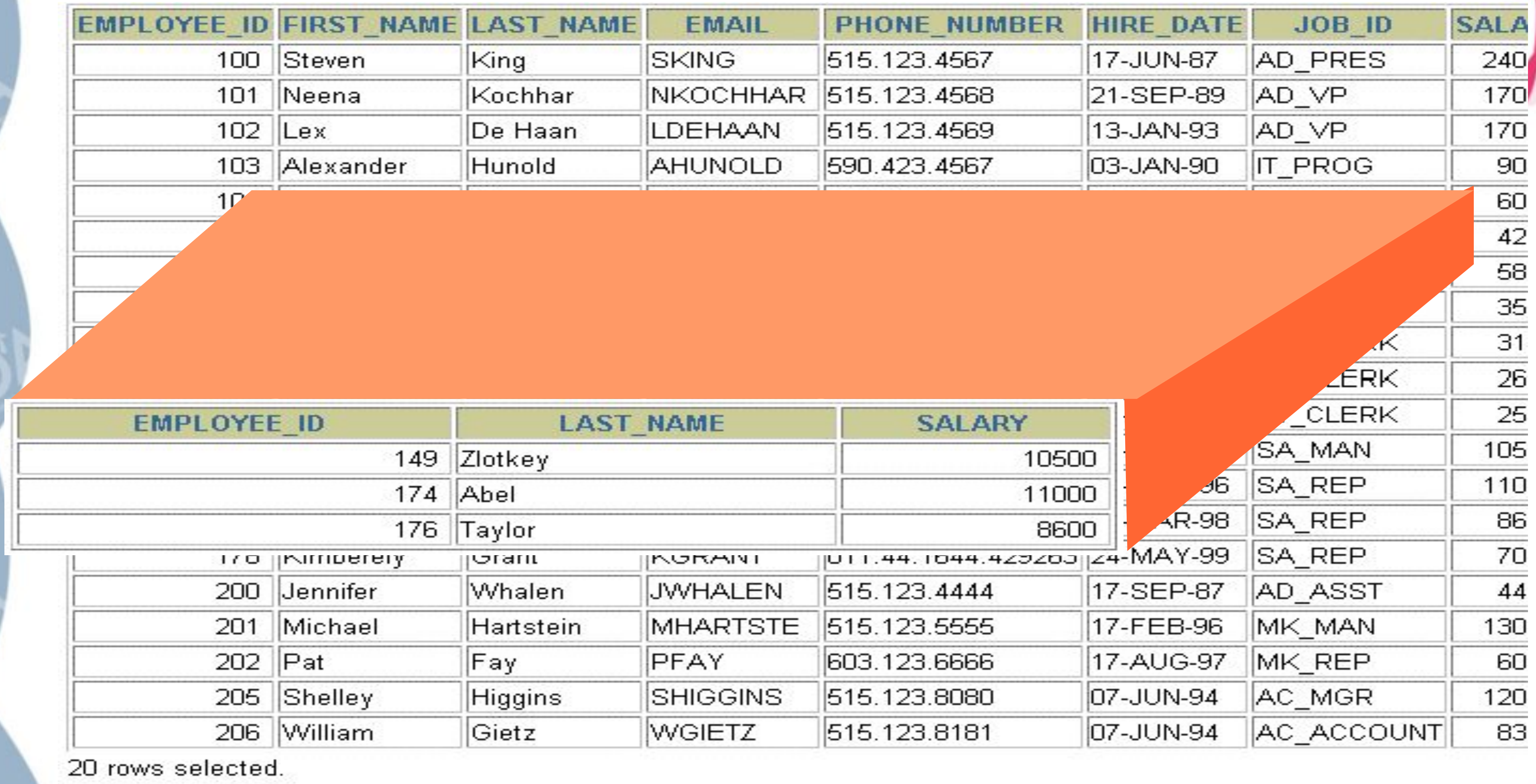
The Database Language - Create View

Objective

- Creating View
- Updatable View
- WITH CHECK Option
- ALTER View
- DROP View

What is a View?

EMPLOYEES Table:



EMPLOYEE_ID	FIRST_NAME	LAST_NAME	EMAIL	PHONE_NUMBER	HIRE_DATE	JOB_ID	SALA
100	Steven	King	SKING	515.123.4567	17-JUN-87	AD_PRES	240
101	Neena	Kochhar	NKOCHHAR	515.123.4568	21-SEP-89	AD_VP	170
102	Lex	De Haan	LDEHAAN	515.123.4569	13-JAN-93	AD_VP	170
103	Alexander	Hunold	AHUNOLD	590.423.4567	03-JAN-90	IT_PROG	90
10							60
							42
							58
							35
						K	31
						ERK	26
						_CLERK	25
						SA_MAN	105
					96	SA_REP	110
					AR-98	SA_REP	86
178	Kimberely	Grant	KGRANT	011.44.1644.429263	24-MAY-99	SA_REP	70
200	Jennifer	Whalen	JWHALEN	515.123.4444	17-SEP-87	AD_ASST	44
201	Michael	Hartstein	MHARTSTE	515.123.5555	17-FEB-96	MK_MAN	130
202	Pat	Fay	PFAY	603.123.6666	17-AUG-97	MK_REP	60
205	Shelley	Higgins	SHIGGINS	515.123.8080	07-JUN-94	AC_MGR	120
206	William	Gietz	WGIETZ	515.123.8181	07-JUN-94	AC_ACCOUNT	83

20 rows selected.

EMPLOYEE_ID	LAST_NAME	SALARY
149	Zlotkey	10500
174	Abel	11000
176	Taylor	8600

Why Use Views?

- To restrict data access
- To make complex queries easy
- To implement ROW and COLUMN level security.
- To present different views of the same data

Updatable Views

Feature	Updatable Views
Number of tables	One
Contain functions	No
Contain groups of data	No
DML operations through a view	Yes

Creating a View

- You embed a subquery within the CREATE VIEW statement.

```
CREATE  VIEW view [(alias[, alias]...)]
 AS subquery
[WITH CHECK OPTION];
```

- The subquery can contain complex SELECT syntax.

Creating a View

- Create a view EMPVU80, that contains details of employees in department 30.

```
CREATE VIEW   empvu30
 AS SELECT    employee_id, last_name, salary  FROM employees
    WHERE department_id = 30;
```

Creating a View

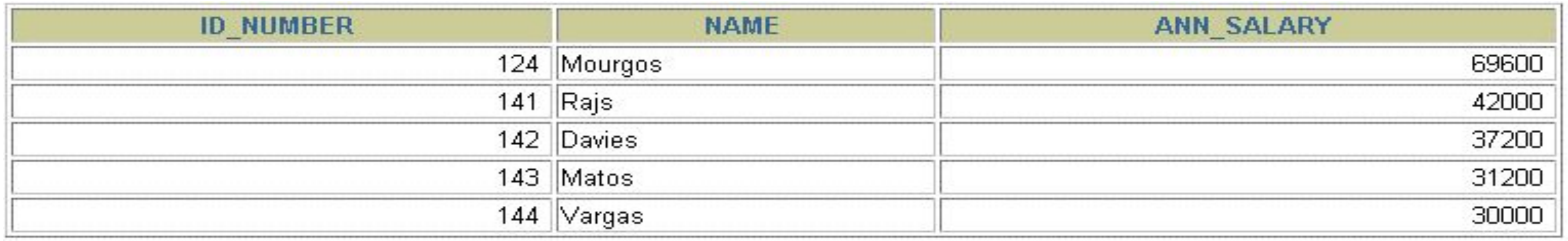
- Create a view by using column aliases in the subquery.

```
CREATE VIEW  salvu50
 AS SELECT  employee_id ID_NUMBER, last_name NAME,
            salary*12 ANN_SALARY
    FROM  employees
    WHERE department_id = 50;
```

- Select the columns from this view by the given alias names.

Retrieving Data from a View

```
SELECT *
FROM  salvu50;
```



ID_NUMBER	NAME	ANN_SALARY
124	Mourgos	69600
141	Rajs	42000
142	Davies	37200
143	Matos	31200
144	Vargas	30000

Creating a Complex View

Create a complex view that contains group functions to display values from two tables.

```
CREATE VIEW dept_sum_vu
  (name, minsal, maxsal, avgsal)
AS SELECT     d.department_name, MIN(e.salary),
              MAX(e.salary),AVG(e.salary)
   FROM   employees e, departments d
   WHERE  e.department_id = d.department_id  GROUP BY
   d.department_name;
```

Non-Updatable View

```
CREATE VIEW EMP_DEPT_VIEW
AS
SELECT employee_id, last_name, salary, dname
from employee
join
department
on employee.department_id=department.department_id;
```

```
Update EMP_DEPT_VIEW
SET salary = salary *.01, dname = 'IT'
WHERE Salary <5000;
```

```
Error:  View or function 'EMP_DEPT_VIEW' is not updatable because the
modification affects multiple base tables.
```

Updatable View

```
insert into emp_dept_view(employee_id, last_name,
salary, dname)
values (111,’Roger',10000,’IT’);
```

```
Error: View or function 'EMP_DEPT_VIEW' is not updatable because the
modification affects multiple base tables.
```

- An UPDATE OR INSERT statement against a view can only effect one target table.

Updatable View

- You can perform DML operations on updatable views.
- You cannot perform DML operations if the view contains the following:
  - Group functions
  - A GROUP BY clause
  - The DISTINCT keyword
  - Columns defined by expressions
  - NOT NULL columns in the base tables that are not selected by the view

WITH CHECK Option

- You can ensure that DML operations performed on the view stay within the domain of the view by using the WITH CHECK OPTION clause.

```
CREATE OR REPLACE VIEW empvu20  AS SELECT  *
   FROM   employees
   WHERE  department_id = 20
                                           ;
    WITH CHECK OPTION
```

- Any attempt to change the department number for any row in the view fails because it violates the WITH CHECK OPTION constraint.

Changing a View

Syntax:

```
ALTER VIEW view_name
  AS
New_Select_Statement;
```

```
ALTER VIEW EMP_DEPT_VIEW
AS
SELECT employee_id, last_name, salary,hire_date,dname
from employee
join
department
on employee.department_id=department.department_id;
```

Removing a View

- You can remove a view without losing data because a view is based on underlying tables in the database.

```
DROP VIEW view_name

DROP VIEW EMP_DEPT_VIEW
```

THANK YOU!