Autonomous University of Zacatecas

Academic Unit of Electrical Engineering

Academic Program of Software Engineering

Database Systems Laboratory II
Practice 15 -Using Views

Professor:
Aldonso Becerra Sánchez

Student:
Daniel Alejandro Morales Castillo

1 Introduction

SQL language allows the realization of projection and selection of data from several tables to satisfy the needs of reports that may be required for a programmer, developer or end user

In the theory class we saw the topic of chapter 11, which contains what views are, we knew how to create, modify, insert and differentiate between the different types of views

2 Development

Activity 1

Write the section that describes the work developed in the following activities. Read all the choices carefully because there might be more than one correct answer. Choose all the correct answers for each question. Explain the reason for your answer.

CREATE SIMPLE AND COMPLEX VIEWS

1. Which of these is a defining characteristic of a complex view, rather than a simple view? (Choose one or more correct answers.)

A. Joining two tables
B. Naming the view's columns with column aliases
C. Restricting the selection of rows with a WHERE clause
D. Performing an aggregation
E. Restricting the projection by selecting only some of the table's columns

Answer: A

One of the characteristics that we are mentioned as fundamental is that complex views have one or more tables and what is a simple view only has one table

create view v1 as select departmentid,departmentname,lastname from departments join employees using (departmentid);

select departmentname,lastname from v1 where departmentid=20;

select d.departmentname,e.lastname from departments d, employees e where d.departmentid=e.departmentid and d.departmentid=20;

The first query will be quicker than the second because (choose the best answer):**2. Consider these three statements:**

create view v1 as select departmentid,departmentname,lastname from departments join employees using (departmentid);

select departmentname,lastname from v1 where departmentid=20;

select d.departmentname,e.lastname from departments d, employees e where d.departmentid=e.departmentid and d.departmentid=20;

The first query will be quicker than the second because (choose the best answer):

A. The view has already done the work of joining the tables.
B. The view uses ISO standard join syntax, which is faster than the Oracle join syntax used in the second query.

C. The view is precompiled, so the first query requires less dynamic compilation than the second query.

D. There is no reason for the first query to be quicker.

Answer: A

because as the view did the work of the join, it should not be necessary to repeat the join process again

3. Study this view creation statement: create view dept30 as select departmentid,employeeid,lastname from employees where departmentid=30 with check option; What might make the following statement fail? (Choose the best answer.) update dept30 set departmentid=10 where employeeid=114;

A. Unless specified otherwise, views will be created as WITH READ ONLY.

B. The view is too complex to allow DML operations.

C. The WITH CHECK OPTION will reject any statement that changes the DEPARTMENTID.

D. The statement will succeed

Answer: C

With the CHEK OPTION we can restrict the data, so when we want to update the data we will be conditioned

RETRIEVE DATA FROM VIEWS

4. There is a simple view SCOTT.DEPTVIEW on the table SCOTT.DEPT. This insert fails with an error: SQL¿ insert into deptview values('SUPPORT','OXFORD'); insert into deptview values('SUPPORT','OXFORD') * ERROR at line 1: ORA-01400: cannot insert NULL into ("SCOTT"."DEPT"."DEPTNO") What might be the problem? (Choose the best answer.)

A. The INSERT violates a constraint on the detail table.
B. The INSERT violates a constraint on the view.
C. The view was created as WITH READ ONLY.
D. The view was created as WITH CHECK OPTION.

Answer: A

In this view, the columns that have the NOT in the original table are not specified NULL CONSTRAINT, the insert statement will always fail because there is values must be specified at the time of insertion

5. To add the number of columns selected by a view: (Choose the best answer.)

A. Add more columns to the underlying table.
B. Issue the alter view statement.
C. Use a correlated subquery in conjunction with the view.
D. Drop and re-create the view with references to select more columns.

Answer: D

When you want to modify a view, we can use the create or replace view to be able to insert new columns to that view, this function helps us in the event that there is already a view, tell us or give us an error and if not, it creates the view.

6. The following statement is issued against the Oracle database. Which line will produce an error? (Choose the best answer.)

A. create view EMPVIEW$_0$1
B. as select E.EMPID, E.LASTNAME, E.FIRSTNAME, A.ADDRESS
C. from EMPLOYEE E, EMPLADDRESS A
D. where E.EMPID = A.EMPID
E. with check option;
F. This statement contains no errors.

Answer: F

I believe that all available answer options are well structured, without errors

Activity 2:

Propose an answer to the following issues:

1. What is a simple view? How does it differ from a complex view? Which view allows the user to insert data into the view's underlying table? Explain.

A view is a logical representation of one or more tables. A view does not contain data. All data is derived from one or more underlying tables, in the case of the simple view it only shows the data in a single table Unlike a complex view, this type of view, apart from showing the data in many tables, can also contain functions or data groups, a simple dress would allow inserting

2. What is a complex view? What are the rules that determine when a complex view can be used to modify data in an underlying table? Explain.

A complex view all your data from many tables and can contain functions or data groups as well as this type of view it does not always allow DML operations through the view.

3. How can constraints be created and enforced on views?

In the views that we have or we have two options for this is the option of, check option that forces us to work with the data domain, and we have the one with ready only to select only one view but that is read-only as is, it cannot be modified

4. On what principle does a view constraint operate?

5. What statement is used to alter the definition of a view?

With CREATE OR REMPLACE VIEW

6. How are views dropped?

With the function DROP VIEW, the syntax is DROP VIEW $_NOMBREDEVISTA$

7. How can you create a view even if the table referenced does not exist?

8. What statement is used to recompile or revalidate an existing view definition?

Re-execute the CREATE OR REPLACE VIEW

9. What is object dependency? About views and tables.

Deleting or changing objects can affect other objects in the database such as views or procedures that depend on them and, in certain instances, can "break" the dependent object. An example can be if a View queries a table and the name of that table changes. The View won't work anymore.

Activity 3:

This exercise must be performed using HR schema

a) Problem 1. 1. The staff in the HR department wants to hide some of the data in the EMPLOYEES table. Create a view called EMPLOYEESVU based on the employee numbers, employee last names, and department numbers from the EMPLOYEES table. The heading for the employee name should be EMPLOYEE.

```
--act3
--a)

CREATE VIEW EMPLOYEES_VU
AS SELECT EMPLOYEE_ID, LAST_NAME AS "EMPLOYEE", DEPARTMENT_ID
FROM EMPLOYEES;
```

Salida de Script

Tarea terminada en 0.17 segundos

```
View EMPLOYEES_VU creado.
```

2. Confirm that the view works. Display the contents of the EMPLOYEESVU view.

```
SELECT * FROM EMPLOYEES_VU;
```

Salida de Script | Resultado de la Consulta

SQL | Se han recuperado 50 filas en 0.005 segundos

	EMPLOYEE_ID	EMPLOYEE	DEPARTMENT_ID
1	100	King	90
2	101	Kochhar	90
3	102	De Haan	90
4	103	Hunold	60
5	104	Ernst	60
6	105	Austin	60
7	106	Pataballa	60
8	107	Lorentz	60
9	108	Greenberg	100
10	109	Faviet	100
11	110	Chen	100
12	111	Sciarra	100
13	112	Urman	100
14	113	Popp	100
15	114	Raphaely	30
16	115	Khoo	30
17	116	Baida	30
18	117	Tobias	30
19	118	Himuro	30
20	119	Colmenares	30
21	120	Weiss	50
22	121	Fripp	50
23	122	Kaufling	50
24	123	Vollman	50
25	124	Mourgos	50
26	125	Nayer	50
27	126	Mikkilineni	50
28	127	Landry	50

3. Using your EMPLOYEESVU view, write a query for the HR department to display all employee names and department numbers

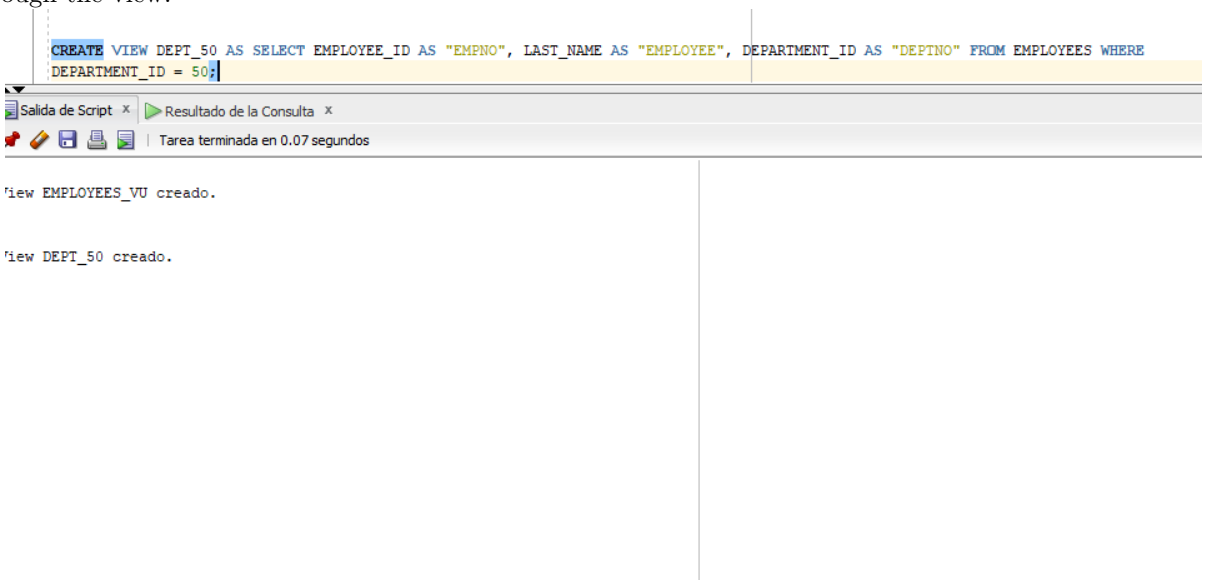
```
SELECT * FROM EMPLOYEES_VU;
```

Salida de Script | Resultado de la Consulta

SQL | Se han recuperado 50 filas en 0.005 segundos

	EMPLOYEE_ID	EMPLOYEE	DEPARTMENT_ID
1	100	King	90
2	101	Kochhar	90
3	102	De Haan	90
4	103	Hunold	60
5	104	Ernst	60
6	105	Austin	60
7	106	Pataballa	60
8	107	Lorentz	60
9	108	Greenberg	100
10	109	Faviet	100
11	110	Chen	100
12	111	Sciarra	100
13	112	Urman	100
14	113	Popp	100
15	114	Raphaely	30
16	115	Khoo	30
17	116	Baida	30
18	117	Tobias	30
19	118	Himuro	30
20	119	Colmenares	30
21	120	Weiss	50
22	121	Fripp	50
23	122	Kaufling	50
24	123	Vollman	50
25	124	Mourgos	50
26	125	Nayer	50
27	126	Mikkilineni	50
28	127	Landry	50

4. Department 50 needs access to its employee data. Create a view named DEPT50 that contains the employee numbers, employee last names, and department numbers for all employees in department 50. You have been asked to label the view columns EMPNO, EMPLOYEE, and DEPTNO. For security purposes, do not allow an employee to be reassigned to another department through the view.



5. Display the structure and contents of the DEPT50 view

```
SELECT * FROM DEPT_50;
```

Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 45 en 0.009 segundos

	EMPNO	EMPLOYEE	DEPTNO
1	120	Weiss	50
2	121	Fripp	50
3	122	Kaufling	50
4	123	Vollman	50
5	124	Mourgos	50
6	125	Nayer	50
7	126	Mikkilineni	50
8	127	Landry	50
9	128	Markle	50
10	129	Bissot	50
11	130	Atkinson	50
12	131	Marlow	50
13	132	Olson	50
14	133	Mallin	50
15	134	Rogers	50
16	135	Gee	50
17	136	Philtanker	50
18	137	Ladwig	50
19	138	Stiles	50
20	139	Seo	50
21	140	Patel	50
22	141	Rajs	50
23	142	Davies	50
24	143	Matos	50
25	144	Vargas	50
26	180	Taylor	50
27	181	Fleaur	50
28	182	Sullivan	50

6. Test your view. Attempt to reassign Matos to department 80. What is the result? Explain.

```
SELECT * FROM DEPT_50;
DESCRIBE DEPT_50;
```

Salida de Script | Resultado de la Consulta

Tarea terminada en 0.597 segundos

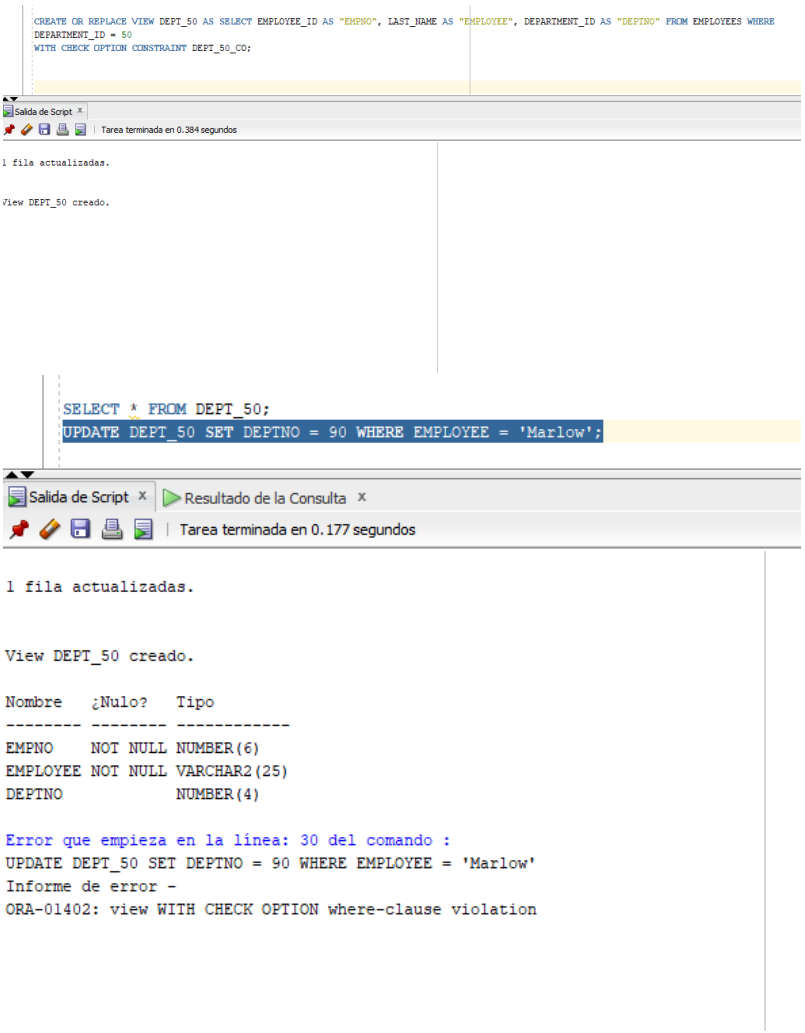
```
View EMPLOYEES_VU creado.


View DEPT_50 creado.

Nombre   ¿Nulo?   Tipo
-------- -------- ------------
EMPNO    NOT NULL NUMBER(6)
EMPLOYEE NOT NULL VARCHAR2(25)
DEPTNO             NUMBER(4)
```

```
UPDATE DEPT_50 SET DEPTNO = 90 WHERE EMPLOYEE = 'Fripp';
```

Salida de Script

Tarea terminada en 0.054 segundos

```
1 fila actualizadas.
```



Professor I moved the number to Matos and it no longer came out in the consultation, but I put the name of another I had forgotten to put the restriction, but I put it in order to see what was asked of us after I realized it An error is returned because the WITH CHECK OPTION constraint is violated, only employee data with department id 50 can be updated.s

b) Problem 2.

1. Create views on the EMPLOYEES (empanonv) and DEPARTMENT (deptanonv) tables that queries representative personal information (hiredate, jobid, salary,commissionpct, departmentid; departmentid, departmentname, locarionid) :

```
--b)

SELECT * FROM EMPLOYEES;
SELECT * FROM DEPARTMENTS;
CREATE OR REPLACE VIEW emp_anon_v AS SELECT
HIRE_DATE, JOB_ID, SALARY, COMMISSION_PCT, DEPARTMENT_ID
FROM EMPLOYEES;

CREATE OR REPLACE VIEW dep_anon_v  AS SELECT
DEPARTMENT_ID, DEPARTMENT_NAME, LOCATION_ID
FROM DEPARTMENTS;
```

```
Salida de Script
Tarea terminada en 0.146 segundos

View EMP_ANON_V creado.


View DEP_ANON_V creado.
```

2. Create a complex view that will join and aggregate (sum salary grouped by department's id and name [salaries], count the number of employees [staff]) the two simple views. Name the view depsumv. Note that there is no reason not to have views of views.

```
CREATE OR REPLACE VIEW dep_sum_v AS SELECT D.DEPARTMENT_ID ,D.DEPARTMENT_NAME,
SUM(E.SALARY) "SALARIES", COUNT(*) "STAFF"
FROM emp_anon_v E JOIN dep_anon_v D ON(E.DEPARTMENT_ID = D.DEPARTMENT_ID)
GROUP  BY D.DEPARTMENT_ID, D.DEPARTMENT_NAME;

SELECT * FROM DEP_SUM_V;
```

```
Salida de Script
Tarea terminada en 0.049 segundos

View DEP_SUM_V creado.
```

3. Query the resulting table:

```
SELECT * FROM DEP_SUM_V;
```

Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 11 en 0.009 segundos

	DEPARTMENT_ID	DEPARTMENT_NAME	SALARIES	STAFF
1	100	Finance	51608	6
2	50	Shipping	140200	43
3	70	Public Relations	10000	1
4	30	Purchasing	32900	7
5	90	Executive	66200	4
6	10	Administration	4400	1
7	110	Accounting	20308	2
8	40	Human Resources	6500	1
9	20	Marketing	19000	2
10	60	IT	28800	5
11	80	Sales	304500	34

4. Execute these statements and show results.

c) Problem 3. 1. Insert a new department through the DEPTANONV view and attempt to insert an employee through EMPANONV: insert into DEPTANONV values(99,'Temp Dept',1800); insert into EMPANONV values(sysdate,'ACMGR',10000,0,99); What is the result? Why? Try to update through it: update empanonv set salary=salary*1.1; What is the result? Why? Then roll back the changes:

```
--c)

INSERT INTO DEP_ANON_V VALUES(99,'Temp Dept', 1800);
INSERT INTO EMP_ANON_V VALUES(sysdate, 'AC_MGR',10000,0,99);
```

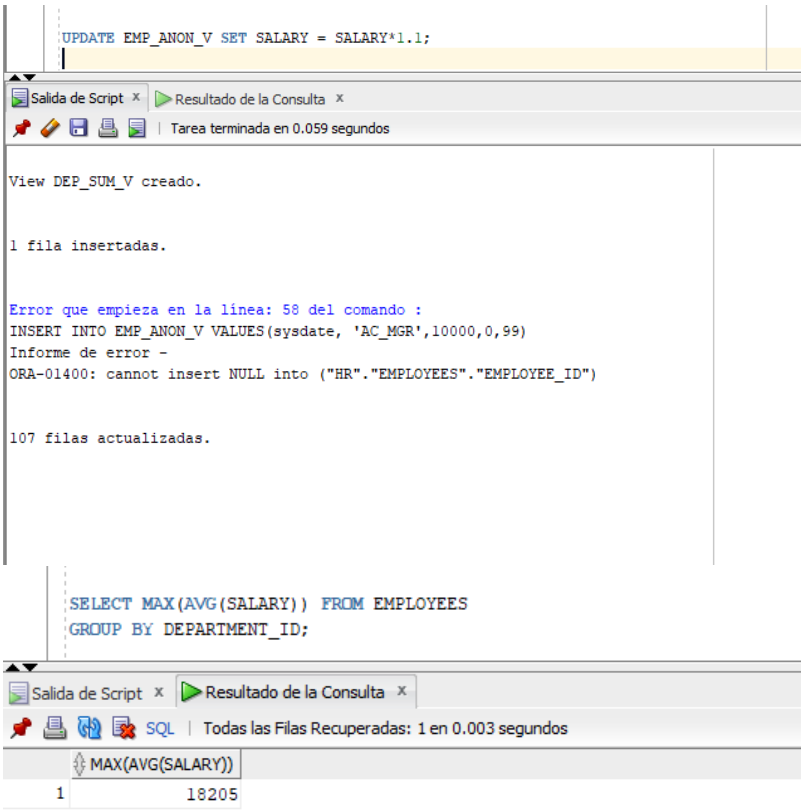
Salida de Script | Resultado de la Consulta

Tarea terminada en 0.437 segundos

```
View DEP_SUM_V creado.


1 fila insertadas.


Error que empieza en la línea: 58 del comando :
INSERT INTO EMP_ANON_V VALUES(sysdate, 'AC_MGR',10000,0,99)
Informe de error -
ORA-01400: cannot insert NULL into ("HR"."EMPLOYEES"."EMPLOYEE_ID")
```

2. Find out the salary of the department with the highest average salary, by querying the EMPLOYEES table (use a subquery in the FROM clause to extract the average salary from employees):

and find the same information from the DEPSUMV view, which is a much simpler query:



3. Execute these statements and show results.

d) Problem 4. From de following tables: CREATE TABLE departments1 (name CHAR(20) PRIMARY KEY, office CHAR(50)); CREATE TABLE employees1 (id SMALLINT PRIMARY KEY, surname CHAR(40), name CHAR(20), salary INTEGER, dep CHAR(20) REFERENCES departments1);

1. Insert some data in both tables.

```
CREATE TABLE departments1 (
name CHAR(20) PRIMARY KEY,
office CHAR(50));

CREATE TABLE employees1 (
id SMALLINT PRIMARY KEY,
surname CHAR(40),
name CHAR(20),
salary INTEGER,
dep CHAR(20) REFERENCES departments1);
```

Salida de Script | Resultado de la Consulta

Tarea terminada en 0.447 segundos

```
1 fila insertadas.


Error que empieza en la línea: 58 del comando :
INSERT INTO EMP_ANON_V VALUES(sysdate, 'AC_MGR',10000,0,99)
Informe de error -
ORA-01400: cannot insert NULL into ("HR"."EMPLOYEES"."EMPLOYEE_ID")


107 filas actualizadas.


Table DEPARTMENTS1 creado.


Table EMPLOYEES1 creado.
```

```
INSERT INTO departments1 VALUES('SOFTWARE MX', 'UAZ');
INSERT INTO departments1 VALUES('DEPARTMENT ITMX', 'UAZ');
INSERT INTO departments1 VALUES('ANSWERSTO DATA BASES', 'UAZ');
INSERT INTO departments1 VALUES('SALES', 'UAZ');
SELECT * FROM departments1;
```

Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 4 en 0.001 segundos

	NAME	OFFICE
1	SOFTWARE MX	UAZ
2	DEPARTMENT ITMX	UAZ
3	ANSWERSTO DATA BASES	UAZ
4	SALES	UAZ

```
INSERT INTO employees1 VALUES(1, 'AMCDanyMX', 'DANIEL', 24000,'SOFTWARE MX');
INSERT INTO employees1 VALUES(2, 'ALDONSOPROFE', 'ALDONSO', 30000,'DEPARTMENT ITMX');
INSERT INTO employees1 VALUES(3, 'FLIKK', 'JACINTO', 20000,'ANSWERSTO DATA BASES');
INSERT INTO employees1 VALUES(4, 'RF200', 'JUAN', 22000,'SALES');
INSERT INTO employees1 VALUES(5, 'CR7', 'CRISTIANO', 12000,'SALES');
SELECT * FROM employees1;
```

Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 5 en 0.002 segundos

	ID	SURNAME	NAME	SALARY	DEP
1	1	AMCDanyMX	DANIEL	24000	SOFTWARE MX
2	2	ALDONSOPROFE	ALDONSO	30000	DEPARTMENT ITMX
3	3	FLIKK	JACINTO	20000	ANSWERSTO DATA BASES
4	4	RF200	JUAN	22000	SALES
5	5	CR7	CRISTIANO	12000	SALES

2. Create a view called EmpDepSales that contains the surname and first name of the employees of the department "Sales" renaming those properties as "lastname" and "firstname". Use alias out of the subquery.

```
SELECT * FROM EMPDEPSALES ORDER BY FIRST_NAME ASC;
```

Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 2 en 0.003 segundos

	LAST_NAMES	FIRST_NAME
1	CR7	CRISTIANO
2	RF200	JUAN

3. List the surname and first name of the employees of the department "Sales" alphabetically.

```
SELECT * FROM EMPDEPSALES ORDER BY FIRST_NAME ASC;
```

Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 2 en 0.003 segundos

	LAST_NAMES	FIRST_NAME
1	CR7	CRISTIANO
2	RF200	JUAN

4. Alter the view EmpDepSales to also include the salary.

```
CREATE OR REPLACE VIEW EMPDEPSALES AS SELECT SURNAME AS "LAST_NAMES", NAME AS "FIRST_NAME" , SALARY AS "SALARY_EMP"
FROM EMPLOYEES1
WHERE DEP = 'SALES';
```

Salida de Script

Tarea terminada en 0.061 segundos

```
iew EMPDEPSALES creado.
```

5. Create a view that displays the minimum, maximum and average salaries of the employees of each department.

```
CREATE VIEW EMPAVERAGE AS SELECT MIN(E.SALARY) AS "MIN", MAX(E.SALARY) AS "MAX", D.DEP
 FROM EMPLOYEES1 E JOIN DEPARTMENTS1 D
 ON(E.DEP = D.DEP)
GROUP BY D.DEP, E.DEP;

SELECT * FROM EMPAVERAGE;
```

Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 4 en 0.004 segundos

	MIN	MAX	DEP
1	30000	30000	DEPARTMENT ITMX
2	24000	24000	SOFTWARE MX
3	12000	22000	SALES
4	20000	20000	ANSWERSTO DATA BASES

6. Describe each view.

```
DESCRIBE EMPDEPSALES;
DESCRIBE EMPAVERAGE;
```

Salida de Script

Tarea terminada en 0.085 segundos

```
Nombre      ¿Nulo? Tipo
----------- ------ -----------
LAST_NAMES         CHAR(40)
FIRST_NAME         CHAR(20)
SALARY_EMP         NUMBER(38)
Nombre ¿Nulo?   Tipo
------ -------- --------
MIN              NUMBER
MAX              NUMBER
DEP     NOT NULL CHAR(20)
```

7. Drop each view created.

```
DROP VIEW EMPDEPSALES;
DROP VIEW EMPAVERAGE;
```

Salida de Script

Tarea terminada en 0.052 segundos

```
Nombre      ¿Nulo? Tipo
----------- ------ -----------
LAST_NAMES         CHAR(40)
FIRST_NAME         CHAR(20)
SALARY_EMP         NUMBER(38)
Nombre ¿Nulo?   Tipo
------ -------- --------
MIN              NUMBER
MAX              NUMBER
DEP     NOT NULL CHAR(20)

Error que empieza en la línea: 126 del comando :
DROP VIEW DEMPDEPSALES
Informe de error -
ORA-00942: table or view does not exist
00942. 00000 -  "table or view does not exist"
*Cause:
*Action:

View EMPAVERAGE borrado.


View EMPDEPSALES borrado.
```

e) Problem 5.

1. Connect to your database as user HR. 2. Create synonyms for the three views created in Exercise b, problem 2:

```
--e)
CREATE SYNONYM EMPLEADOS FOR emp_anon_v;
CREATE SYNONYM DEPARTAMENTOS FOR dep_anon_v;
CREATE SYNONYM SUMADEPARTAMENTOS FOR dep_sum_v;
```

```
Salida de Script
Tarea terminada en 0.405 segundos

Synonym EMPLEADOS creado.

Synonym DEPARTAMENTOS creado.

Synonym SUMADEPARTAMENTOS creado.
```

3. Confirm that the synonyms are identical to the underlying object:

```
SELECT * FROM EMPLEADOS;
SELECT * FROM DEPARTAMENTOS;
SELECT * FROM SUMADEPARTAMENTOS;
```

Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 11 en 0.003 segundos

	DEPARTMENT_ID	DEPARTMENT_NAME	SALARIES	STAFF
1	100	Finance	56768.8	6
2	50	Shipping	154220	43
3	70	Public Relations	11000	1
4	30	Purchasing	36190	7
5	90	Executive	72820	4
6	10	Administration	4840	1
7	110	Accounting	22338.8	2
8	40	Human Resources	7150	1

4. Confirm that the synonyms work (even to the extent of producing the same errors) by running the statements in b and c against the synonyms instead of the views:

```
INSERT INTO DEPARTAMENTOS VALUES(99,'Temp Dept', 1900);
INSERT INTO EMPLEADOS VALUES(sysdate, 'AC_MGR',10000,0,99);


UPDATE EMP_ANON_V SET SALARY = SALARY*1.1;

SELECT MAX(AVG(SALARY)) FROM EMPLOYEES
GROUP BY DEPARTMENT_ID;

SELECT MAX(SALARIES/STAFF)
FROM DEP_SUM_V;
```

```
Salida de Script | Resultado de la Consulta
Tarea terminada en 0.056 segundos

Error que empieza en la linea: 143 del comando :
INSERT INTO DEPARTAMENTOS VALUES(99,'Temp Dept', 1900)
Informe de error -
ORA-00001: unique constraint (HR.DEPT_ID_PK) violated


Error que empieza en la linea: 144 del comando :
INSERT INTO EMPLEADOS VALUES(sysdate, 'AC_MGR',10000,0,99)
Informe de error -
ORA-01400: cannot insert NULL into ("HR"."EMPLOYEES"."EMPLOYEE_ID")
```

```
UPDATE EMPLEADOS SET SALARY = SALARY*1.1;

SELECT MAX(AVG(SALARY)) FROM EMPLEADOS
GROUP BY DEPARTMENT_ID;

SELECT MAX(SALARIES/STAFF)
FROM SUMADEPARTAMENTOS;
```

Salida de Script | Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 1 en 0.004 segundos

	MAX(SALARIES/STAFF)
1	20025.5

5. Drop two of the views:

```
DROP VIEW EMP_ANON_V;
DROP VIEW DEP_ANON_V;
```

Salida de Script | Tarea terminada en 0.077 segundos

```
View EMP_ANON_V borrado.

View DEP_ANON_V borrado.
```

6. Query the complex view that is based on the dropped views: Note that the query fails.

```
SELECT * FROM SUMADEPARTAMENTOS;
```

Salida de Script | Resultado de la Consulta

SQL | En Ejecución:SELECT * FROM SUMADEPARTAMENTOS en 0 segundos

```
ORA-04063: view "HR.DEP_SUM_V" has errors
04063. 00000 -  "%s has errors"
*Cause:    Attempt to execute a stored procedure or use a view that has
           errors.  For stored procedures, the problem could be syntax errors
           or references to other, non-existent procedures.  For views,
           the problem could be a reference in the view's defining query to
           a non-existent table.
           Can also be a table which has references to non-existent or
           inaccessible types.
*Action:   Fix the errors and/or create referenced objects as necessary.
Error en la línea: 159, columna: 15
```

7. Attempt to recompile the broken view: This will fail as well.
8. Drop the DEPSUMV view:

```
DROP VIEW DEP_SUM_V;
```

Salida de Script | Resultado de la Consulta | Tarea terminada en 0.06 segundos

```
View EMP_ANON_V borrado.

View DEP_ANON_V borrado.

View DEP_SUM_V borrado.
```

9. Query the synonym for a dropped view: This will fail.

```
SELECT * FROM SUMADEPARTAMENTOS;
```

Salida de Script | Resultado de la Consulta | SQL | En Ejecución:SELECT * FROM SUMADEPARTAMENTOS en 0 segundos

```
ORA-00980: synonym translation is no longer valid
00980. 00000 -  "synonym translation is no longer valid"
*Cause:    A synonym did not translate to a legal target object. This
           could happen for one of the following reasons:
           1. The target schema does not exist.
           2. The target object does not exist.
           3. The synonym specifies an incorrect database link.
           4. The synonym is not versioned but specifies a versioned
           target object.
*Action:   Change the synonym definition so that the synonym points at
           a legal target object.
Error en la línea: 163, columna: 15
```

10. Recompile the broken synonym: Note that this does not give an error, but rerun the query from step 9. It is definitely still broken.
11. Tidy up by dropping the synonyms:

```
DROP SYNONYM SUMADEPARTAMENTOS;
DROP SYNONYM EMPLEADOS;
DROP SYNONYM DEPARTAMENTOS;
```

Salida de Script | Resultado de la Consulta | Tarea terminada en 0.063 segundos

```
View DEP_SUM_V borrado.

Synonym SUMADEPARTAMENTOS borrado.

Synonym EMPLEADOS borrado.

Synonym DEPARTAMENTOS borrado.
```

The NOTE: Capture an image for each statement output

3 PRE-EVALUATION

Practices pre-Assessment for Database Systems Laboratory II Pre-Assessment PRACTICE 15 carried out by student

1 COMPLIES WITH THE REQUESTED FUNCTIONALITY
YES

4 HAS THE CORRECT INDENTATION
YES

6 HAS AN EASY WAY TO ACCESS THE PROVIDED FILES
YES

7 HAS A REPORT WITH IDC FORMAT
YES

8 REPORT INFORMATION IS FREE OF SPELLING ERRORS
YES

9 DELIVERED IN TIME AND FORM
YES

10 IS FULLY COMPLETED (SPECIFY THE PERCENTAGE COMPLETED)
YES,100 percent

4 Conclusion

This practice was very helpful to me to finish understanding and practicing with the views, it is a subject that is relatively simple but like everyone you must pay attention when doing them to do them correctly.