



REPORTE 8

BASES DE DATOS

DIAZ MEDINA JESÚS KAIMORTS
PROFESOR: M. EN C. HERNÁNDEZ CONTRERAS EULER
| GRUPO: 2CM6 |

INDICE

MARCO TEÓRICO.....	2
Álgebra Relacional.	2
Proyección (π).....	2
Selección (δ).....	2
Producto Cartesiano(X).....	3
DESARROLLO DE LA PRÁCTICA.....	1
Diagrama de la base de datos.	1
Instrucciones.	1

MARCO TEÓRICO.

ÁLGEBRA RELACIONAL.

- ✚ Es un conjunto de operaciones simples sobre tablas relacionales, a partir de las cuales se definen operaciones más complejas mediante composición. Definen, por tanto, un pequeño lenguaje de manipulación de datos.
- ✚ El álgebra relacional consiste de algunas simples, pero poderosas maneras de construir nuevas relaciones a partir de otras. Si pensamos que las relaciones iniciales son los datos almacenados entonces las nuevas relaciones se pueden ver como respuestas a algunas consultas deseadas.

PROYECCIÓN (π)

- ✚ Crea una nueva relación a partir de otra, pero incluyendo sólo algunas de las columnas.
- ✚ $\pi A_1, A_3, A_6 (R)$

Título	Año	Páginas	Tipo de film	Estudio
Star Wars	1977	124	Color	Fox
Mighty Ducks	1991	104	Color	Disney
Wayne's Wold	1992	95	Color	Paramount

Ejemplo.

- ✚ π título, año, paginas (Movie)

Título	Año	Páginas
Star Wars	1977	124
Mighty Ducks	1991	104
Wayne's Wold	1992	95

- ✚ π Tipo de Film (Movie)

Tipo de film
Color
Color
Color

SELECCIÓN (δ)

- ✚ Crea una nueva relación a partir de otra, pero incluyendo sólo algunas de las tuplas a partir de un criterio dado.

- ✚ El criterio se basa en restricciones sobre los atributos de la relación R y no pueden incluirse otras relaciones en dicho criterio que no estén en R.
- ✚ $\delta A_3 > 16, \delta A_3 > 16 \text{ AND } A_3 < 45 (R), \delta \text{ nombre} = 'Carlos' \text{ AND } \text{edad} = 45 (R)$

Título	Año	Páginas	Tipo de film	Estudio
Star Wars	1977	124	Color	Fox
Mighty Ducks	1991	104	Color	Disney
Wayne's Woldr	1992	95	Color	Paramount

Ejemplo

- ✚ $\delta \text{ length} \geq 100 (\text{Movie})$

Título	Año	Páginas	Tipo de film	Estudio
Star Wars	1977	124	Color	Fox
Mighty Ducks	1991	104	Color	Disney

- ✚ $\delta \text{ length} \geq 100 \text{ AND } \text{Estudio} = 'Fox' (\text{Movie})$

Título	Año	Páginas	Tipo de film	Estudio
Star Wars	1977	124	Color	Fox

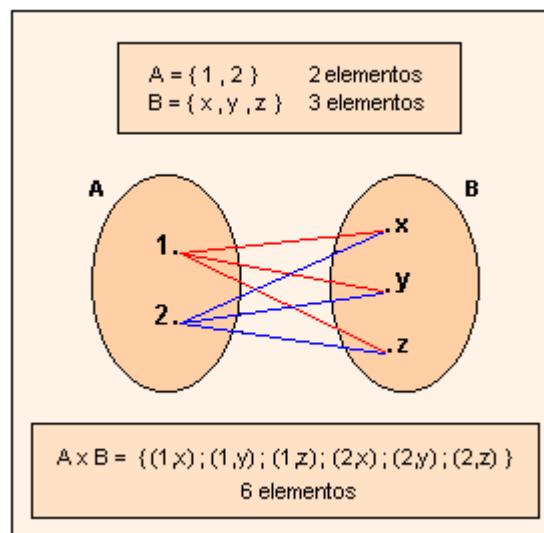
- ✚ $\pi \text{ Titulo, Estudio}(\delta \text{ length} \geq 100 (\text{Movie}))$

Título	Estudio
Star Wars	Fox
Mighty Ducks	Disney

PRODUCTO CARTESIANO(X)

- ✚ El producto cartesiano de dos conjuntos A x B es el conjunto de todos los pares ordenados que se pueden formar con un elemento perteneciente al conjunto A y un elemento del conjunto B.
- ✚ Los elementos de A x B son pares ordenados. Cada par que se forma con un elemento del conjunto A y uno del conjunto B, en ese orden y recibe el nombre de par ordenado. Sus elementos se colocan entre paréntesis, separados por coma.

$$(x, y) \neq (y, x)$$



Ejemplo

Empleados

CLAVE	NOMBRE	OFICIO
001	Juan Carlos Torres Méndez	Chofer
002	Mónica Comes Pérez	Enfermera
003	Carmen Cruz Cruz	Dentista
004	Miguel Contreras León	Ingeniero

Colonias

CLAVE	COLONIA	C.P.
001	Méndez	86003
002	Rovirosa	86105
003	Sabina	87100
004	Tamulte	85300

Empleados.Nombre, Empleados.Oficio, Empleados.Colonia δ Empleados.Nombre = 'Miguel Contreras León' \wedge Empleados.Clave = Colonia.Clave (Empleados X Colonias).

CLAVE	NOMBRE	OFICIO	CLAVE	COLONIA	C.P.
001	Juan Carlos Torres Méndez	Chofer	001	Méndez	86003
001	Juan Carlos Torres Méndez	Chofer	002	Rovirosa	86105
001	Juan Carlos Torres Méndez	Chofer	003	Sabina	87100
001	Juan Carlos Torres Méndez	Chofer	004	Tamulte	85300
002	Mónica Comes Pérez	Enfermera	001	Méndez	86003
002	Mónica Comes Pérez	Enfermera	002	Rovirosa	86105
002	Mónica Comes Pérez	Enfermera	003	Sabina	87100
002	Mónica Comes Pérez	Enfermera	004	Tamulte	85300
003	Carmen Cruz Cruz	Dentista	001	Méndez	86003
003	Carmen Cruz Cruz	Dentista	002	Rovirosa	86105
003	Carmen Cruz Cruz	Dentista	003	Sabina	87100
003	Carmen Cruz Cruz	Dentista	004	Tamulte	85300
004	Miguel Contreras León	Ingeniero	002	Rovirosa	86105

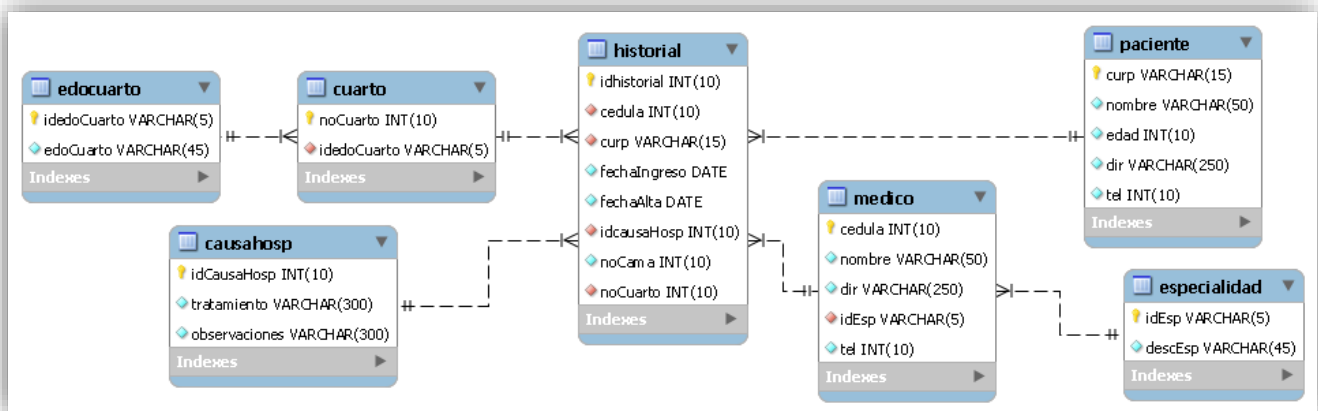
004	Miguel Contreras León	Ingeniero	003	Sabina	87100
004	Miguel Contreras León	Ingeniero	001	Méndez	86003
004	Miguel Contreras León	Ingeniero	004	Tamulte	85300

Finalmente, el resultado será.

NOMBRE	OFICIO	COLONIA
Miguel Contreras León	Ingeniero	Tamulte

DESARROLLO DE LA PRÁCTICA

DIAGRAMA DE LA BASE DE DATOS.



INSTRUCCIONES.

1. Crear la base de datos correspondiente, usarla y cargar el script (proporcionado por el profesor).

```
mysql> CREATE DATABASE HOSPITAL;
Query OK, 1 row affected (0.00 sec)
```

```
mysql> USE AGENCIA;
Database changed
mysql> SOURCE C:\Users\Home\Documents\ESCOM\3er Semestre\02 Bases de Datos\P8\hosp1.sql
Query OK, 0 rows affected (0.00 sec)
```

Generaremos las siguientes consultas.

- Mostrar todos los datos de los pacientes.

```
SELECT * FROM Paciente;
```

```
mysql> SELECT * FROM Paciente;
```

curp	nombre	edad	dir	tel
ABHA911215	ABAD HERNANDEZ ALDO	19	conocida	55555555
ACEC890217	APARICIO CORAZA ELOISA CONSUELO	21	conocida	55555555
ACNC880920	AVILA COLIN NURIA DEL CARMEN	22	conocida	55555555
AGRK800719	AGUILAR RUIZ KEYLA	23	Campo Deportivo 618	7198835
AMAU890611	ARVIZU MENDOZA AGUSTIN URIEL	21	conocida	55555555
AMDI890318	ARELLANES MOLINA DAMIAN ISAID	22	conocida	55555555

Se muestran 106 registros

- Mostrar el nombre y la edad de los pacientes.

```
SELECT nombre, edad
FROM paciente ORDER BY 2;
```

Se mostrarán 106 registros.

```
mysql> SELECT nombre, edad FROM paciente ORDER BY 2;
```

nombre	edad
FERNANDEZ BENHUMEA JONATHAN	18
FLORES MARTINEZ FABIAN ERNESTO	18
FLORES RAMIREZ BLANCA LISET	18
FRANCO ROJAS ALEJANDRO	18

- Mostrar los datos del paciente que tienen el siguiente CURP: MALD770810.

```
SELECT * FROM paciente
WHERE CURP LIKE "MALD770810";
```

curp	nombre	edad	dir	tel
MALD770810	MARTINEZ LOPEZ DANIEL	26	Rio Papaloapan 306	7113863

- Mostrar aquellos datos de los pacientes que tienen 26 años, nombre y edad.

```
SELECT nombre, edad
FROM paciente
WHERE edad = 26;
```

nombre	edad
MARTINEZ FRANCO ESLY	26
MARTINEZ LOPEZ DANIEL	26
RODRIGUEZ ESPINDOLA GUADALUPE	26

- Mostrar los pacientes que tienen más de 25 años de edad.

```
SELECT nombre, edad
FROM paciente
WHERE edad > 25 ORDER BY 2;
```

nombre	edad
MARTINEZ FRANCO ESLY	26
MARTINEZ LOPEZ DANIEL	26
RODRIGUEZ ESPINDOLA GUADALUPE	26
LOPEZ PALACIOS MIRIAM NAYELI	27
RODRIGUEZ BARRERA JAEI	27
GARCIA RIOS OMAR	28
PALACIOS PEREZ NESTOR	28
GOMEZ PABLO RAMIRO	29
PEREZ PEREZ NOHEMI	29
LEZAMA ESTRADA ELIZABETH	30
MENDOZA SOLIS RUBI	30

7. Visualizar pacientes cuya edad sea igual o mayor a 27 años.

```
SELECT nombre, edad FROM paciente
WHERE edad >=27 --edad>26
ORDER BY 2;
```

```
mysql> SELECT nombre, edad FROM paciente
-> WHERE edad >=27 --edad>26
-> ORDER BY 2;
Empty set (0.00 sec)
```

8. Visualizar pacientes que tienen menor o igual a 27 años.

```
SELECT nombre, edad FROM paciente
WHERE edad < 28 --edad <=27
ORDER BY edad;
```

```
mysql> SELECT nombre, edad FROM paciente
-> WHERE edad < 28 --edad <=27
-> ORDER BY edad;
+-----+-----+
| nombre | edad |
+-----+-----+
| FERNANDEZ BENHUMEA JONATHAN | 18 |
| FLORES MARTINEZ FABIAN ERNESTO | 18 |
| FLORES RAMIREZ BLANCA LISET | 18 |
+-----+-----+
```

9. Saber quiénes son los pacientes que tienen 27 años.

```
+-----+-----+
| nombre | edad |
+-----+-----+
| LOPEZ PALACIOS MIRIAM NAYELI | 27 |
| RODRIGUEZ BARRERA JAEL | 27 |
+-----+-----+
```

```
SELECT nombre, edad
FROM paciente
WHERE edad = 27
ORDER BY 2;
```

10. Saber quiénes son los pacientes que tienen 26, 27 y 28 años.

```
SELECT nombre, edad FROM paciente
WHERE edad BETWEEN 26 AND 28
ORDER BY 2;
```

```
--Una segunda manera
SELECT nombre, edad FROM paciente
WHERE edad IN(26,27,28)
ORDER BY 2;
```

```
--Una tercera manera
SELECT nombre, edad FROM paciente
WHERE (edad = 26 OR edad = 27 OR
edad = 28)
ORDER BY 2;
```

```
+-----+-----+
| nombre | edad |
+-----+-----+
| MARTINEZ FRANCO ESLY | 26 |
| MARTINEZ LOPEZ DANIEL | 26 |
| RODRIGUEZ ESPINDOLA GUADALUPE | 26 |
| LOPEZ PALACIOS MIRIAM NAYELI | 27 |
| RODRIGUEZ BARRERA JAEL | 27 |
| GARCIA RIOS OMAR | 28 |
| PALACIOS PEREZ NESTOR | 28 |
+-----+-----+
```

11. Visualizar aquellos pacientes que tienen 26 años o bien, que tengan 28 años.

```
+-----+-----+
| nombre | edad |
+-----+-----+
| MARTINEZ FRANCO ESLY | 26 |
| MARTINEZ LOPEZ DANIEL | 26 |
| RODRIGUEZ ESPINDOLA GUADALUPE | 26 |
| GARCIA RIOS OMAR | 28 |
| PALACIOS PEREZ NESTOR | 28 |
+-----+-----+
```

```
SELECT nombre, edad
FROM paciente
WHERE edad IN(26,28)
ORDER BY 2;
```


12. Visualizar aquellos pacientes que tiene 26 años en adelante y que no tengan 28 años.

nombre	edad
MARTINEZ FRANCO ESLY	26
MARTINEZ LOPEZ DANIEL	26
RODRIGUEZ ESPINDOLA GUADALUPE	26
LOPEZ PALACIOS MIRIAM NAYELI	27
RODRIGUEZ BARRERA JAEI	27
GOMEZ PABLO RAMIRO	29
PEREZ PEREZ NOHEMI	29
LEZAMA ESTRADA ELIZABETH	30
MENDOZA SOLIS RUBI	30

```
SELECT nombre, edad FROM paciente
WHERE edad >= 26 AND edad <> 28
ORDER BY 2;
```

13. Mostrar el historial de aquellos pacientes que ingresaron el 26/03/2003.

```
SELECT * FROM historial
WHERE fechaIngreso = "2003/03/26";
```

idhistorial	cedula	curp	fechaIngreso	fechaAlta	idcausaHosp	noCama	noCuarto
4	31232	MESR731128	2003-03-26	2003-03-27	2	25	514
5	41258	PABS791225	2003-03-26	2003-04-01	1	23	512

14. Mostrar el historial de los pacientes que ingresaron después del 25 de marzo del 2003.

```
SELECT * FROM historial
WHERE fechaIngreso > "2003-03-25"
ORDER BY fechaIngreso;
```

idhistorial	cedula	curp	fechaIngreso	fechaAlta	idcausaHosp	noCama	noCuarto
4	31232	MESR731128	2003-03-26	2003-03-27	2	25	514
5	41258	PABS791225	2003-03-26	2003-04-01	1	23	512
6	96510	MALD770810	2003-03-27	2003-04-01	1	22	512
7	88743	PABS791225	2003-03-28	2003-04-05	2	50	517
8	89784	ORVE790526	2003-03-29	2003-04-01	1	51	517

Hay 101 registros.

15. Visualizar el historial de los pacientes que ingresaron desde el 25 a 27 de marzo del 2003.

```
SELECT * FROM historial
WHERE fechaIngreso BETWEEN "2003-03-25" AND "2003.03.27"
ORDER BY fechaIngreso;
```

idhistorial	cedula	curp	fechaIngreso	fechaAlta	idcausaHosp	noCama	noCuarto
3	23555	GOPR740604	2003-03-25	2003-03-30	1	24	513
4	31232	MESR731128	2003-03-26	2003-03-27	2	25	514
5	41258	PABS791225	2003-03-26	2003-04-01	1	23	512
6	96510	MALD770810	2003-03-27	2003-04-01	1	22	512

16. Conocer cuántos registros existen de pacientes cuya causa de hospitalización es igual a 2.

```
SELECT count(*)
FROM historial
WHERE idCausaHosp = 2;
```

count(*)
8

17. Actualizar el teléfono del médico Omar Cortes Landeros.

nombre	tel
CORTES LANDEROS OMAR	5544233

```
SELECT nombre, tel from medico
WHERE nombre like "Cort%Land%Oma%";
```

```
UPDATE medico SET TEL = 55555577
WHERE nombre like "Cort%Land%Oma%";
```

nombre	tel
CORTES LANDEROS OMAR	55555577

```
SELECT nombre, tel from medico
WHERE nombre like "Cort%Land%Oma%";
```

18. Actualizar el domicilio de los Neurólogos y ponerles: "VIVEN EN EL HOSPITAL".

nombre	dir
CORTES LANDEROS OMAR	conocida
PERDOMO VERGARA BENITO	conocida
LOPEZ MUNIOZ VERONICA	Hda. De Gavia 998
MONTER GARCIA EDGAR	Puerto Alvarado 14

```
SELECT nombre, dir FROM medico
WHERE idEsp = (SELECT idEsp FROM especialidad
WHERE descEsp like "Neur%");
```

```
UPDATE medico
SET dir ="VIVEN EN EL HOSPITAL"
WHERE idEsp = (SELECT idEsp FROM especialidad
WHERE descEsp like "Neur%");
```

nombre	dir
CORTES LANDEROS OMAR	VIVEN EN EL HOSPITAL
PERDOMO VERGARA BENITO	VIVEN EN EL HOSPITAL
LOPEZ MUNIOZ VERONICA	VIVEN EN EL HOSPITAL
MONTER GARCIA EDGAR	VIVEN EN EL HOSPITAL

```
SELECT nombre, dir FROM medico
WHERE idEsp = (SELECT idEsp FROM especialidad
WHERE descEsp like "Neur%");
```

19. Dar de alta 2 médicos. Uno: Urólogo, Dos: Ginecólogo.

```
INSERT INTO medico VALUES (181897,"Garcia Dorantes Miguel","En su casa",
(SELECT idEsp FROM especialidad
WHERE descEsp like "Urolo%"),52525656),
(181892,"Lopez Lopez Karla","En su casa",
(SELECT idEsp FROM especialidad
WHERE descEsp like "Ginec%"),91817356);
```

cedula	nombre	dir	idEsp	tel
181892	Lopez Lopez Karla	En su casa	GN	91817356
181897	Garcia Dorantes Miguel	En su casa	UR	52525656

```
SELECT * FROM medico
WHERE cedula
in(181897,181892);
```

20. Para aquellos pacientes que fueron dados de alta en diciembre del 2009 cambiar su dirección: "DESCONOCIDA".

nombre	dir	fechaAlta
ONOFRE DIAZ JORGE ANTONIO	conocida	2009-12-25
FLORES MARTINEZ FABIAN ERNESTO	conocida	2009-12-26
GUTIERREZ ORTINEZ EDUARDO	conocida	2009-12-26
MARTINEZ GALICIA LITAE	conocida	2009-12-26
JOAQUIN SANDOVAL JOSE DOMINGO	conocida	2009-12-30
SANCHEZ PEREZ URIEL AXEL	conocida	2009-12-31
SANCHEZ SANCHEZ JULIA LETICIA	conocida	2009-12-31
VAZQUEZ GUTIERREZ KAREN MARISOL	conocida	2009-12-31

```
SELECT p.nombre, p.dir, h.fechaAlta
FROM paciente p, historial h
WHERE h.curp = p.curp
AND h.fechaAlta BETWEEN "2009.12.01"
AND "2009/12/31"
ORDER BY 3;
```

```
UPDATE paciente p, historial h
SET p.dir = "DESCONOCIDA"
WHERE h.curp = p.curp
AND h.fechaAlta BETWEEN "2009.12.01"
AND "2009/12/31";
```

nombre	dir	fechaAlta
ONOFRE DIAZ JORGE ANTONIO	DESCONOCIDA	2009-12-25
FLORES MARTINEZ FABIAN ERNESTO	DESCONOCIDA	2009-12-26
GUTIERREZ ORTINEZ EDUARDO	DESCONOCIDA	2009-12-26
MARTINEZ GALICIA LITAE	DESCONOCIDA	2009-12-26
JOAQUIN SANDOVAL JOSE DOMINGO	DESCONOCIDA	2009-12-30
SANCHEZ PEREZ URIEL AXEL	DESCONOCIDA	2009-12-31
SANCHEZ SANCHEZ JULIA LETICIA	DESCONOCIDA	2009-12-31
VAZQUEZ GUTIERREZ KAREN MARISOL	DESCONOCIDA	2009-12-31

```
SELECT p.nombre, p.dir, h.fechaAlta
FROM paciente p, historial h
WHERE h.curp = p.curp
AND h.fechaAlta BETWEEN "2009.12.01"
AND "2009/12/31"
ORDER BY 3;
```

21. Eliminar los pacientes que fueron atendidos por el médico Samuel Duran Becerril.

El siguiente código no se ejecutará, pero es importante mencionarlo.

```
DELETE FROM paciente p, historial h, medico m
WHERE p.curp = h.curp
AND h.cedula = m.cedula
AND m.nombre like "Dur%";
--No funciona
```

```
SELECT p.nombre, p.curp, m.cedula, m.nombre
FROM paciente p, historial h, medico m
WHERE p.curp = h.curp
AND h.cedula = m.cedula
AND m.nombre like "Dur% Bece% Sam%";
```

```
DELETE FROM paciente
WHERE curp in( "BBJG881021", "MCGI910122", "VELF890818");
```

Seguiremos con el código utilizado el cual funciona

- a) Visualizamos con qué tablas está relacionada la tabla historial

```
SHOW CREATE TABLE historial;
```

```
CONSTRAINT `FK_historial_1` FOREIGN KEY (`cedula`) REFERENCES `medico` (`cedula`),
CONSTRAINT `FK_historial_2` FOREIGN KEY (`curp`) REFERENCES `paciente` (`curp`),
CONSTRAINT `FK_historial_3` FOREIGN KEY (`noCuarto`) REFERENCES `cuarto` (`noCuarto`),
CONSTRAINT `historial_ibfk_1` FOREIGN KEY (`idcausaHosp`) REFERENCES `causaHosp` (`idCausaHosp`) ON DELETE CASCADE ON UPDATE CASCADE
ENGINE=InnoDB DEFAULT CHARSET=latin1 |
```

- b) Borrar constraint de la FK.

```
ALTER TABLE historial DROP FOREIGN KEY FK_historial_2;

DELETE FROM paciente
WHERE curp in( "BBJG881021", "MCGI910122", "VELF890818");
```

- c) Consultar y hacer enlace de las FK.

```
SELECT p.nombre, p.curp, m.cedula, m.nombre
FROM paciente p, historial h, medico m
WHERE p.curp = h.curp
AND h.cedula = m.cedula
AND m.nombre like "Dur% Bece% Sam%";

ALTER TABLE historial ADD FOREIGN KEY (curp)
REFERENCES (curp) ON DELETE CASCADE ON UPDATE CASCADE;
```

CONCLUSIÓN.

Durante esta práctica se utilizaron las distintas operaciones de algebra relacional: Proyección, Plano Cartesiano y Selección, fueron empleadas para restringir la información de la base de datos (los registros), previamente cargados, con el fin de preservar las reglas de integridad.

Gracias a esto, se comprendió las ventajas de usar las vistas, las cuáles se han enumerado anteriormente, pero dentro de ésta práctica, particularmente, ayudó a facilitar la búsqueda de “consultas complejas”, las cuales en algún momento involucren más de una relación. Finalmente, se puso en práctica todos, o gran parte, de los conocimientos adquiridos respecto a vistas, consultas, englobados en los compiladores DDL y DML.