

- 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_ek;
```

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, first-name || ' ' || last name as Employee
 department-id
 From employees

2. Display the contents of the EMPLOYEES_VU view.

Select * from EMPLOYEES_VU

3. Select the view name and text from the USER_VIEWS data dictionary views.

```
Select view_name, text  
from user_views
```

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

```
Select employees, 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 view dept50 As  
Select  
employee_id As empno, lastname as employee,  
department_id As deptno
```

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

```
DESC dept50;  
Select * from dept50
```

7. Attempt to reassign Matos to department 80.

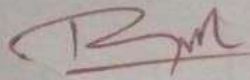
```
Update dept50  
SET dept = 80 where employee = 'Matos';
```

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 View salary_vu as  
Select
```

```
e.lastname as employee,  
d.department_name as department  
e.salary As salary  
v.grade_level As grade
```

```
from employees e  
Join departments d on e.department_id = d.department_id  
Join job-grades J on e.salary Between J.lowest_sal  
AND  
highest_sal
```


| Evaluation Procedure | Marks awarded |
|----------------------|--|
| Query(5) | 5 |
| Execution (5) | 5 |
| Viva(5) | 5 |
| Total (15) | 15 |
| Faculty Signature |  |