



PROYECTO FINAL

Contenido

Descripción de la temática de la base de datos	2
Modelo de negocios	2
Situación Problemática	2
Objetivo	2
Necesidades a cubrir	2
Diagramas de entidad relación.....	3
Conceptual	3
Esquemático	4
Tablas adicionales	5
Listado de las tablas que comprenden la base de datos.....	6
Vistas	10
Funciones	15
Store Procedures.....	17
Triggers.....	20
Inteligencia Empresarial (BI)	21
Tecnologías utilizadas.....	22
Link repositorio en github del archivo .sql	23

Descripción de la temática de la base de datos

Modelo de negocios

El modelo de negocios a presentar será una tienda de informática la cual además de comercializar productos de tecnología relacionados a la computación, nos brinda servicio técnico especializado.

Situación Problemática

Previo a la implementación de esta DB los registros de cada Área de la tienda eran cargados en tablas de Excel con la problemática de exceso de redundancia y tiempo empleado para el análisis y/o consulta de la información.

Objetivo

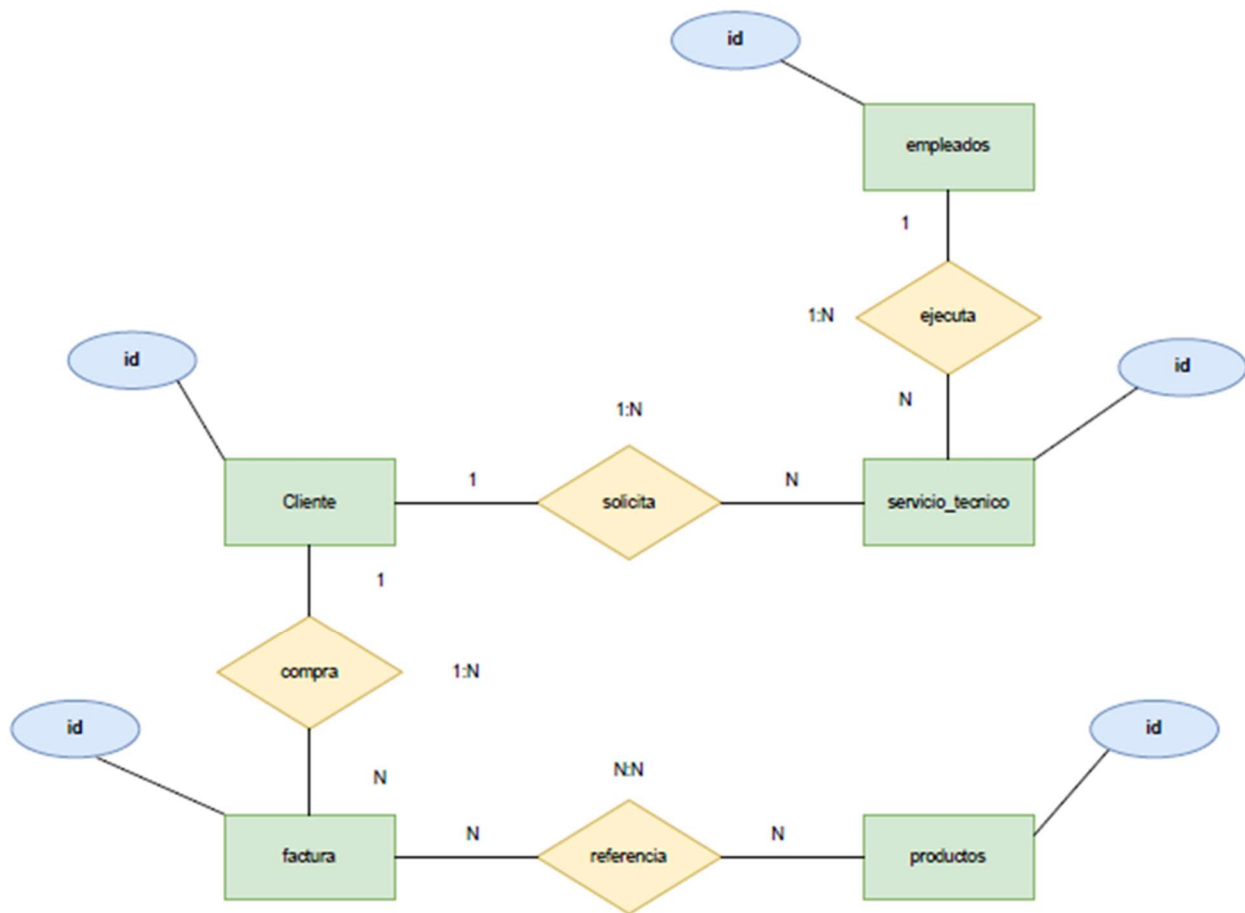
Nos centraremos en la creación de un sistema gestor de DB bajo un enfoque disciplinado el cual nos permita cuantificar la información.

Necesidades a cubrir

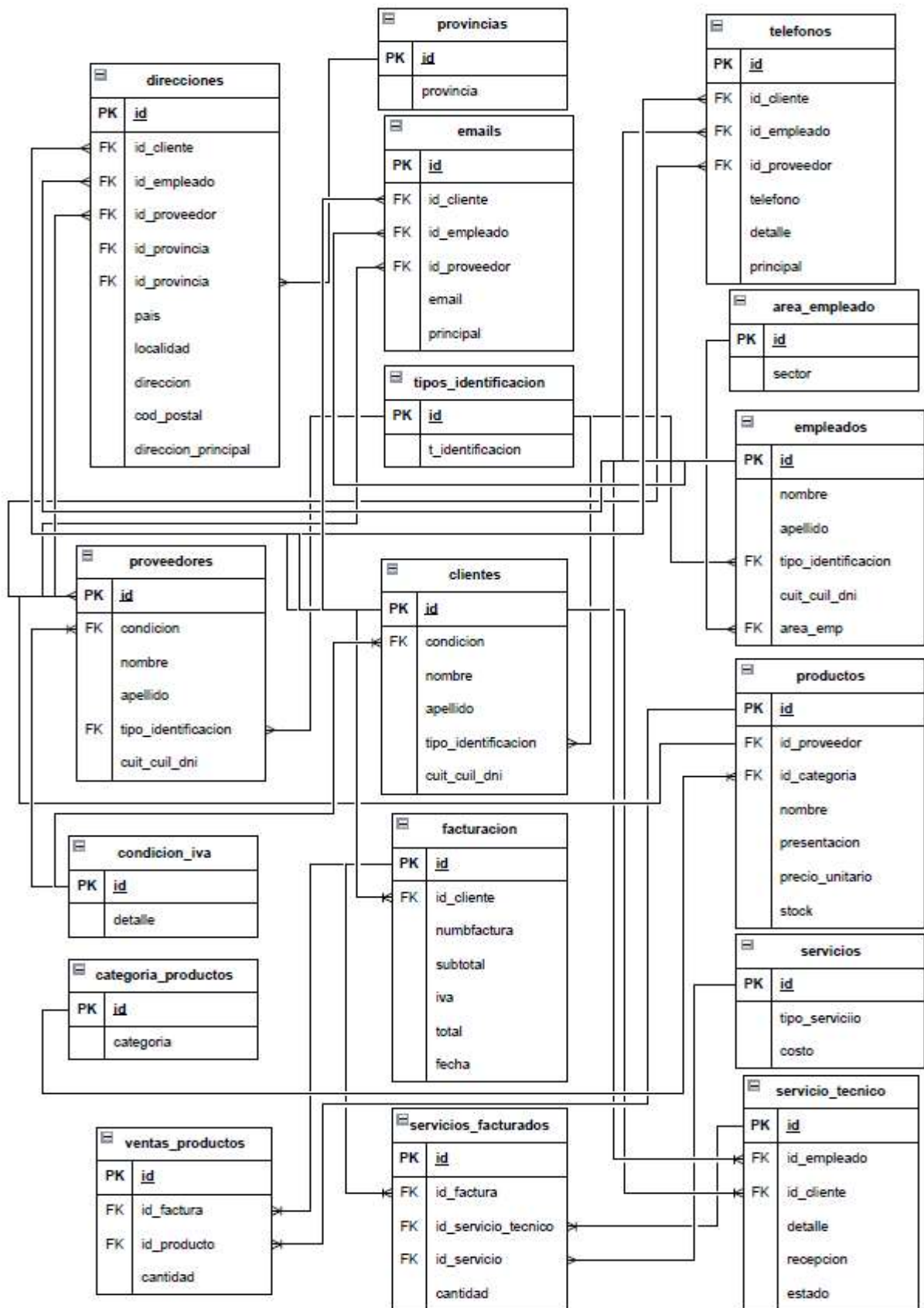
Reducir el tiempo de análisis y gestión datos, convirtiendo la información en recursos valiosos a la hora de tomar decisiones o evaluar estrategias de mercado.

Diagramas de entidad relación

Conceptual



Esquemático



Tablas adicionales

ventas_productos_respaldo	
PK	<u>id</u>
FK	id_cliente
FK	id_producto
	cantidad

ante_val_prod	
PK	<u>id</u>
FK	id_prod
	precio_ant
	fecha

mov_empleados	
PK	<u>id</u>
FK	id_empleado
	id_area_anterior
	fecha

Listado de las tablas que comprenden la base de datos

Tabla	condicion_iva					
Desc	Listado de las diferentes posiciones frente al IVA con el fin de eliminar redundancias					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	TINYINT		NOT NULL	SI	AUTO_INCREMENT
	condicion	VARCHAR	20	NOT NULL		

Tabla	tipos_identificacion					
Desc	Listado de las diferentes tipos de identificación con el fin de eliminar redundancias					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	TINYINT		NOT NULL	SI	AUTO_INCREMENT
	t_identificacion	VARCHAR	20	NOT NULL		

Tabla	cliente					
Desc	Almacenamos la información de los clientes					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	INT		NOT NULL	SI	AUTO_INCREMENT
FK	id_condicion	TINYINT		NOT NULL	NO	
	nombre	VARCHAR	30	NOT NULL	NO	
	apellido	VARCHAR	30	NULL	NO	
FK	tipo_identificacion	TINYINT		NOT NULL	NO	
	cuit_cuil_dni	VARCHAR	15	NOT NULL	NO	

Tabla	proveedores					
Desc	Almacenamos la información de los proveedores					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	INT		NOT NULL	SI	AUTO_INCREMENT
FK	id_condicion	TINYINT		NOT NULL	NO	
	nombre	VARCHAR	30	NOT NULL	NO	
	apellido	VARCHAR	30	NULL	NO	
FK	tipo_identificacion	TINYINT		NOT NULL	NO	
	cuit_cuil_dni	VARCHAR	15	NOT NULL	NO	

Tabla	area_empleado					
Desc	Sectores identificados con un codigo unico a lo cuales pertenece cada empleado					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	TINYINT		NOT NULL	SI	AUTO_INCREMENT
	sector	VARCHAR	25	NOT NULL		

Tabla	empleado					
Desc	Almacenamos la información de los clientes					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	INT		NOT NULL	SI	AUTO_INCREMENT
	nombre	VARCHAR	30	NOT NULL	NO	
	apellido	VARCHAR	30	NULL	NO	
FK	tipo_identificacion	TINYINT		NOT NULL	NO	
	cuit_cuil_dni	VARCHAR	15	NOT NULL	NO	
FK	area_emp	TINYINT		NOT NULL	NO	

Tabla	provincias					
Desc	Listado de provincias donde radican clientes o proveedores					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	TINYINT		NOT NULL	SI	AUTO_INCREMENT
	provincias	VARCHAR	25	NOT NULL		

Tabla	direcciones					
Desc	Almacenamos direcciones de clientes y/o proveedores					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	INT		NOT NULL	SI	AUTO_INCREMENT
FK	id_cliente	INT		NULL		0
FK	id_empleado	INT		NULL		0
FK	id_proveedor	INT		NULL		0
FK	id_provincia	TINYINT		NOT NULL		
	pais	VARCHAR	20	NULL		ARGENTINA
	localidad	VARCHAR	30	NOT NULL		
	direccion	VARCHAR	30	NOT NULL		
	cod_postal	VARCHAR	10	NULL		
	direccion_principal	TINYINT		NULL		1

Tabla	email					
Desc	Generamos una tabla con email de nuestros clientes, proveedores y empleados					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	INT		NOT NULL	SI	AUTO_INCREMENT
FK	id_cliente	INT		NULL		
FK	id_empleado	INT		NULL		
FK	id_proveedor	INT		NULL		
	email	VARCHAR	100	NOT NULL		
	principal	TINYINT		NULL		1

Tabla	telefonos					
Desc	Generamos una tabla para almacenar el contacto de nuestros clientes, proveedores y empleados					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	INT		NOT NULL	SI	AUTO_INCREMENT
FK	id_cliente	INT		NULL		0
FK	id_empleado	INT		NULL		0
FK	id_proveedor	INT		NULL		0
	telefono	VARCHAR	15	NOT NULL		
	detalle	VARCHAR	20	NULL		
	principal	TINYINT		NULL		1

Tabla	facturacion					
Desc	tabla la cual almacenamos las facturas de las ventas realizadas					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	INT		NOT NULL	SI	AUTO_INCREMENT
FK	id_cliente	INT		NOT NULL		
	numbfactura	VARCHAR	15	NOT NULL		
	subtotal	DECIMAL	(10,2)	NOT NULL		
	iva	DECIMAL	(8,2)	NULL		0.00
	total	DECIMAL	(11,2)	NOT NULL		
	fecha	DATE		NOT NULL		

Tabla	categoria_productos					
Desc	categoria en la cual agrupamos productos con características similares					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	INT		NOT NULL	SI	AUTO_INCREMENT
	categoria	VARCHAR	25	NOT NULL		

Tabla	servicios					
Desc	Almacenamos listado de servicios ofrecidos y valores de los mismos					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	INT		NOT NULL	SI	AUTO_INCREMENT
	tipo_servicio	VARCHAR	35	NOT NULL		
	costo	DECIMAL	(7,2)	NOT NULL		

Tabla	productos					
Desc	Almacenamos los productos comercializados y sus valores					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	INT		NOT NULL	SI	AUTO_INCREMENT
FK	id_proveedor	INT		NOT NULL		
FK	id_categoria	INT		NOT NULL		
	nombre	VARCHAR	45	NOT NULL		
	presentacion	VARCHAR	45	NOT NULL		
	precio_unitario	DECIMAL	(10,2)	NOT NULL		
	stock	INT		NOT NULL		0

Tabla	ventas_productos					
Desc	Tabla donde ingresamos los productos vendidos por operación					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	INT		NOT NULL	SI	AUTO_INCREMENT
FK	id_factura	INT		NOT NULL		
FK	id_producto	INT		NOT NULL		
	cantidad	INT		NOT NULL		

Tabla	servicio_tecnico					
Desc	Almacenamos información de los servicios realizados por los tecnicos					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	INT		NOT NULL	SI	AUTO_INCREMENT
FK	id_empleado	INT		NOT NULL		
FK	id_cliente	INT		NOT NULL		
	detalle	VARCHAR	300	NOT NULL		
	recepción	DATETIME		NULL		CURRENT_TIMESTAMP
	estado	VARCHAR	15	NULL		pendiente'

Tabla	ervicios_facturados					
Desc	Almacenamos los servicios que han sido facturados					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	INT		NOT NULL	SI	AUTO_INCREMENT
FK	id_factura	INT		NOT NULL		
FK	id_servicio_tecnico	INT		NOT NULL		
FK	id_servicio	INT		NOT NULL		
	cantidad	INT		NOT NULL		

Tabla	mov_empleados					
Desc	Tabla donde llevaremos almacenados los cambios de areas de nuestros empleados					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	INT		NOT NULL	SI	AUTO_INCREMENT
FK	id_empleado	INT		NOT NULL		
FK	id_area_anterior	TINYINT		NOT NULL		
	fecha	TIMESTAMP		NOT NULL		CURRENT_TIMESTAMP

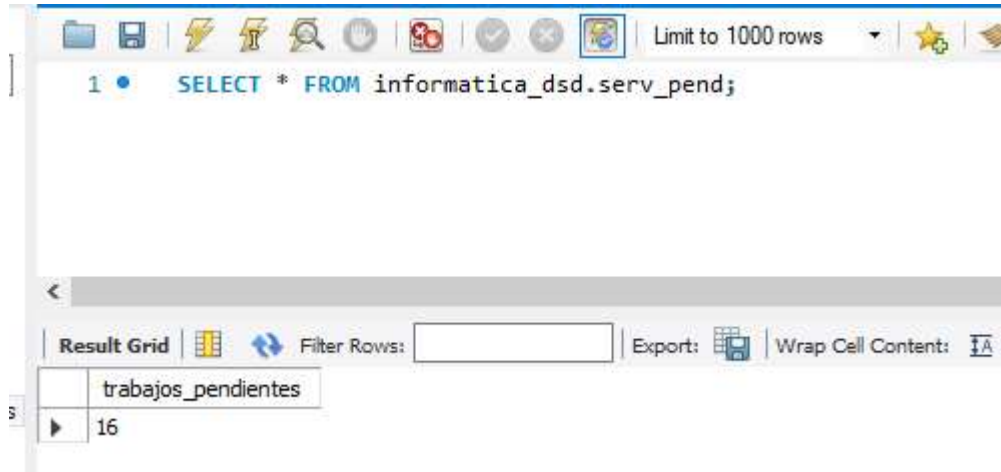
Tabla	ventas_productos_respaldo					
Desc	Tabla de respaldo que almacenara al ejecutarse un TRIGGER la cantidad de productos vendidos					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	INT		NOT NULL	SI	AUTO_INCREMENT
FK	id_factura	INT		NOT NULL		
FK	id_producto	INT		NOT NULL		
	cantidad	INT		NOT NULL		

Tabla	ante_val_prod					
Desc	Historial de precios anteriores de nuestros productos en caso de cambio de precios					
KEY	COLUMN	TYPE	LENGHT	NOT NULL	UNIQUE	DEFAULT
PK	id	INT		NOT NULL	SI	AUTO_INCREMENT
FK	id_producto	INT		NOT NULL		
	precio_ant	DECIMAL	(8,2)	NOT NULL		
	fecha	TIMESTAMP		NOT NULL		CURRENT_TIMESTAMP

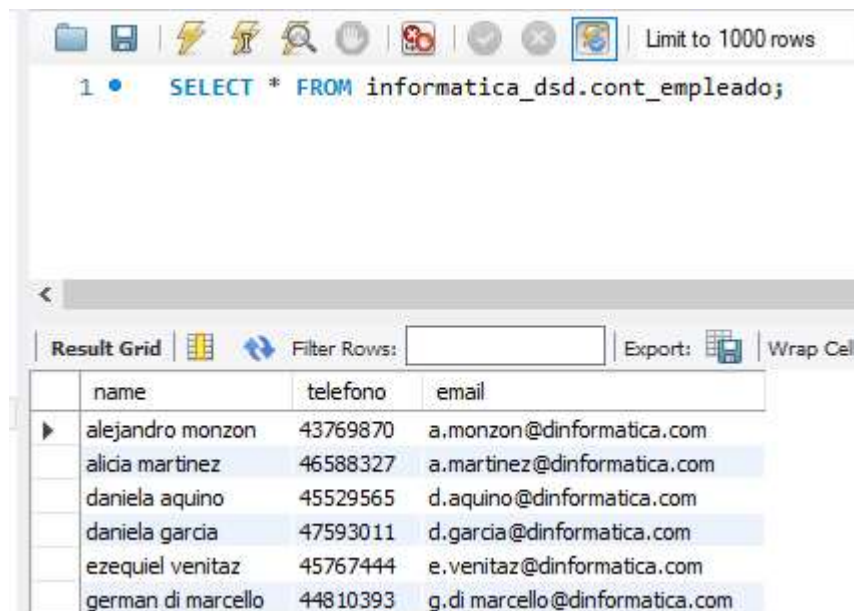
Vistas

Listado de vistas, descripción y ejemplo.

- `serv_pend`: Devolverá aquellos trabajos que aun estén pendientes de servicio técnico.



- `Cont_empleados`: Devolverá nombre y apellido del empleado junto a su teléfono y email.



- Prodxcat: Podremos observar la cantidad de productos por cada categoría.

```
1 SELECT * FROM informatica_dsd.prodxcat;
```

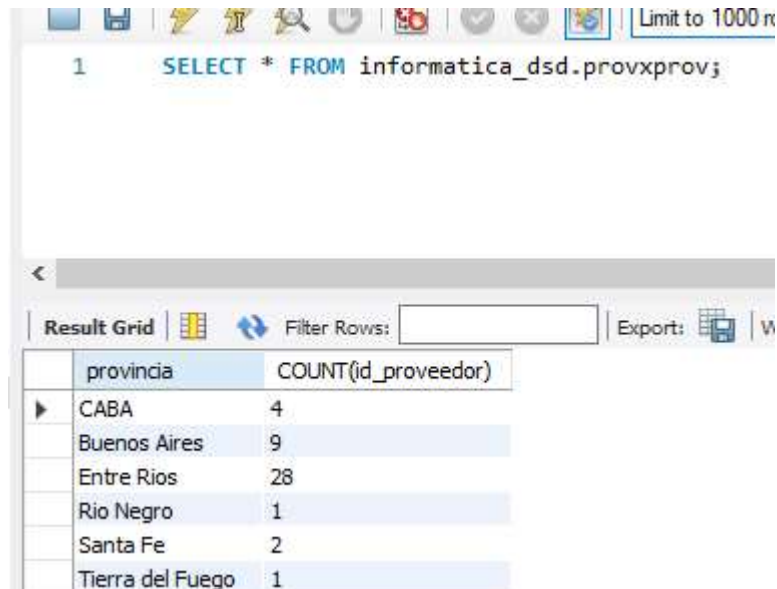
categoria	productos
Equipos y Notebook	4
Procesadores	3
Mothers	4
Placas de Video	3
Memorias RAM	4
Almacenamiento	2

- Cant_vend: Observaremos las cantidades de productos vendidos a la fecha.

```
1 SELECT * FROM informatica_dsd.cant_vend;
```

producto	cantidades
2	3
3	7
4	6
6	4
7	2
8	1

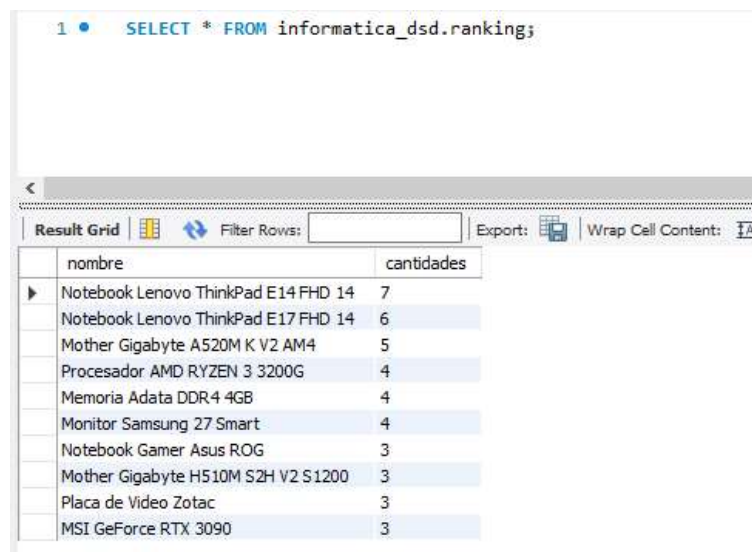
- Provxprov: obtendremos la cantidad de proveedores por provincia.



```
1 SELECT * FROM informatica_dsd.provxprov;
```

provincia	COUNT(id_proveedor)
CABA	4
Buenos Aires	9
Entre Rios	28
Rio Negro	1
Santa Fe	2
Tierra del Fuego	1

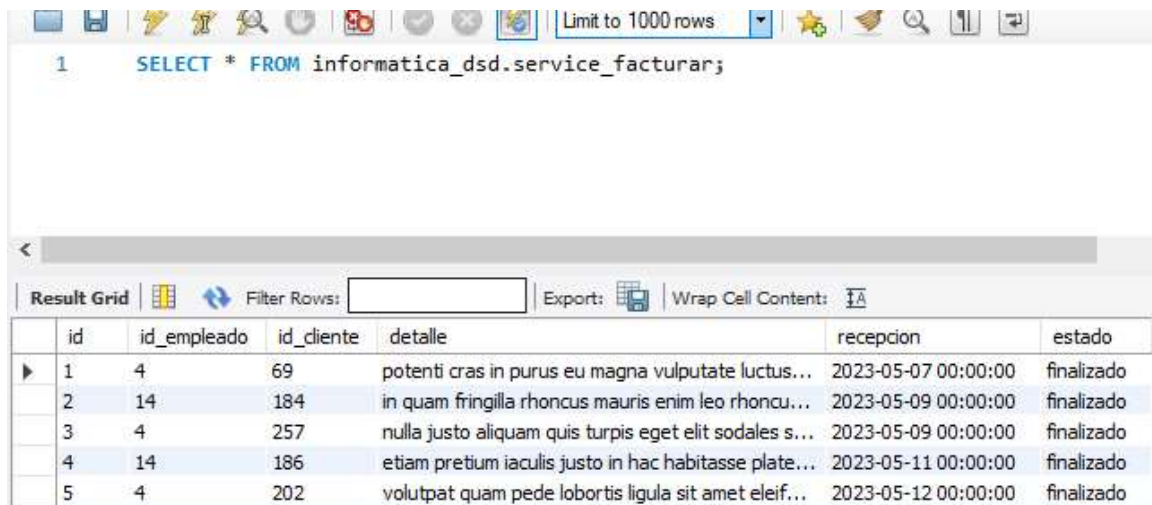
- Ranking: Mostrara el top de 10 productos más vendidos a la fecha.



```
1 SELECT * FROM informatica_dsd.ranking;
```

nombre	cantidades
Notebook Lenovo ThinkPad E14 FHD 14	7
Notebook Lenovo ThinkPad E17 FHD 14	6
Mother Gigabyte A520M K V2 AM4	5
Procesador AMD RYZEN 3 3200G	4
Memoria Adata DDR4 4GB	4
Monitor Samsung 27 Smart	4
Notebook Gamer Asus ROG	3
Mother Gigabyte H510M S2H V2 S1200	3
Placa de Video Zotac	3
MSI GeForce RTX 3090	3

- Service_facturar: Listado de servicios técnicos finalizados que pueden ser facturados.

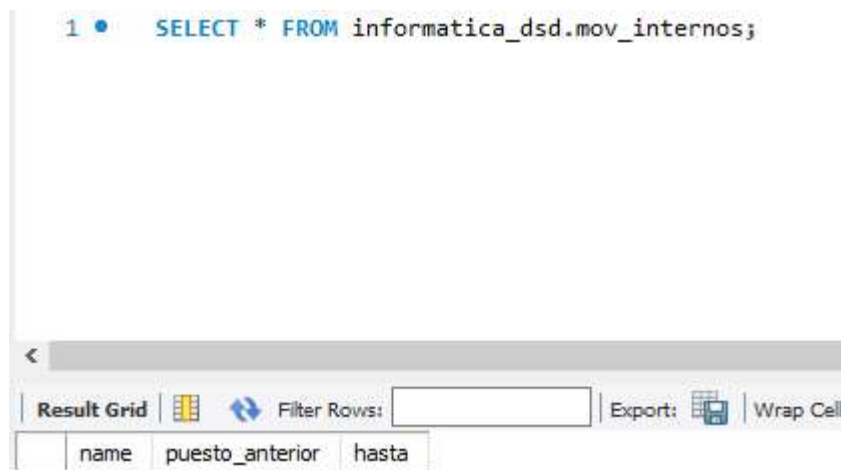


1 `SELECT * FROM informatica_dsd.service_facturar;`

Result Grid | Filter Rows: | Export: | Wrap Cell Content: ☐

	id	id_empleado	id_cliente	detalle	repcion	estado
▶	1	4	69	potenti cras in purus eu magna vulputate luctus...	2023-05-07 00:00:00	finalizado
	2	14	184	in quam fringilla rhoncus mauris enim leo rhoncu...	2023-05-09 00:00:00	finalizado
	3	4	257	nulla justo aliquam quis turpis eget elit sodales s...	2023-05-09 00:00:00	finalizado
	4	14	186	etiam pretium iaculis justo in hac habitasse plate...	2023-05-11 00:00:00	finalizado
	5	4	202	volutpat quam pede lobortis ligula sit amet eleif...	2023-05-12 00:00:00	finalizado

- Mov_internos: ejecutado el TRIGGER **cambios_de_area** al llamar la sentencia **UPDATE** esta vista devolverá un listado con los campos: nombre, puesto_anterior y fecha hasta la que perteneció a esa área.



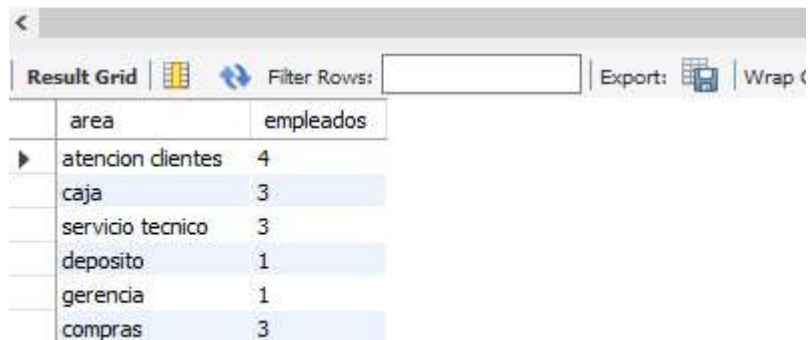
1 • `SELECT * FROM informatica_dsd.mov_internos;`

Result Grid | Filter Rows: | Export: | Wrap Cell

	name	puesto_anterior	hasta
--	------	-----------------	-------

- cupo_sector: podremos visualizar la cantidad de empleados en cada sector de la empresa.

```
1 • SELECT * FROM informatica_dsd.cupo_sector;
```



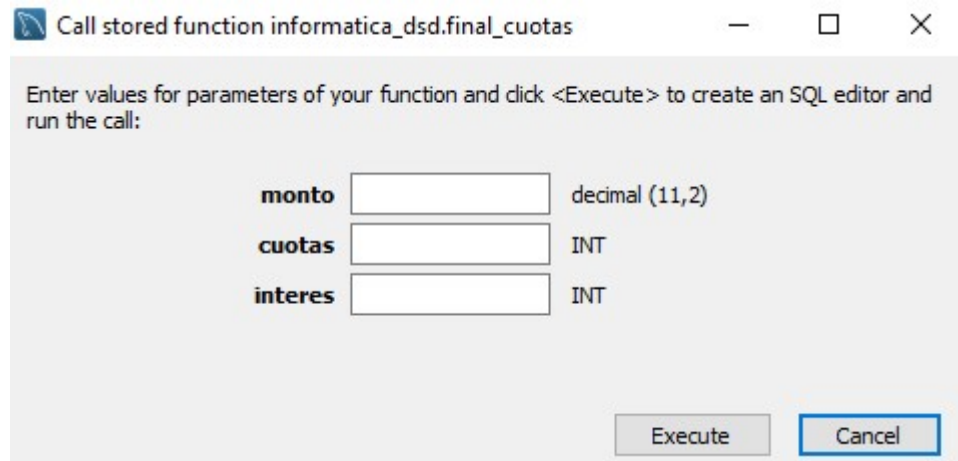
The screenshot shows a database query result grid. At the top, there is a toolbar with a back arrow, a 'Result Grid' tab, a grid icon, a 'Filter Rows:' input field, an 'Export:' button with a grid icon, and a 'Wrap' button. Below the toolbar is a table with two columns: 'area' and 'empleados'. The table contains six rows of data, with the first row expanded to show a sub-row.

area	empleados
atencion clientes	4
caja	3
servicio tecnico	3
deposito	1
gerencia	1
compras	3

Funciones

- **Final_cuotas:** Llamando a dicha función podremos calcular en cuanto quedaría las cuotas en caso de financiar la compra, más el interés agregado.

Datos de entrada: **monto, cuotas e interes**



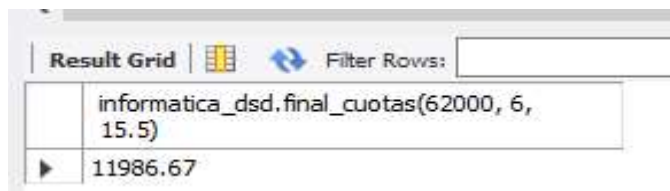
Call stored function informatica_dsd.final_cuotas

Enter values for parameters of your function and click <Execute> to create an SQL editor and run the call:

monto	<input type="text"/>	decimal (11,2)
cuotas	<input type="text"/>	INT
interes	<input type="text"/>	INT

Execute Cancel

Ejemplo:



Result Grid		Filter Rows:
	informatica_dsd.final_cuotas(62000, 6, 15.5)	
▶	11986.67	

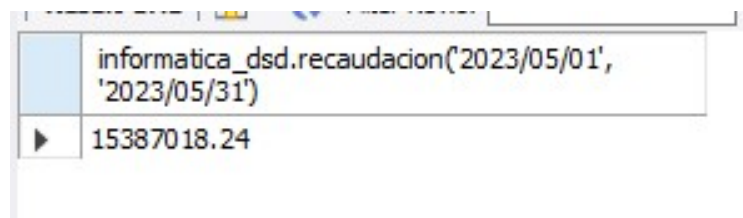
- Recaudacion: Devolverá la suma del importe de ventas entre fechas ingresadas.

Datos de entrada: ***desde (fecha inicial), hasta (fecha final)***.



The screenshot shows a dialog box titled 'Call stored function informatica_dsd.recaudacion'. It contains a text area with the instruction: 'Enter values for parameters of your function and click <Execute> to create an SQL editor and run the call:'. Below this, there are two input fields. The first is labeled 'desde' and has a 'DATE' type indicator. The second is labeled 'hasta' and also has a 'DATE' type indicator. At the bottom right, there are two buttons: 'Execute' and 'Cancel'.

Ejemplo:



	informatica_dsd.recaudacion('2023/05/01', '2023/05/31')
▶	15387018.24

Store Procedures

- Dsd_ordenar: Indicamos a través de un parámetro el campo de ordenamiento de la tabla clientes y por medio del otro parámetro si es de forma ascendente o descendente.

Datos de entrada: **campo, ordenamiento (ascendente / descendente).**

Call stored procedure informatica_dsd.dsd_ordenar

Enter values for parameters of your procedure and click <Execute> to create an SQL editor and run the call:

campo [IN] CHAR(30)

ordenamiento [IN] CHAR(30)

Execute Cancel

Ejemplo:

```
1 • call informatica_dsd.dsd_ordenar('apellido', 'descendente');  
2
```

result Grid					
Filter Rows: <input type="text"/>					
Export: Wrap Cell Content:					
id	id_condicion	nombre	apellido	tipo_identificacion	cuit_cuil_dni
236	3	izabel	zorzoni	3	31538007
250	3	armin	youings	3	31725264
291	2	filmore	wurst	1	20292046468
120	2	verene	worman	1	27294530201
229	3	chris	wolfendale	3	32743136
58	2	noel	wittrington	1	27325701363

- **Actualiza_estado:** Por medio del procedimiento podremos modificar el campo estado de un registro en la tabla “servicio_tecnico”, informando si está finalizado, por ejemplo.

Datos de entrada: **numb_id (id), situacion.**

Call stored procedure informatica_dsd.actualiza_estado

Enter values for parameters of your procedure and click <Execute> to create an SQL editor and run the call:

numb_id [IN] INT

situacion [IN] CHAR(20)

Execute Cancel

Ejemplo:

1. Ejecutamos query.

```
1 • SELECT *
2 FROM servicio_tecnico
3 WHERE id = 8;
```

id	id_empleado	id_cliente	detalle	repcion	estado
8	14	96	accumsan tellus nisi eu orci mauris lacinia sapien...	2023-05-29 00:00:00	NULL
NULL	NULL	NULL	NULL	NULL	NULL

2. Ejecutamos Procedure e ingresamos valores.

Call stored procedure informatica_dsd.actualiza_estado

Enter values for parameters of your procedure and click <Execute> to create an SQL editor and run the call:

numb_id [IN] INT

situacion [IN] CHAR(20)

Execute **Cancel**

3. Volvemos a ejecutar la sentencia del paso 1 y observamos la actualización del campo 'estado'.

```

1 • SELECT *
2 FROM servicio_tecnico
3 WHERE id = 8;

```

	id	id_empleado	id_cliente	detalle	repcion	estado
▶	8	14	96	accumsan tellus nisi eu orci mauris lacinia sapien...	2023-05-29 00:00:00	Finalizado
*	NULL	NULL	NULL	NULL	NULL	NULL

Triggers

- **Cambios_de_area:** Este disparador se acciona una vez actualizado el campo “area_emp” en la tabla empleados con el fin de llevar un registro en la tabla “mov_empleados” de puestos anteriores del mismo.
 - ◆ Tipo: **BEFORE**
 - ◆ Monitorea: **UPDATE**
 - ◆ Tabla: **empleados**
- **Control_stock:** Al vender un producto se descontará en el stock de dicho producto en la tabla productos.
 - ◆ Tipo: **BEFORE**
 - ◆ Monitorea: **INSERT**
 - ◆ Tabla: **ventas_productos**
- **Respaldo_ventasproduc:** Este disparador cumple la función de realizar un respaldo en la tabla “Ventas_productos_respaldo” en caso de borrar un registro de ventas.
 - ◆ Tipo: **BEFORE**
 - ◆ Monitorea: **DELETE**
 - ◆ Tabla: **ventas_productos**
- **Valor_anterior:** En caso de actualizaciones en los precios de venta de un producto guardaremos por medio del disparador valores anteriores en la tabla ante_val_prod.
 - ◆ Tipo: **BEFORE**
 - ◆ Monitorea: **UPDATE**
 - ◆ Tabla: **productos**

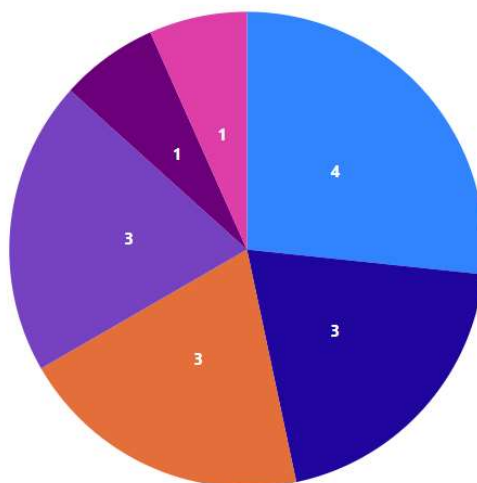
Inteligencia Empresarial (BI)

Informes gráficos derivados de las vistas **ranking** y **cupo_sector** que proporcionan representaciones visuales de los datos obtenidos.



Empleados por sector

sector ● atencion clientes ● caja ● compras ● servicio tecnico ● deposito ● gerencia



Tecnologías utilizadas

Sistema	Producto	Versión
DB	MySQL community server GPL	8.0.34
BI	Microsoft Power BI Desktop	2.116.966.0 64-bit

Link repositorio en github del archivo .sql

https://github.com/demiandg/tienda_de_informatica.git