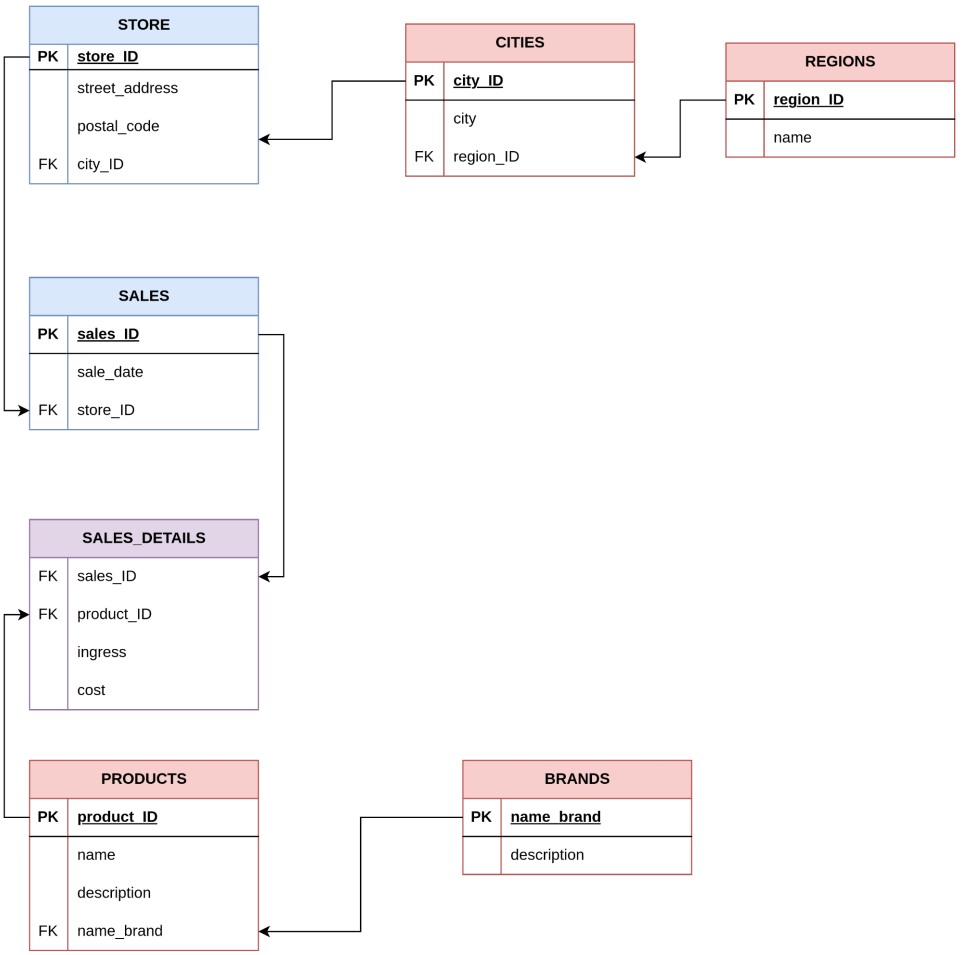
Práctica 4:

CREATE TABLE-CREATE VIEW

Exercici 1 - Taules




```
CREATE database store_pet;

CREATE TABLE regions(
        region_id INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
   name VARCHAR(100) NOT NULL
);

CREATE TABLE cities(
        city_id INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
   city VARCHAR(100) NOT NULL,
   region_id INT UNSIGNED NOT NULL,
   FOREIGN KEY (region_id) REFERENCES regions(region_id)
);
```

```
CREATE TABLE store(
        store_id INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
    street_address VARCHAR(100) NOT NULL,
    postal_code VARCHAR(100) NULL,
    city_id INT UNSIGNED NOT NULL,
    FOREIGN KEY (city_id) REFERENCES cities(city_id)
);

CREATE TABLE sales(
    sales_id INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
    sale_date DATE NOT NULL,
    store_id INT UNSIGNED NOT NULL,
    FOREIGN KEY (store_id) REFERENCES store(store_id)
);

CREATE TABLE brands(
    name_brand VARCHAR(100) NOT NULL PRIMARY KEY,
    description VARCHAR(300) NOT NULL
);

CREATE TABLE products(
    products_id INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL ,
    description VARCHAR(100) NULL,
    name_brand VARCHAR(100) NOT NULL,
    FOREIGN KEY (name_brand) REFERENCES brands(name_brand)
);

CREATE TABLE sales_details(
    products_id INT UNSIGNED NOT NULL,
        sales_id INT UNSIGNED NOT NULL ,
    ingress INT NOT NULL,
    cost INT NOT NULL,

    PRIMARY KEY(products_id, sales_id),
    FOREIGN KEY (products_id) REFERENCES products(products_id),
    FOREIGN KEY (sales_id) REFERENCES sales(sales_id)
);
```

Exercici 2 - Taules

1. SELECT de “Store” :

```
5 •  SELECT * FROM store_pet.store;
```

Result Grid | Filter Rows: | Edit:

store_id	street_address	postal_code	city_id
1	Carrer Rubio i Ors 15	8940	1
2	Carrer Major 1	8970	2
3	Carrer Camps Blancs 20	8830	3
4	Carrer de Calabria 174	8015	4
5	Carrer del dos de Maig	8912	5
6	Avinguda del Carrilet 24	8907	6
7	Carrer del Volcans	17800	7
12	Carretera de Barcelona 23	8203	8
13	Avinguda Jaume I	8223	9
15	Rambla Nova 25	43001	10
16	Passeig de Miramar	43830	11
17	Carrer de Valencia 40	8013	4
18	Carrer de la Miranda 10	8940	1
NULL	NULL	NULL	NULL

2. SELECT de “Products”:

```
1 •  SELECT * FROM products;
```

Result Grid | Filter Rows: | Edit:

products_id	name	description	name_brand
1	Bombilla de luz solar	NULL	Exo Terra
2	Transportin	NULL	Outech
3	Bebedero	NULL	Outech
4	Comida para perros carne	NULL	Nature's Variety
5	Comida para gatos salmon	NULL	Nature's Variety
6	Filtro carbón activo	NULL	Aquaclear
7	Filtro Mochila	NULL	Aquaclear
8	Snack Perros buey	NULL	Vitakraft
9	Mousse de pollo gatos	NULL	Vitakraft
10	Pienso apra conejos	NULL	Vitakraft
NULL	NULL	NULL	NULL

3. SELECT de “sales”:

```
1 •  SELECT * FROM sales;
```

Result Grid | Filter Rows: | Edit:

sales_id	sale_date	store_id
1	2015-02-05	1
2	2015-02-09	1
3	2018-04-05	1
4	2015-08-05	8
5	2015-02-05	4
6	2015-03-15	4
7	2018-07-19	2
8	2016-01-19	3
9	2016-02-19	7
10	2018-03-19	2
11	2017-03-19	6
12	2016-06-01	11
13	2016-10-05	13
14	2016-11-29	7

4. SELECT de “sales_details”:

```
5 •  SELECT * FROM store_pet.sales_details;
```

Result Grid | Filter Rows: | Edit:

products_id	sales_id	ingress	cost
1	1	50	30
1	5	40	25
1	10	25	15
2	2	20	15
2	7	15	10
2	11	30	20
3	3	15	10
3	8	20	12
3	12	25	15
4	4	80	60
4	9	50	40
4	13	35	25
5	5	25	20
5	14	40	30
5	18	30	25
6	6	35	25
6	15	50	40

5. SELECT de “brands”:

```
9 •  SELECT * FROM store_pet.brands;
```

Result Grid | Filter Rows: | Edit:

name_brand	description
Aquaclear	Marca que provee filtros para peceras
Exo Terra	Productos para el cuidado de reptiles
Nature's Variety	Marca especializa en la alimentación para animales
Outech	Marca que provee apratos, tales como jaulas, b...
Vitakraft	Marca especializada en la alimentación para ani...
NULL	NULL

6. SELECT de “cities”:

```
8 •  SELECT * FROM store_pet.cities;
```

Result Grid | Filter Rows:

city_id	city	region_id
1	Cornella de Llobregat	1
2	Sant Joan Despi	1
3	Sant Boi	1
4	Barcelona	2
5	Badalona	2
6	Hospitalet de Llobregat	2
7	Olot	3
8	Sabadell	4
9	Terrasa	4
10	Tarragona	5
11	Torredembarra	5
NULL	NULL	NULL

7. SELECT de “regions”:

```
9 •  SELECT * FROM store_pet.regions;
```

Result Grid | Filter Rows:

region_id	name
1	Baix Llobregat
2	Barcelones
3	Garrotxa
4	Valles Occidental
5	Tarragones
NULL	NULL

Exercici 3 - Taules

- INFORMATION SCHEMA tablas:

```
SELECT table_name, table_type, engine
FROM information_schema.tables
WHERE table_schema = 'store_pet';
```

	TABLE_NAME	TABLE_TYPE	ENGINE
▶	brands	BASE TABLE	InnoDB
	cities	BASE TABLE	InnoDB
	products	BASE TABLE	InnoDB
	regions	BASE TABLE	InnoDB
	sales	BASE TABLE	InnoDB
	sales_details	BASE TABLE	InnoDB
	store	BASE TABLE	InnoDB

- INFORMATION SCHEMA columnas:

```
SELECT table_name, column_name, data_type, character_maximum_length, is_nullable
FROM information_schema.columns
WHERE table_schema = 'store_pet';
```

TABLE	COLUMN	TYPE	MAX LENGHT	NULLABLE
brands	description	varchar	300	NO
brands	name_brand	varchar	100	NO
cities	city	varchar	100	NO
cities	city_id	int		NO
cities	region_id	int		NO
products	description	varchar	100	YES
products	name	varchar	100	NO
products	name_brand	varchar	100	NO
products	products_id	int		NO

regions	name	varchar	100	NO
regions	region_id	int		NO
sales	sale_date	date		NO
sales	sales_id	int		NO
sales	store_id	int		NO
sales_details	cost	int		NO
sales_details	ingress	int		NO
sales_details	products_id	int		NO
sales_details	sales_id	int		NO
store	city_id	int		NO
store	postal_code	varchar	100	YES
store	store_id	int		NO
store	street_address	varchar	100	NO

- INFORMATION SCHEMA columnas:

```
SELECT constraint_name, table_name, column_name, referenced_table_name,
referenced_column_name
FROM information_schema.key_column_usage
WHERE table_schema = 'store_pet' AND referenced_table_name IS NOT NULL;
```

CONSTRAINT_NAME	TABLE_NAME	COLUMN_NAME	REFERENCED_TABLE_NAME	REFERENCED_COLUMN_NAME
cities_ibfk_1	cities	region_id	regions	region_id
products_ibfk_1	products	name_brand	brands	name_brand
sales_ibfk_1	sales	store_id	store	store_id
sales_details_ibfk_1	sales_details	products_id	products	products_id
sales_details_ibfk_2	sales_details	sales_id	sales	sales_id
store_ibfk_1	store	city_id	cities	city_id

- INFORMATION SCHEMA primary keys:

```
SELECT constraint_name, table_name, column_name
FROM information_schema.key_column_usage
WHERE table_schema = 'store_pet' AND constraint_name = 'PRIMARY';
```

	CONSTRAINT_NAME	TABLE_NAME	COLUMN_NAME
▶	PRIMARY	brands	name_brand
	PRIMARY	cities	city_id
	PRIMARY	products	products_id
	PRIMARY	regions	region_id
	PRIMARY	sales	sales_id
	PRIMARY	sales_details	products_id
	PRIMARY	sales_details	sales_id
	PRIMARY	store	store_id

- INFORMATION SCHEMA datos NULL:

```
SELECT table_name, column_name, data_type
FROM information_schema.columns
WHERE table_schema = 'store_pet' AND is_nullable = 'YES';
```

	TABLE_NAME	COLUMN_NAME	DATA_TYPE
▶	products	description	varchar
	store	postal_code	varchar

Exercici 1 - Vistes

- Crear la vista:

```
CREATE VIEW v_product_sales AS
SELECT
    s.sales_id,
    s.sale_date,
    c.city,
    r.name AS region,
    p.name AS product,
    sd.ingress,
    sd.cost,
    sd.ingress - sd.cost AS profit

FROM sales s
    JOIN store st ON s.store_id = st.store_id
    JOIN cities c ON st.city_id = c.city_id
    JOIN regions r ON c.region_id = r.region_id
    JOIN sales_details sd ON s.sales_id = sd.sales_id
    JOIN products p ON sd.products_id = p.products_id;
```

- CONSULTAS:

- Llistat de productes de "Cornella de Llobregat"

```
SELECT DISTINCT product
FROM v_product_sales
WHERE city = 'Cornella de Llobregat';

SELECT DISTINCT p.name
FROM products p
JOIN brands b ON p.name_brand = b.name_brand
JOIN sales_details sd ON p.products_id = sd.products_id
JOIN sales s ON sd.sales_id = s.sales_id
JOIN store st ON s.store_id = st.store_id
JOIN cities c ON st.city_id = c.city_id
WHERE c.city = 'Cornella de Llobregat';
```

	product
▶	Bombilla de luz solar
	Transportin
	Bebedero
	Comida para perros carne

- Total ingress, cost, profit (se tiene que calcular) de "Baix Llobregat"

```
SELECT
   SUM(ingress) AS total_ingress,
   SUM(cost) AS total_cost,
   SUM(profit) AS total_profit
FROM v_product_sales
WHERE region = 'Baix Llobregat';
```

```
SELECT
   SUM(sd.ingress) AS total_ingress,
   SUM(sd.cost) AS total_cost,
   SUM(sd.ingress - sd.cost) AS total_profit
FROM sales_details sd
JOIN sales s ON sd.sales_id = s.sales_id
JOIN store st ON s.store_id = st.store_id
JOIN cities c ON st.city_id = c.city_id
JOIN regions r ON c.region_id = r.region_id
WHERE r.name = 'Baix Llobregat';
```

	total_ingress	total_cost	total_profit
▶	380	262	118

- Llistat de productes de la regions "Barcelones"

```
SELECT DISTINCT product
FROM v_product_sales
WHERE region = 'Barcelones';
```

```
SELECT DISTINCT p.name
FROM products p
JOIN brands b ON p.name_brand = b.name_brand
JOIN sales_details sd ON p.products_id = sd.products_id
JOIN sales s ON sd.sales_id = s.sales_id
JOIN store st ON s.store_id = st.store_id
JOIN cities c ON st.city_id = c.city_id
JOIN regions r ON c.region_id = r.region_id
WHERE r.name = 'Barcelones';
```

	product
▶	Bombilla de luz solar
	Comida para gatos salmon
	Mousse de pollo gatos
	Filtro carbón activo
	Snack Perros buey
	Transportin

- Total d'ingressos, costos, beneficis de la region "Barcelones"

```
SELECT
   SUM(ingress) AS total_ingress,
   SUM(cost) AS total_cost,
   SUM(profit) AS total_profit
FROM v_product_sales
WHERE region = 'Barcelones';
```

```
SELECT
   SUM(sd.ingress) AS total_ingress,
   SUM(sd.cost) AS total_cost,
   SUM(sd.ingress - sd.cost) AS total_profit
FROM sales_details sd
JOIN sales s ON sd.sales_id = s.sales_id
JOIN store st ON s.store_id = st.store_id
JOIN cities c ON st.city_id = c.city_id
JOIN regions r ON c.region_id = r.region_id
WHERE r.name = 'Barcelones';
```

total_ingress	total_cost	total_profit
195	135	60

- Total de productes de cada city.

```
SELECT
   city,
   COUNT(DISTINCT product) AS total_products
FROM v_product_sales
GROUP BY city;
```

```
SELECT
   c.city,
   COUNT(DISTINCT p.name) AS
total_products
FROM products p
JOIN brands b ON p.name_brand =
b.name_brand
JOIN sales_details sd ON p.products_id =
sd.products_id
JOIN sales s ON sd.sales_id = s.sales_id
JOIN store st ON s.store_id = st.store_id
JOIN cities c ON st.city_id = c.city_id
GROUP BY c.city;
```

city	total_products
Barcelona	5
Cornella de Llobregat	4
Hospitalet de Llobregat	1
Olot	3
Sabadell	3
Sant Boi	4
Sant Joan Despi	4
Terrasa	3
Torredembarra	3

- El producte amb ingress més elevats de cada city.

```
SELECT
   city,
   product,
   MAX(ingress) AS max_ingress
FROM v_product_sales
GROUP BY city, product;


WITH MaxIngressPerCity AS (
   SELECT
      c.city,
      p.name AS product,
      MAX(sd.ingress) AS max_ingress
   FROM products p
   JOIN brands b ON p.name_brand = b.name_brand
   JOIN sales_details sd ON p.products_id = sd.products_id
   JOIN sales s ON sd.sales_id = s.sales_id
   JOIN store st ON s.store_id = st.store_id
   JOIN cities c ON st.city_id = c.city_id
   GROUP BY c.city, p.name
)

SELECT
   city,
   product,
   max_ingress
FROM MaxIngressPerCity;
```

Cornella de Llobregat	Bombilla de luz solar	50
Cornella de Llobregat	Transportin	20
Cornella de Llobregat	Bebedero	15
Cornella de Llobregat	Comida para perros carne	35
Sant Joan Despi	Transportin	15
Sant Joan Despi	Filtro Mochila	50

Sant Joan Despi	Bombilla de luz solar	25
Sant Joan Despi	Pienso apra conejos	40
Sant Boi	Bebedero	20
Sant Boi	Snack Perros buey	45
Sant Boi	Pienso apra conejos	30
Sant Boi	Filtro Mochila	35
Barcelona	Bombilla de luz solar	40
Barcelona	Comida para gatos salmon	25
Barcelona	Mousse de pollo gatos	25
Barcelona	Filtro carbón activo	35
Barcelona	Snack Perros buey	40
Hospitalet de Llobregat	Transportin	30
Olot	Comida para perros carne	50
Olot	Mousse de pollo gatos	30
Olot	Comida para gatos salmon	40
Sabadell	Comida para perros carne	80
Sabadell	Filtro carbón activo	45
Sabadell	Snack Perros buey	30
Terrasa	Filtro carbón activo	50

Terrasa	Comida para gatos salmon	30
Terrasa	Mousse de pollo gatos	40
Torredembarra	Bebedero	25
Torredembarra	Filtro Mochila	20
Torredembarra	Pienso apra conejos	15

Exercici 2 - vistes

1. Proposeu 3 vistes per a fer consultes:

1. **Vista d'assignatures d'un departament**

```
CREATE VIEW v_assig_dep as
select a.id as id_assignatura, a.nom as assignatura, a.tipus, a.curs, a.quatrimestre,
        d.nom as nom_departament

from departament d
    join professor pr on d.id = pr.id_departament
    join assignatura a on pr.id_professor = a.id_professor;
```

2. **Vista professors en cada departament**

```
CREATE VIEW v_prof_depar as
select pr.id_professor,
       concat(p.nom,' ',p.cognom1,' ',p.cognom2) as nom_i_cognoms, p.sexe as sexe,
       d.id as id_departament, d.nom as nom_departament

from departament d
    join professor pr on d.id = pr.id_departament
    join persona p on pr.id_professor = p.id;
```

3. **Alumnes matriculats a cada assignatura**

```
CREATE OR REPLACE VIEW v_alum_a_assig as
select p.id,
    concat(p.nom,' ',p.cognom1,' ',p.cognom2) as nom_i_cognoms,
    g.nom as nom_grau,
    if (a.curs = 1,'primer',
    if (a.curs = 2,'segon',
    if (a.curs = 3,'tercer',
    if (a.curs = 4,'quart','altres')))) as curs_assignatura,
    a.nom as assignatura

from alumne_es_matricula_assignatura ama
    join assignatura a on ama.id_assignatura = a.id
    join persona p on ama.id_alumne = p.id
    join grau g on a.id_grau = g.id;
```

2. Creu una vista per a fer inserts a la taula de persones, aquesta vista ha de mostrar les persones que tenen un nif que comenci per '3'.

```
CREATE OR REPLACE VIEW v_persones_nif3 as
select *

from persona
where nif LIKE '3%'
with CHECK OPTION;
```

Feu un parell de INSERTS a partir de la vista, un que sigui OK i un que doni ERROR:

- OK:

```
INSERT INTO v_persones_nif3
VALUES (25, '36902806M', 'Javier', 'Morán', 'De Anta', 'MAdrid', 'C/ Real del barrio alto', '950254837', '1991-03-28', 'H', 'alumne');
```

- ERROR:

```
INSERT INTO v_persones_nif3
VALUES (24, '86902806M', 'Blanca', 'Crous', 'Cappacio', 'Barcelona', 'C/ Real del barrio alto', '950254837', '1991-03-28', 'M', 'alumne');
```

Exercici 3 - Vistes

BD_STORE:

- INFORMATION SCHEMA todas las vistas

```
SELECT
   table_name AS view_name,
   view_definition
FROM information_schema.views

WHERE table_schema = 'store_pet';
```

	view_name	VIEW_DEFINITION
▶	v_product_sales	select `s`.`sales_id` AS `sales_id`,`s`.`sale_...

- INFORMATION SCHEMA obtener columnas de la vista v_product_sales:

```
 SELECT
   table_name AS view,
   column_name,
   data_type,
   ordinal_position
FROM information_schema.columns
WHERE table_schema = 'store_pet' AND table_name = 'v_product_sales';
```

	view	COLUMN_NAME	DATA_TYPE	ORDINAL_POSITION
▶	v_product_sales	city	varchar	3
	v_product_sales	cost	int	7
	v_product_sales	ingress	int	6
	v_product_sales	product	varchar	5
	v_product_sales	profit	bigint	8
	v_product_sales	region	varchar	4
	v_product_sales	sale_date	date	2
	v_product_sales	sales_id	int	1

- INFORMATION SCHEMA obtener elementos de la vista v_product_sales:

```
SELECT
   TABLE_NAME AS view_name,
   COLUMN_NAME,
   COLUMN_TYPE,
   DATA_TYPE,
   CHARACTER_MAXIMUM_LENGTH,
   IS_NULLABLE
FROM information_schema.COLUMNS
WHERE TABLE_SCHEMA = 'store_pet' AND TABLE_NAME = 'v_product_sales';
```

view_name	COLUMN_NAME	COLUMN_TYPE	DATA_TYPE	CHARACTER_MAXIMUM_LENGTH	IS_NULLABLE
v_product_sales	city	varchar(100)	varchar	100	NO
v_product_sales	cost	int	int	NULL	NO
v_product_sales	ingress	int	int	NULL	NO
v_product_sales	product	varchar(100)	varchar	100	NO
v_product_sales	profit	bigint	bigint	NULL	NO
v_product_sales	region	varchar(100)	varchar	100	NO
v_product_sales	sale_date	date	date	NULL	NO
v_product_sales	sales_id	int unsigned	int	NULL	NO

BD_UNIVERSITAT:

Informació de totes les vistes:

```
SELECT
   table_name AS view_name,
   view_definition
FROM information_schema.views

WHERE table_schema = 'universitat';
```

	view_name	VIEW_DEFINITION
▶	v_alum_a_assig	select `p`.`id` AS `id`,concat(`p`.`nom`,'',`...
	v_assig_dep	select `a`.`id` AS `id_assignatura`,`a`.`nom...
	v_persones_nif3	select `universitat`.`persona`.`id` AS `id`,`u...
	v_prof_depar	select `pr`.`id_professor` AS `id_professor`,c...

Información sobre las columnas de la vista v_assig_dep:

```
SELECT
   table_name AS view,
   column_name,
   data_type,
   ordinal_position
FROM information_schema.columns
WHERE table_schema = 'universitat' AND table_name = 'v_assig_dep';
```

	view	COLUMN_NAME	DATA_TYPE	ORDINAL_POSITION
▶	v_assig_dep	id_assignatura	int	1
	v_assig_dep	assignatura	varchar	2
	v_assig_dep	tipus	enum	3
	v_assig_dep	curs	tinyint	4
	v_assig_dep	quatrimestre	tinyint	5
	v_assig_dep	nom_departament	varchar	6

Información sobre los elementos de la vista v_prof_depar:

```
SELECT
    TABLE_NAME AS view_name,
    COLUMN_NAME,
    COLUMN_TYPE,
    DATA_TYPE,
    CHARACTER_MAXIMUM_LENGTH,
    IS_NULLABLE
FROM information_schema.COLUMNS
WHERE TABLE_SCHEMA = 'universitat' AND TABLE_NAME = 'v_prof_depar';
```

view_name	COLUMN_NAME	COLUMN_TYPE	DATA_TYPE	CHARACTER_MAXIMUM_LENGTH	IS_NULLABLE
v_prof_depar	id_professor	int unsigned	int	NULL	NO
v_prof_depar	nom_i_cognoms	varchar(127)	varchar	127	YES
v_prof_depar	sexe	enum('H','M')	enum	1	NO
v_prof_depar	id_departament	int unsigned	int	NULL	NO
v_prof_depar	nom_departament	varchar(50)	varchar	50	NO