

- Group By clause
- Distinct keyword
- Columns contain by expressions
- NOT NULL columns in the base table that are not selected by the view

**Example:** (Using the WITH CHECK OPTION clause)

```
CREATE OR REPLACE VIEW empvu20
AS SELECT *
FROM employees
WHERE department_id=20
WITH CHECK OPTION CONSTRAINT empvu20_ck;
```

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

**Example** – (Execute this and note the error)

```
UPDATE empvu20 SET department_id=10 WHERE employee_id=201;
```

### **Denying DML operations**

Use of WITH READ ONLY option.

Any attempt to perform a DML on any row in the view results in an oracle server error.

**Try this code:**

```
CREATE OR REPLACE VIEW empvu10(employee_number, employee_name, job_title)
AS SELECT employee_id, last_name, job_id
FROM employees
WHERE department_id=10
WITH READ ONLY;
```

### **Find the Solution for the following:**

1. Create a view called EMPLOYEE\_VU based on the employee numbers, employee names and department numbers from the EMPLOYEES table. Change the heading for the employee name to EMPLOYEE.

*create view employee-vu as select employee-id, last-name  
as employee, department-id from employees;*

2. Display the contents of the EMPLOYEES\_VU view.

*select \* from employee-vu;*

3. Select the view name and text from the USER\_VIEWS data dictionary views.

```
select view_name, text from user_views where  
view_name = 'employee-vu';
```

4. Using your EMPLOYEES\_VU view, enter a query to display all employees names and department.

```
select employee, department_id from employee-vu;
```

5. Create a view named DEPT50 that contains the employee number, employee last names and department numbers for all employees in department 50. Label the view columns EMPNO, EMPLOYEE and DEPTNO. Do not allow an employee to be reassigned to another department through the view.

```
create or replace view dept50 as select employee_id as  
empno, last_name as employee, department_id as deptno  
from employees where department_id = 50 with check option;
```

6. Display the structure and contents of the DEPT50 view.

```
desc dept50;  
select * from dept50;
```

7. Attempt to reassign Matos to department 80.

```
update dept50 set deptno = 80 where employee = 'matos';
```

8. Create a view called SALARY\_VU based on the employee last names, department names, salaries, and salary grades for all employees. Use the Employees, DEPARTMENTS and JOB\_GRADE tables. Label the column Employee, Department, salary, and Grade respectively.

```
create or replace view salary-vu as select  
e.last_name as employee, d.department_name as  
department, e.salary as salary, j.grade_level  
as grade from employee e join departments d  
on e.department_id = d.department_id join  
job-grades j on e.salary between j.lowest_sal  
and j.highest_sal;
```