


Laboratorio de Computación III

Docentes

- Angel Simón - Profesor
asimon@docentes.frgp.utn.edu.ar
- Laura Vélez - Jefe de trabajos prácticos
lvelez@docentes.frgp.utn.edu.ar
- Gonzalo Ligeró - Ayudante de trabajos prácticos
gonzalo.ligeró@alumnos.frgp.utn.edu.ar
- Lucas Saputo - Ayudante de trabajos prácticos
lucas.saputo@alumnos.frgp.utn.edu.ar

Contenidos

- Normalización de Base de Datos
 - Conceptos fundamentales del Modelo Relacional
 - Consultas de Acción
 - Consultas de Selección
 - Vistas
 - Procedimientos almacenados
 - Transacciones
 - Triggers
- 
- A large, solid blue shape that starts from the bottom left and extends diagonally upwards to the right, filling the bottom right corner of the slide.

Evaluaciones

Modalidad de evaluación sumativa



Primer parcial

- Lenguaje T-SQL
- Individual



Segundo parcial

- Lenguaje T-SQL
- Individual

Evaluaciones

Modalidad de evaluación por proyecto (Hasta 10 grupos)



Primer parcial

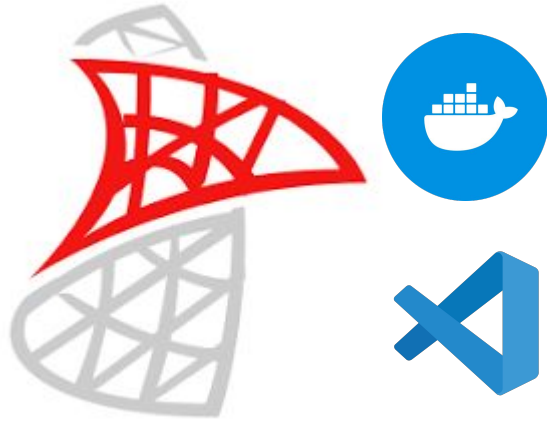
- Presentación y defensa del proyecto
- Grupal con nota individual



Segundo parcial

- Presentación y defensa del proyecto terminado
- Grupal con nota individual

Software



GitHub



Servidor: SQL Server Express

Cliente: SQL Server Management Studio Express / Azure Data Studio

Alternativa:

Servidor: Contenedor de SQL Server Express en Docker

Cliente: DBeaver, Visual Studio Code + SQL Server plugin

De archivos de datos a bases de datos

```
class Empleado{  
    private:  
        int ID;  
        Fecha nacimiento;  
        char apellidos[50];  
        char nombres[50];  
        char genero;  
        float sueldo;  
    public:  
        // setters  
        // getters  
};
```

```
class EmpleadoArchivo{  
    public:  
        static bool agregar(Empleado);  
        static bool modificar(Empleado);  
        static bool eliminar(Empleado);  
        static Empleado buscar(int);  
};
```



De archivos de datos a bases de datos

```
CREATE TABLE empleados(  
    id bigint not null primary key,  
    nacimiento date null,  
    apellidos varchar(50) not null,  
    nombres varchar(50) not null,  
    genero char null,  
    sueldo money not null check (sueldo > 0)  
)
```



De archivos de datos a bases de datos:

Tipos de dato en C++

- **Texto**
- **Números enteros**
- **Números reales**
- **Fechas**

De archivos de datos a bases de datos: Tipos de dato

Texto

C/C++



char



string

T-SQL



char



nchar



varchar



nvarchar

De archivos de datos a bases de datos: Tipos de dato

Números enteros

C/C++

 **bool**

 **short**

 **int**

 **long**

T-SQL

 **bit**

 **tinyint**

 **smallint**

 **int**

 **bigint**

De archivos de datos a bases de datos: Tipos de dato

Números reales

C/C++



float



double

T-SQL



decimal



money



smallmoney



float

De archivos de datos a bases de datos: Tipos de dato

Fechas

C/C++



struct tm



time_t

T-SQL



time



date



smalldatetime



datetime

Bases de datos: Tipos

Relacionales

■ **Microsoft SQL Server**

■ **Oracle**

■ **PostgreSQL**

■ **MySQL**

■ **MariaDB**

NoSQL

■ **MongoDB**

■ **Firebase Realtime Database**

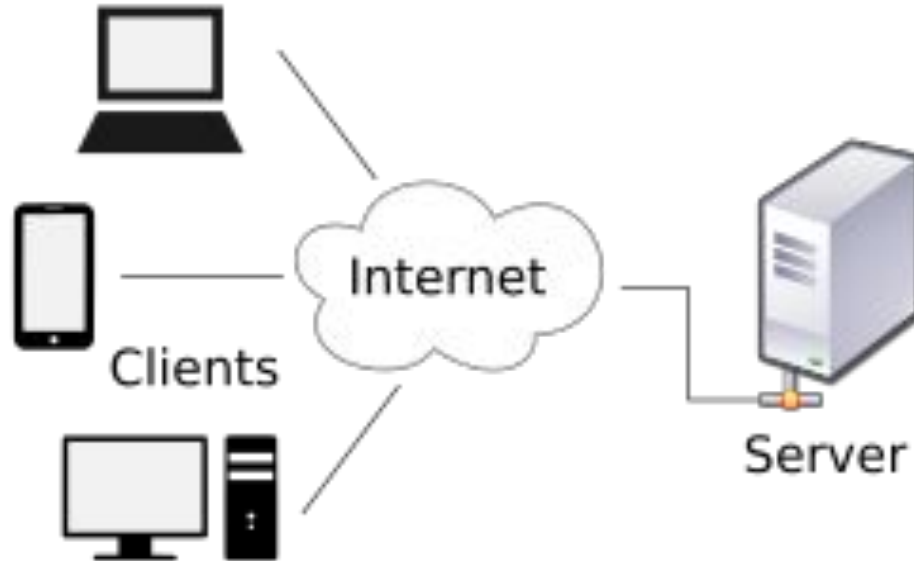
■ **Redis**

■ **Cassandra**

■ **CouchDB**

■ **DynamoDB**

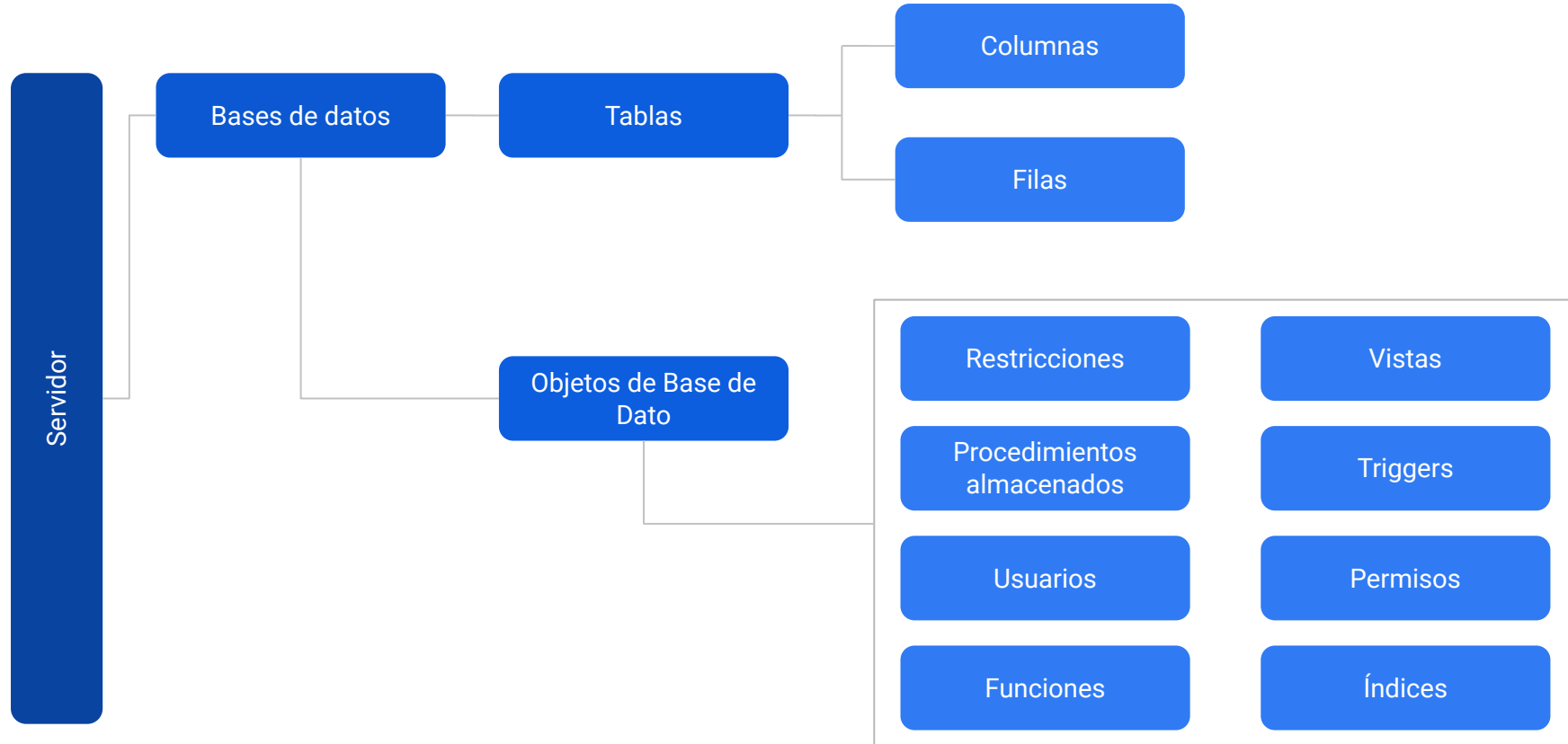
Bases de datos: Arq. Cliente Servidor



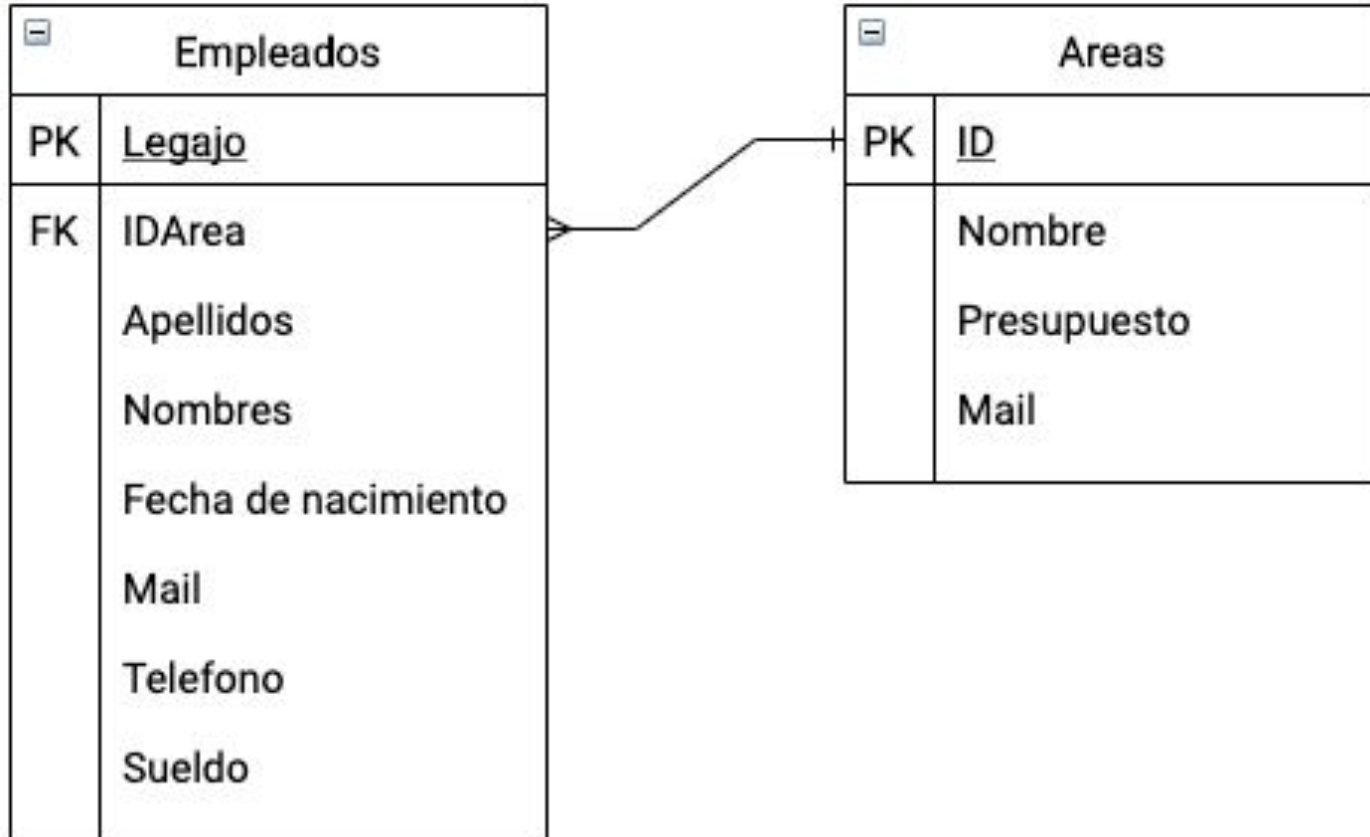
Ejemplos

Ejemplo	Cliente	Servidor
Correo electrónico	Microsoft Outlook, Mozilla Thunderbird, Gmail App	Servidor SMTP, POP3, IMAP
Servicios web	Navegador web, aplicaciones móviles, etc	Servidor HTTP, HTTPS
Bases de datos	Azure Data Studio, pgAdmin, MySQL Workbench	TCP/IP
Juegos en línea	Steam, Epic, Gog	TCP/IP, UDP
Chats	WhatsApp, Telegram, Facebook Messenger, Slack	HTTP, HTTPS Websockets
Escritorio remoto	Cliente de VNC	Servidor de VNC

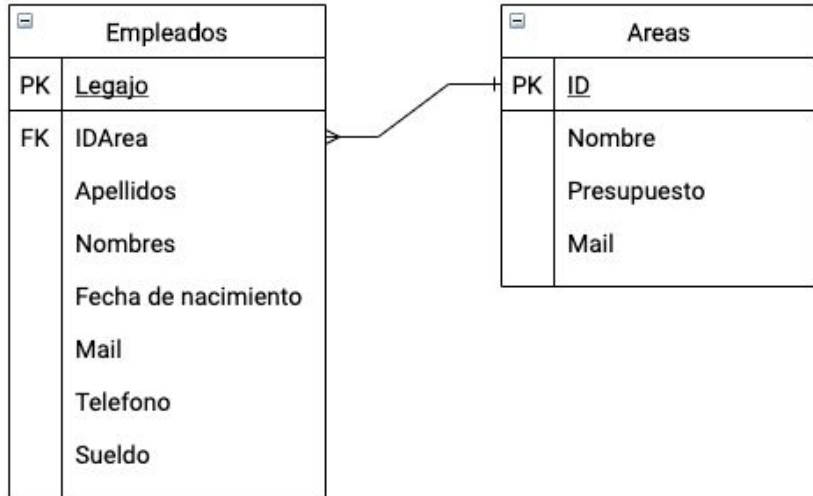
Bases de datos: Elementos de bases de datos



Ejemplo de BD: Diagrama de Entidad Relación



Ejemplo de Base de Datos



Restricciones:

- El legajo debe ser autonumérico.
 - El IDArea del empleado debe existir en Áreas.
 - La fecha de nacimiento debe ser menor a la actual.
 - El mail no puede repetirse.
 - El teléfono es opcional.
 - El Sueldo debe ser mayor a \$0.
-
- El ID debe ser autonumérico.
 - El presupuesto debe ser mayor a \$0.
 - El mail debe ser único.

Restricción

Una restricción es una condición impuesta a una columna o conjunto de columnas para asegurar la integridad de la información.

 Clave primaria / única

 Clave foránea

 Check

 Not Null

 Default

Modelo relacional: Clave primaria/única

Restricción que no permite valores duplicados en la/s columna/s definida como clave.

Legajo	Apellidos	Nombres
1000	Simón	Angel
2000	Pérez	Juan
2000	Rodríguez	Sabrina

Modelo relacional: Clave primaria/única

Restricción que no permite valores duplicados en la/s columna/s definida como clave.

Legajo	IDSucursal	Apellidos	Nombres
1000	100	Simón	Angel
1000	200	Lara	Brian
2000	200	González	Juan
1000	200	Rodríguez	Sabrina

Modelo relacional: Clave foránea

Restricción que no permite valores en la/s columna/s definidas como clave que no existan en la/s columna/s definidas como referentes.

Legajo	Apellido	IDCarrera
1000	Simón	1
2000	Pérez	2
3000	Rodríguez	10

ID	Nombre	Nivel
1	Técnico Universitario en Programación	Tecnicatura
2	Ingeniería mecánica	Grado
3	Maestría en seguridad informática	Posgrado

Modelo relacional: Nulidad

La capacidad de indicar que un atributo no posee un valor.

Legajo	Apellido y nombres	Sexo	Fecha graduación
1000	Pérez, Juan	M	10/3/2019
2000	Lux, Romina	F	11/3/2019
3000	López, Carla	null	25/3/2019
4000	Ruiz, Damián	null	null



Modelo relacional: Check

Regla de validación que posee una columna y que debe cumplirse.

Ejemplos:

- Una fecha debe ser menor a la fecha de hoy.
- Un importe debe ser mayor a cero.
- Un número debe estar entre 1 y 10.
- Un carácter sólo puede contener E, C, T
- El largo de una cadena debe ser mayor a 5.

NOTA: Si una columna acepta valores nulos y posee un check. Entonces el valor nulo es válido ante la validación del check.

Codificación de Base de Datos

