



ESCUOLA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA EN COMPUTACIÓN

ESCUOLA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
CARRERA DE INGENIERIA EN COMPUTACIÓN

Laboratorio de:

Fundamentos de bases de datos

Práctica No.:

Tema: Vistas

Objetivos:

Creacion de vistas de base de datos

Marco teórico:

1. Vistas (Views)

Una vista es un objeto de base de datos que actúa como una **tabla virtual**. No almacena datos por sí misma (salvo excepciones como las vistas materializadas), sino que almacena la definición de una consulta SELECT.

Características Principales

Característica	Descripción
Abstracción y Simplificación	Oculta la complejidad de las consultas subyacentes (como JOINs complejos).
Seguridad	Permite restringir el acceso a columnas o filas específicas sin dar acceso a la tabla base completa.
Actualización Dinámica	Los datos mostrados reflejan automáticamente los cambios realizados en las tablas base.

Tipos de Vistas

Tipo de Vista	Descripción	Detalles y Restricciones
Vistas Simples	Basadas en una sola tabla.	<ul style="list-style-type: none">Generalmente permiten operaciones DML (INSERT, UPDATE, DELETE).No contienen funciones de grupo (GROUP BY, DISTINCT).
Vistas Complejas	Basadas en múltiples tablas (usando JOIN).	<ul style="list-style-type: none">Pueden contener funciones de agregación o expresiones calculadas.Generalmente no son actualizables directamente.
Vistas Materializadas (Oracle)	Almacenan físicamente los datos resultantes.	<ul style="list-style-type: none">Mejoran el rendimiento en consultas complejas.



		• Requieren estrategias de refresco (REFRESH) para sincronizarse.
--	--	-------------------------------------------------------------------

Sintaxis y Cláusulas Importantes

La creación estándar sigue la estructura CREATE VIEW nombre AS SELECT A continuación se detallan cláusulas específicas importantes:

Cláusula	Función
OR REPLACE / OR ALTER	Permite modificar o reemplazar una vista existente sin eliminarla primero.
WITH CHECK OPTION	Garantiza que los datos insertados o actualizados a través de la vista cumplan con la condición WHERE definida en ella.
WITH READ ONLY	Bloquea cualquier intento de modificación de datos (INSERT, UPDATE, DELETE) a través de la vista.
FORCE (Oracle)	Permite crear la vista incluso si las tablas base a las que hace referencia no existen todavía.

Operaciones DML (Restricciones de Actualización)

Una vista **NO** es actualizable si contiene cualquiera de los siguientes elementos en su definición:

Elemento Restrictivo	Ejemplo
Funciones de Agregación	SUM, AVG, COUNT, MAX, MIN
Agrupamientos	Cláusulas GROUP BY o HAVING
Operadores de Conjunto	UNION, INTERSECT, MINUS
Otros	Subconsultas correlacionadas o uso de DISTINCT

2. Procedimientos Almacenados y Programación Persistente

Desde la versión SQL:1999, SQL soporta la definición de funciones y procedimientos para registrar la "lógica de negocio" dentro de la propia base de datos.

Conceptos Fundamentales

Concepto	Definición
PSM (Persistent Storage Module)	Parte de la norma SQL que define los constructores procedimentales. Permite ejecutar lógica directamente en el servidor de BD.
Extensiones Propietarias	Implementaciones específicas de cada motor: PL/SQL (Oracle) y TransactSQL (SQL Server), las cuales pueden variar en sintaxis respecto al estándar.

Definición y Uso

Un procedimiento se crea con CREATE PROCEDURE y se diferencia de una función principalmente en su forma de invocación.

Característica	Procedimiento Almacenado
Parámetros	Admite parámetros de entrada (IN) y de salida (OUT).



Invocación	Se invoca explícitamente mediante la instrucción CALL (no dentro de una expresión SELECT).
Ejemplo	call proc_recuento_cuentas('Cliente', variable_resultado);

Constructores Procedimentales (Control de Flujo)

SQL soporta estructuras de control similares a los lenguajes de programación tradicionales:

Tipo	Instrucción	Descripción
Compuestas	BEGIN ... END	Bloques que agrupan varias instrucciones SQL y declaraciones de variables locales.
Bucles	WHILE	Ejecuta el bloque mientras se cumpla una condición (while n < 10 do ...).
	REPEAT	Repite el bloque hasta que se cumpla una condición (until n=0).
	FOR	Itera sobre los resultados de una consulta, abriendo implícitamente un cursor para leer fila por fila.
Condicionales	IF-THEN-ELSE	Estructura de decisión basada en condiciones lógicas (ej. evaluar saldos).
	CASE	Estructura selectiva múltiple (similar a switch en C/C++).

Manejo de Excepciones

Elemento	Uso
Declaración	DECLARE nombre CONDITION para definir errores personalizados.
Manejador	DECLARE EXIT HANDLER define la acción a tomar (salir o continuar) al ocurrir el error.
Predefinidas	Condiciones estándar del sistema como SQLEXCEPTION, SQLWARNING y NOT FOUND.

Desarrollo de la práctica:

1. Ingresar a sqlplus o sql-developer con usuario que se indique en clases.
2. Ejecutar las sentencias que se indiquen en clases

```
SQL> CREATE USER fmerino IDENTIFIED BY 123
  2  DEFAULT TABLESPACE USERS
  3  TEMPORARY TABLESPACE TEMP
  4  PROFILE DEFAULT;
```

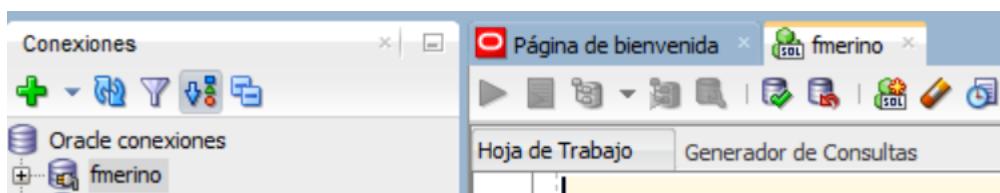
Usuario creado.

```
SQL>
```



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA EN COMPUTACIÓN

```
SQL> GRANT CONNECT TO fmerino;  
Concesión terminada correctamente.  
  
SQL> ALTER USER fmerino QUOTA UNLIMITED ON USERS;  
Usuario modificado.  
  
SQL>
```





ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA EN COMPUTACIÓN

Hoja de Trabajo Generador de Consultas

```
CREATE TABLE MATRICULAS (
    mat_id NUMBER(8) PRIMARY KEY,
    est_id NUMBER(6),
    curso_id NUMBER(4),
    mat_nota NUMBER(4,2),
    mat_periodo VARCHAR2(10),
    FOREIGN KEY (est_id) REFERENCES ESTUDIANTES(est_id),
    FOREIGN KEY (curso_id) REFERENCES CURSOS(curso_id)
);
```

Salida de Script x

Table ESTUDIANTES creado.

Table CURSOS creado.

Table MATRICULAS creado.

Tarea terminada en 0,063 segundos



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA EN COMPUTACIÓN

Hoja de Trabajo Generador de Consultas

```
CREATE OR REPLACE VIEW V_ESTUDIANTES_SISTEMAS AS
SELECT est_id,
       est_nombre || ' ' || est_apellido AS nombre_completo,
       est_creditos,
       est_promedio
  FROM ESTUDIANTES
 WHERE est_carrera = 'Sistemas'
   WITH READ ONLY;
```

Salida de Script x

View V_ESTUDIANTES_SISTEMAS creado.

Tarea terminada en 0,089 segundos



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA EN COMPUTACIÓN

Tareas (filtrado)

Vistas

V_ESTUDIANTES_SISTEMAS

- EST_ID
- NOMBRE_COMPLETO
- EST_CREDITOS
- EST_PROMEDIO

Índices

Página de bienvenida fmerino V_ESTUDIANTES_SISTEMAS

Columnas Datos Permisos Dependencias Detalles Disparadores SQL Errores

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS	INSERTABLE	UPDATABLE	DELETABLE
1 EST_ID	NUMBER(6)	No	(null)	1 (null)	NO	NO	NO	NO
2 NOMBRE_COMPLETO	VARCHAR2(101)	Yes	(null)	2 (null)	NO	NO	NO	NO
3 EST_CREDITOS	NUMBER(3)	Yes	(null)	3 (null)	NO	NO	NO	NO
4 EST_PROMEDIO	NUMBER(4, 2)	Yes	(null)	4 (null)	NO	NO	NO	NO

Hoja de Trabajo Generador de Consultas

```
-- Consultar la vista
SELECT * FROM V_ESTUDIANTES_SISTEMAS
WHERE est_promedio >= 80;
```

Salida de Script Resultado de la Consulta

SQL | Todas las Filas Recuperadas: 0 en 0,028 segundos

EST_ID	NOMBRE_...	EST_CRE...	EST_PRO...
--------	------------	------------	------------



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA EN COMPUTACIÓN

Hoja de Trabajo Generador de Consultas

```
CREATE OR REPLACE VIEW V_REPORTE_MATRICULAS AS
SELECT
    e.est_id,
    e.est_nombre || ' ' || e.est_apellido AS estudiante,
    c.curso_nombre,
    c.curso_creditos,
    m.mat_nota,
```

Salida de Script x Resultado de la Consulta x

SQL | Tarea terminada en 0,039 segundos

View V_ESTUDIANTES_SISTEMAS creado.

View V_REPORTE_MATRICULAS creado.

Hoja de Trabajo Generador de Consultas

```
SELECT estudiante, curso_nombre, mat_nota, estado
FROM V_REPORTE_MATRICULAS
WHERE mat_periodo = '2024-1' AND estado = 'APROBADO';
```

Salida de Script x Resultado de la Consulta x

SQL | Todas las Filas Recuperadas: 0 en 0,023 segundos

ESTUDIA... CURSO_N... MAT_NOTA ESTADO



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA EN COMPUTACIÓN

Hoja de Trabajo Generador de Consultas

```
CREATE OR REPLACE VIEW V_ESTADISTICAS_CARRERA AS
SELECT
    est_carrera AS carrera,
    COUNT(*) AS total_estudiantes,
    ROUND(AVG(est_promedio), 2) AS promedio_general,
    MAX(est_promedio) AS promedio_maximo,
    MIN(est_promedio) AS promedio_minimo,
    ROUND(AVG(est_creditos), 0) AS creditos_promedio
FROM ESTUDIANTES
GROUP BY est_carrera;
```

Salida de Script x Resultado de la Consulta x

SQL | Tarea terminada en 0,062 segundos

View V_ESTADISTICAS_CARRERA creado.

Hoja de Trabajo Generador de Consultas

```
-- Consulta
SELECT * FROM V_ESTADISTICAS_CARRERA
ORDER BY promedio_general DESC;
```

Salida de Script x Resultado de la Consulta x

SQL | Todas las Filas Recuperadas: 0 en 0,009 segundos

CARRERA	TOTAL_E...	PROMEDI...	PROMEDI...	CRE...
---------	------------	------------	------------	--------



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA EN COMPUTACIÓN

Hoja de Trabajo Generador de Consultas

```
CREATE OR REPLACE VIEW V_ESTUDIANTES_DESTACADOS AS
SELECT est_id, est_nombre, est_apellido, est_promedio
FROM ESTUDIANTES
WHERE est_promedio >= 85
WITH CHECK OPTION CONSTRAINT chk_promedio_destacado;
```

Salida de Script X

View V_ESTUDIANTES_DESTACADOS creado.

Tarea terminada en 0,032 segundos



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA EN COMPUTACIÓN

Hoja de Trabajo Generador de Consultas

```
-- Esto funcionará
INSERT INTO V_ESTUDIANTES_DESTACADOS
VALUES (1001, 'Maria', 'González', 90.5);
```

Salida de Script x

View V_ESTUDIANTES_DESTACADOS creado.

1 fila insertadas.



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA EN COMPUTACIÓN

Hoja de Trabajo Generador de Consultas

```
-- Esto generará ERROR (viola WITH CHECK OPTION)
INSERT INTO V_ESTUDIANTES_DESTACADOS
VALUES (1002, 'Juan', 'Pérez', 75.0);
```

Salida de Script x

| Tarea terminada en 0,091 segundos

```
Error que empieza en la línea: 2 del comando -
INSERT INTO V_ESTUDIANTES_DESTACADOS
VALUES (1002, 'Juan', 'Pérez', 75.0)
Error en la línea de comandos : 2 Columna : 13
Informe de error -
Error SQL: ORA-01402: violación de la cláusula WHERE en la vista

https://docs.oracle.com/error-help/db/ora-01402/01402. 00000 -
*Cause: An INSERT or UPDATE statement was attempted on a view
created with the CHECK OPTION. This would have resulted
creation of a row that would not satisfy the view's WHERE
clause.
*Action: Examine the view's WHERE clause in the dictionary
table VIEWS. If the current view does not have the CHECK
OPTION, then its FROM clause must reference a second view
that is defined using the CHECK OPTION. The second view's WHERE
clause must also be satisfied by any INSERT or UPDATE
statements. To insert the row, it may be necessary to
```



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA EN COMPUTACIÓN

The screenshot shows the Oracle SQL Developer interface. At the top, there is a toolbar with various icons for database operations like running scripts, connecting, and searching. To the right of the toolbar, the user's name 'fmerino' is displayed. Below the toolbar, there are two tabs: 'Hoja de Trabajo' (Worksheet) and 'Generador de Consultas' (Query Builder). The 'Hoja de Trabajo' tab is active and contains the following SQL code:

```
-- Oracle
CREATE OR REPLACE VIEW V_ESTUDIANTES_EDITABLE AS
SELECT est_id, est_nombre, est_apellido, est_carrera
FROM ESTUDIANTES;
```

In the bottom panel, there is a 'Salida de Script' (Script Output) window. This window shows the execution results of the script. It includes a toolbar with icons for running, saving, and opening files, and a status message indicating the task was completed in 0,031 seconds. The output text in the window is:

```
View V_ESTUDIANTES_EDITABLE creado.
```



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA EN COMPUTACIÓN

Hoja de Trabajo Generador de Consultas

```
-- Operaciones permitidas
UPDATE V_ESTUDIANTES_EDITABLE
SET est_carrera = 'Electrónica'
WHERE est_id = 1001;
```

Salida de Script X

| Tarea terminada en 0,064 segundos

View V_ESTUDIANTES_EDITABLE creado.

1 fila actualizadas.

Hoja de Trabajo Generador de Consultas

```
DELETE FROM V_ESTUDIANTES_EDITABLE
WHERE est_id = 1002;
```

Salida de Script X

| Tarea terminada en 0,055 segundos

0 filas eliminado



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA EN COMPUTACIÓN

Hoja de Trabajo Generador de Consultas

```
-- Creación de Vista Materializada
CREATE MATERIALIZED VIEW MV_RESUMEN_CURSOS
BUILD IMMEDIATE
REFRESH COMPLETE ON DEMAND
AS
SELECT
    c.curso_departamento,
    COUNT(*) AS total_cursos,
    SUM(c.curso_creditos) AS creditos_totales,
    COUNT(DISTINCT m.est_id) AS estudiantes_unicos
FROM CURSOS c
LEFT JOIN MATRICULAS m ON c.curso_id = m.curso_id
GROUP BY c.curso_departamento;
```

Salida de Script x

! Tarea terminada en 0,657 segundos

you the required privilege(s).

Materialized view MV_RESUMEN_CURSOS creado.



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA EN COMPUTACIÓN

Hoja de Trabajo Generador de Consultas

```
EXEC DBMS_MVIEW.REFRESH ('MV_RESUMEN_CURSOS');
```

Salida de Script X

Script Editor | Tarea terminada en 0,507 segundos

Materialized view MV_RESUMEN_CURSOS created.

Procedimiento PL/SQL terminado correctamente.

Tablas de Colas | Disparadores | Tipos | Secuencias | Vistas Materializadas | Logs de Vistas Materiales | Sinónimos

Acciones... | Columnas | Datos | Índices | Permisos | Dependencias | Detalles | SQL

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT
1 CURSO_DEPARTAMENTO	VARCHAR2(30 BYTE)	Yes	(null)
2 TOTAL_CURSOS	NUMBER	Yes	(null)
3 CREDITOS_TOTALES	NUMBER	Yes	(null)
4 ESTUDIANTES_UNICOS	NUMBER	Yes	(null)



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA EN COMPUTACIÓN

Hoja de Trabajo Generador de Consultas

```
select * from tab;
```

Salida de Script x Resultado de la Consulta x

SQL | Todas las Filas Recuperadas: 9 en 0,034 segundos

	TNAME	TABTYPE	CLUSTERID
1	ESTUDIANTES	TABLE	(null)
2	CURSOS	TABLE	(null)
3	MATRICULAS	TABLE	(null)
4	V_ESTUDIANTES_SISTEMAS	VIEW	(null)
5	V_Reporte_Matriculas	VIEW	(null)
6	V_Estadisticas_Carrera	VIEW	(null)
7	V_Estudiantes_Destacados	VIEW	(null)
8	V_Estudiantes_Editables	VIEW	(null)
9	MV_Resumen_Cursos	TABLE	(null)



**ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERÍA DE SISTEMAS
INGENIERÍA EN COMPUTACIÓN**

-- Ver todas las vistas del usuario
SELECT view_name, text
FROM USER_VIEWS;

	VIEW_NAME	TEXT
1	V_ESTUDIANTES_SISTEMAS	SELECT est_id, est_nombre '
2	V_Reporte_Matriculas	SELECT e.est_id, e.est_nombre '
3	V_ESTADISTICAS_CARRERA	SELECT est_carrera AS carrera, (
4	V_ESTUDIANTES_DESTACADOS	SELECT est_id, est_nombre, est_a
5	V_ESTUDIANTES_EDITABLE	SELECT est_id, est_nombre, est_a

The screenshot shows the Oracle SQL Developer interface. The top menu bar includes 'Hoja de Trabajo' and 'Generador de Consultas'. The main area contains two SQL statements:

```
SELECT view_name, text  
FROM USER_VIEWS;  
  
-- Ver definición de una vista específica  
SELECT text FROM USER_VIEWS  
WHERE view_name = 'V_ESTUDIANTES_SISTEMAS';
```

Below the SQL editor, there are three tabs: 'Salida de Script', 'Resultado de la Consulta', and 'Resultado de la Consulta 1'. The status bar at the bottom indicates 'Todas las Filas Recuperadas: 1 en 0,02 segundos'.



The screenshot shows the Oracle SQL Developer interface. In the top-left tab, 'Hoja de Trabajo' (Worksheet) is selected. The code area contains the following SQL command:

```
-- Oracle y SQL Server
DROP VIEW V_ESTUDIANTES_SISTEMAS;
```

In the bottom panel, the status bar displays: 'Salida de Script x | Resultado de la Consulta x | Resultado de la Consulta x' and 'Tarea terminada en 0,302 segundos' (Task completed in 0,302 seconds).

Output window content:

```
View V_ESTUDIANTES_SISTEMAS borrado.
```

Análisis de resultados:

En la práctica fue posible comprobar el funcionamiento de los tipos de vistas en Oracle, se observó las características de cada uno, como en las vistas simples que permitieron las operaciones DML directas, pese a eso la falta de datos dentro de nuestra "base" hizo más difícil visualizar el funcionamiento de las vistas, que de todas formas dieron resultados favorables a los comandos ejecutados.

Conclusiones y recomendaciones:

En conclusión, las vistas constituyen un mecanismo fundamental para la seguridad y abstracción de la base de datos, permitiendo simplificar consultas complejas (como los JOINs en V_REPORTES_MATRICULAS) y ocultar la estructura real de las tablas al usuario final. La implementación de cláusulas como WITH CHECK OPTION y READ ONLY demostró ser efectiva para garantizar la integridad de los datos, asegurando que las modificaciones realizadas a través de la vista cumplan estrictamente con las reglas de negocio definidas en su creación, evitando así la inserción de datos "invisibles" para la vista.

Por otro lado, se determinó que la elección entre una vista estándar y una vista materializada depende de la necesidad de actualización frente al rendimiento. Las vistas estándar garantizan datos en tiempo real al no almacenar información, ideal para entornos transaccionales, mientras que las vistas materializadas, como la creada en la práctica, son superiores para consultas de agregación pesada (COUNT, SUM) al almacenar el resultado físico, con la condición de gestionar adecuadamente sus tiempos de refresco (ON DEMAND) para mantener la consistencia.

Bibliografía:

- [1] B. Chancusig, Capítulo 4.1: *Vistas*. 2025.
- [2] B. Chancusig, *Capítulo 4: Lenguajes de consulta - Programación persistente*. 2025.