

PRÁCTICA AD01

INTRODUCCION ACCESO A DATOS

26/09/2022

CÉSAR BOUZAS SOTO

Contenido

1.	Investigación de los principales gestores de BD.	2
1.0	¿Que tipos de bases de datos existen?	2
1.0.1	Según su flexibilidad.....	2
1.0.2	Según la forma de su organización.	2
1.1	¿Qué son los procedimientos Almacenados?	6
1.2	¿Qué son las transacciones?	6
1.3	Tabla resumen Sistemas gestores:	7
2	Instalación de software.....	8
2.1	Instalación de Netbeans 14.....	8
2.2	2.-Instalar varios gestores de BD.....	8
2.2.1	Dbeaver.	8
2.2.2	Mysql.....	9
2.2.3	Heidi	10
2.2.4	XAMPP+mariaDB.....	12

1. Investigación de los principales gestores de BD.

1.0 ¿Que tipos de bases de datos existen?

1.0.1 Según su flexibilidad.

1.0.1.1 Estáticas.

Son bases de datos de consulta cuyos datos no pueden modificarse.

1.0.1.2 Dinámicas.

Son aquellas donde los datos pueden actualizarse o incluso modificarse. La mayoría puede ser actualizada en tiempo real.

1.0.2 Según la forma de su organización.

1.0.2.1 Jerárquicas.

Las bases de datos jerárquicas son aquellas organizadas en forma de un árbol al revés. Almacenan la información en forma de registros dentro de una estructura jerárquica, es de aquí que proviene su nombre.

Cada registro de este "árbol" es llamado nodo. Nodos son registros que contienen alguna información de interés y a partir del nodo raíz son enlazados los otros nodos descendientes: padres e hijos. Cada nodo padre puede tener varios nodos hijos, pero cada nodo hijo solo puede tener un solo nodo padre.

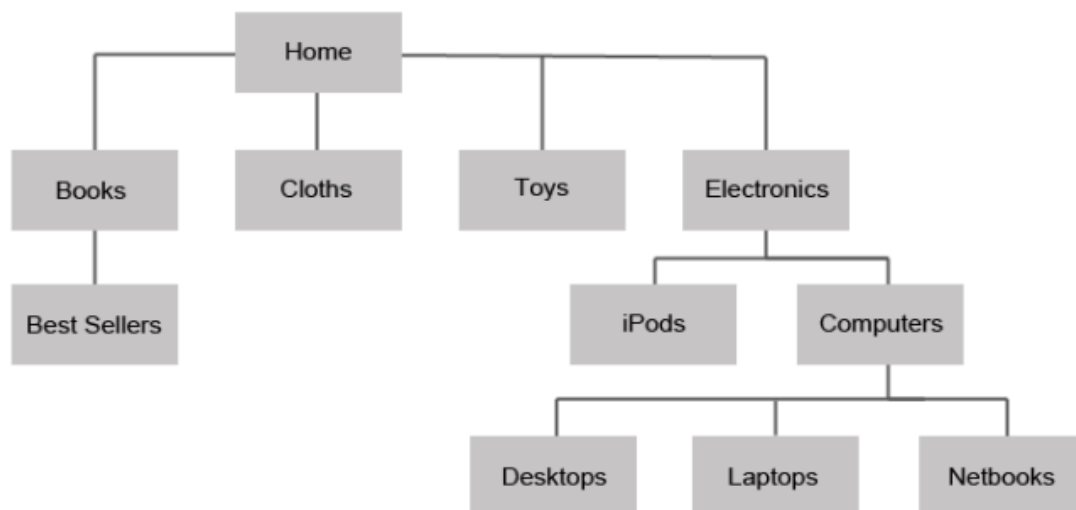


Ilustración 1 Bd Jerárquica.

1.0.2.2 Red.

Esta base de datos es una variación de la anterior. La diferencia está en que en la base de datos jerárquica un nodo hijo no puede tener varios padres y aquí sí.

Las características de estas bases de datos son semejantes a las de las bases de datos jerárquicas, aunque estas son mucho más potentes y complejas.

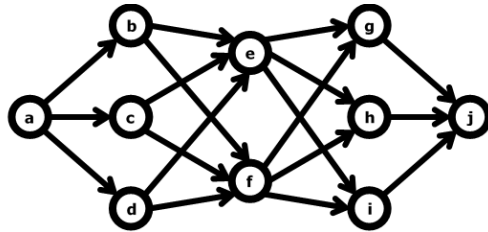


Ilustración 2 Bd en Red.

1.0.2.3 Relacionales.

Las bases de datos relacionales son las más usadas actualmente para administrar datos de forma dinámica. Permite crear todo tipo de datos y relacionarlos entre sí.

Los datos son almacenados en registros que son organizados en tablas, de esta forma pueden asociarse los elementos entre sí muy fácilmente, además se pueden cruzar sin ninguna dificultad.

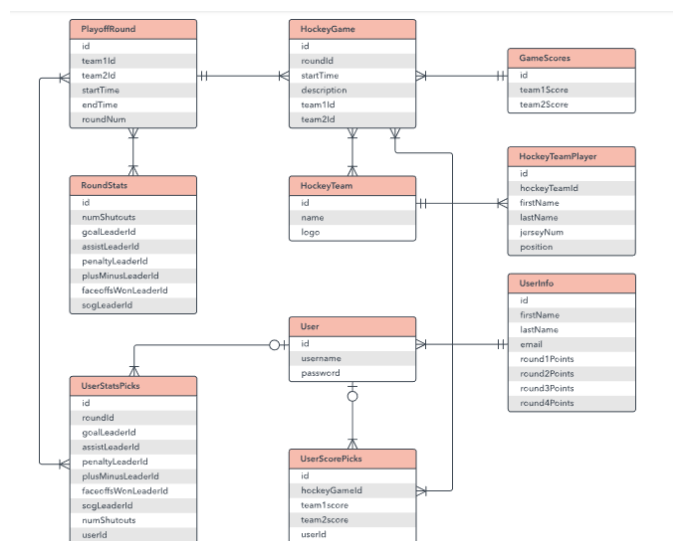


Ilustración 3 Bd Relacional

1.0.2.4 Deductivas.

Conocidas también como bases de datos lógicas. Se utilizan generalmente en buscadores, pero pueden usarse de otras formas.

Permiten almacenar los datos y consultarlos a través de búsquedas que utilizan reglas y normas previamente almacenadas.

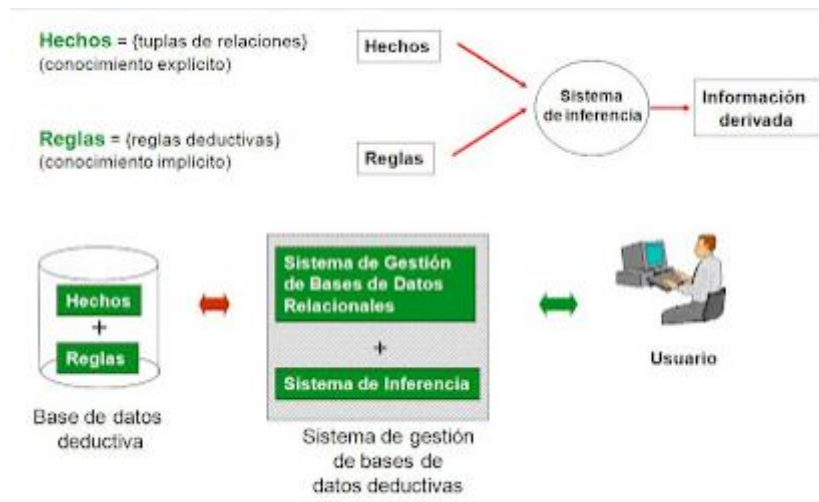


Ilustración 4 Bd deductivas

1.0.2.5 Multidimensionales.

Estas bases de datos utilizan conceptualmente la idea de un cubo de datos. Donde las informaciones se almacenan en la intersección de tres o más atributos. Esta concepción puede ser algo compleja pero su uso es bastante simple.

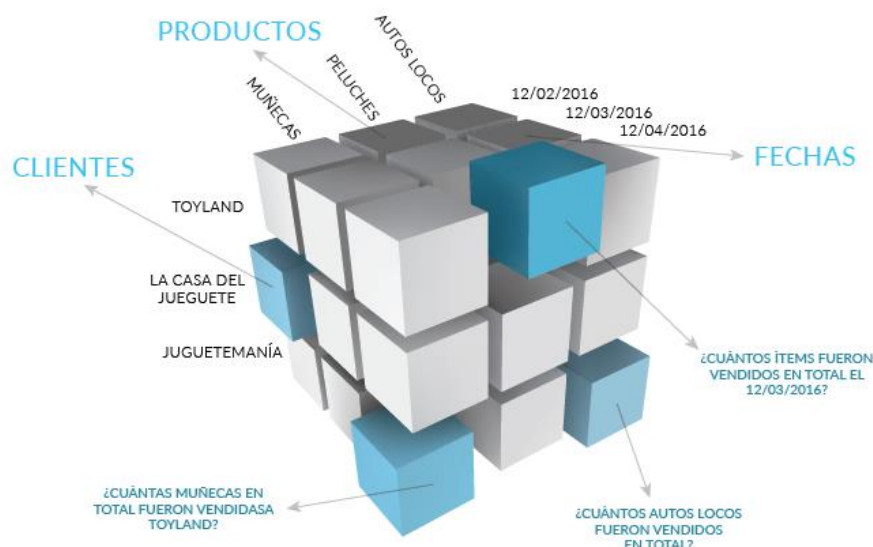


Ilustración 5 Bd multidimensionales.

1.0.2.6 Orientada a Objetos.

Una base de datos orientada a objetos almacena datos complejos y relaciones entre datos directamente, sin asignar filas y columnas, y esto hace que sean más adecuadas para aplicaciones que tratan con datos muy complejos. Los objetos tienen relaciones “muchos a muchos” y son accesibles mediante el uso de punteros.

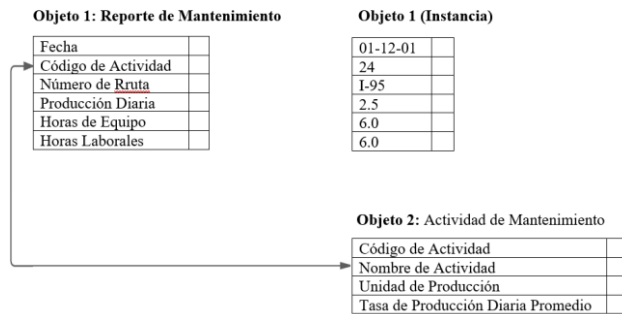


Ilustración 6 Bd Orientada a Objetos

1.0.2.7 No relacional orientada a Documentos.

Una base de datos de documentos es un tipo de base de datos no relacional que ha sido diseñada para almacenar y consultar datos como documentos de tipo JSON. Las bases de datos de documentos facilitan a los desarrolladores el almacenamiento y la consulta de datos en una base de datos mediante el mismo formato de modelo de documentos que emplean en el código de aplicación. La naturaleza flexible, semiestructurada y jerárquica de los documentos y las bases de datos de documentos permite que evolucionen según las necesidades de las aplicaciones. El modelo de documentos funciona bien con casos de uso como catálogos, perfiles de usuario y sistemas de administración de contenido en los que cada documento es único y evoluciona con el tiempo. Las bases de datos de documentos permiten una indexación fácil, potentes consultas ad hoc y análisis de colecciones de documentos.

```

1  [
2    {
3      "year" : 2013,
4      "title" : "Turn It Down, Or Else!",
5      "info" : {
6        "directors" : [ "Alice Smith", "Bob Jones"],
7        "release_date" : "2013-01-18T00:00:00Z",
8        "rating" : 6.2,
9        "genres" : ["Comedy", "Drama"],
10       "image_url" : "http://ia.media-imdb.com/images/N/09ERNAU7F5797A37LU8HN09AMUP908RLlo5JF90ENR7LJKQ7@@_V1_SX400_.jpg",
11       "plot" : "A rock band plays their music at high volumes, annoying the neighbors.",
12       "actors" : ["David Mattheuman", "Jonathan G. Neff"]
13     }
14   },
15   {
16     "year": 2015,
17     "title": "The Big New Movie",
18     "info": {
19       "plot": "Nothing happens at all.",
20       "rating": 0
21     }
22   }
23 ]
  
```

Ilustración 7 Bd orientada a Documentos

1.1 ¿Qué son los procedimientos Almacenados?

Un procedimiento almacenado (stored procedure en inglés) es un programa (o procedimiento) almacenado físicamente en una base de datos. Su implementación varía de un gestor de bases de datos a otro. La ventaja de un procedimiento almacenado es que al ser ejecutado, en respuesta a una petición de usuario, es ejecutado directamente en el motor de bases de datos, el cual usualmente corre en un servidor separado. Como tal, posee acceso directo a los datos que necesita manipular y sólo necesita enviar sus resultados de regreso al usuario, deshaciéndose de la sobrecarga resultante de comunicar grandes cantidades de datos salientes y entrantes.

Los procedimientos pueden ser ventajosos: cuando una base de datos es manipulada desde muchos programas externos. Al incluir la lógica de la aplicación en la base de datos utilizando procedimientos almacenados, la necesidad de embeber la misma lógica en todos los programas que acceden a los datos es reducida. Esto puede simplificar la creación y, particularmente, el mantenimiento de los programas involucrados.

Podemos ver un claro ejemplo de estos procedimientos cuando requerimos realizar una misma operación en un servidor dentro de algunas o todas las bases de datos y a la vez dentro de todas o algunas de las tablas de las bases de datos del mismo.

Los usos 'típicos' de los procedimientos almacenados se aplican en la validación de datos, integrados dentro de la estructura del banco de datos. Los procedimientos almacenados usados con tal propósito se llaman comúnmente disparadores, o triggers. Otro uso común es la 'encapsulación' de un API para un proceso complejo o grande que podría requerir la 'ejecución' de varias consultas SQL, tales como la manipulación de un conjunto de datos enorme para producir un resultado resumido

1.2 ¿Qué son las transacciones?

Una transacción en un sistema de gestión de bases de datos es un conjunto de órdenes que se ejecutan formando una unidad de trabajo, es decir, en forma indivisible o atómica.

Un SGBD se dice transaccional si es capaz de mantener la integridad de datos, haciendo que estas transacciones no puedan finalizar en un estado intermedio. Cuando por alguna causa el sistema debe cancelar la transacción, empieza a deshacer las órdenes ejecutadas hasta dejar la base de datos en su estado inicial (llamado punto de integridad), como si la orden de la transacción nunca se hubiese realizado. Una transacción debe contar con ACID (un acrónimo inglés) que quiere decir: Atomicidad, consistencia, aislamiento y durabilidad.

Para esto, el lenguaje de consulta de datos provee los mecanismos para especificar que un conjunto de acciones deben constituir una transacción.

- **BEGIN TRANSACTION:** Especifica que va a empezar una transacción.
- **COMMIT TRANSACTION:** Le indica al motor que puede considerar la transacción completada con éxito.
- **ROLLBACK TRANSACTION:** Indica que se ha alcanzado un fallo y que debe restablecer la base al punto de integridad.

1.3 Tabla resumen Sistemas gestores:

Sistema Gestor	Url	Tipo	Lenguajes	Procedimientos Almacenados	Transacciones	Multiplataforma
MySQL	https://www.mysql.com	Relacional	SQL	si	si	si
SqlServer	https://www.microsoft.com/es-es/sql-server	Relacional	SQL	si	si	Windows /Linux
Oracle	https://www.oracle.com/es/downloads	Relacional	PL/SQL	si	si	si
Microsoft Access	https://support.microsoft.com	Relacional	SQL	si	si	Windows
PostgreSQL	https://www.postgresql.org	Relacional	SQL	si	si	si
Informix	https://www.ibm.com/es-es/products/informix	hibrido	SQL/no SQL	si	si	Windows /Linux
SyBASE	https://www.sap.com	Relacional	SQL	si	si	Multiplataforma
FireBird	https://firebirdsql.org	Relacional	SQL	si	si	Multiplataforma
DB2	https://www.ibm.com/es-es/	Relacional	SQL	si	si	Windows /Linux

2 Instalación de software

2.1 Instalación de Netbeans 14.

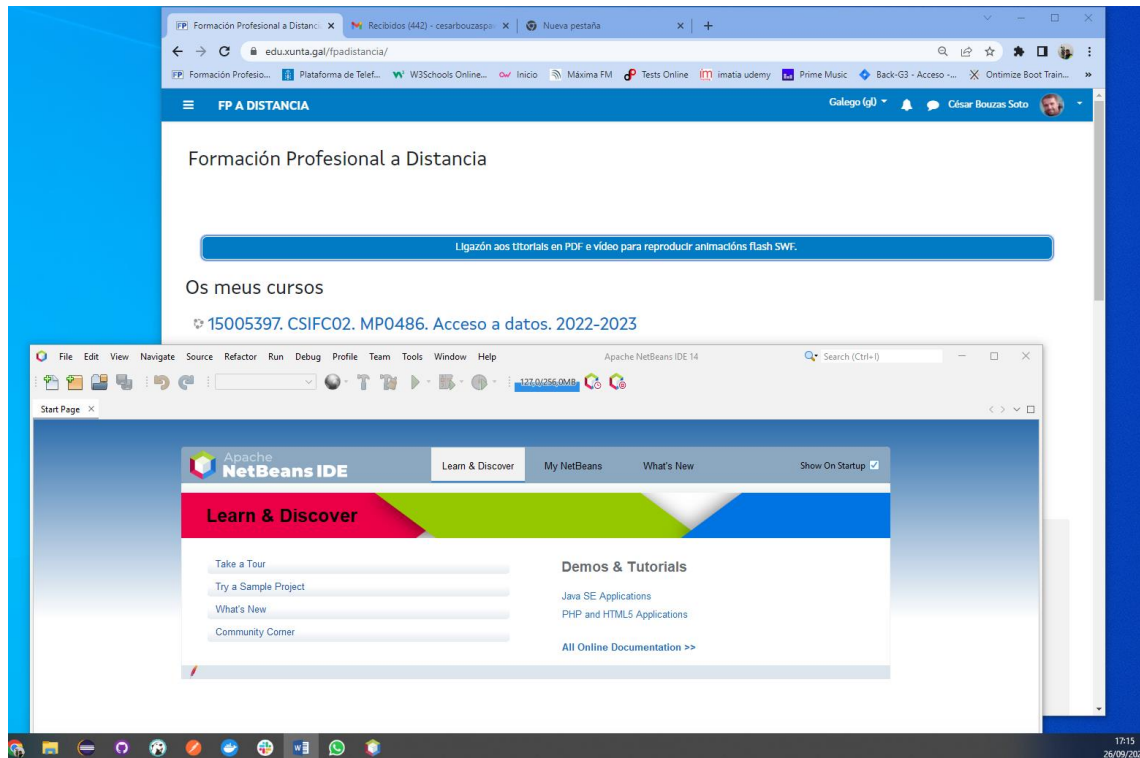


Ilustración 8 Netbeans 14 ejecutándose

2.2 2.-Instalar varios gestores de BD.

2.2.1 Dbeaver.

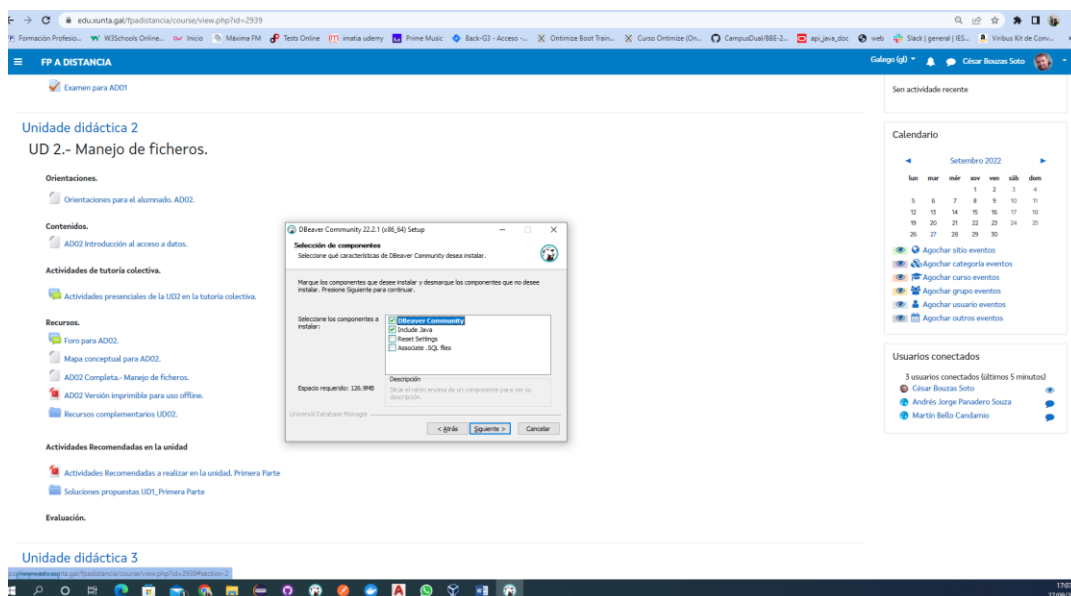
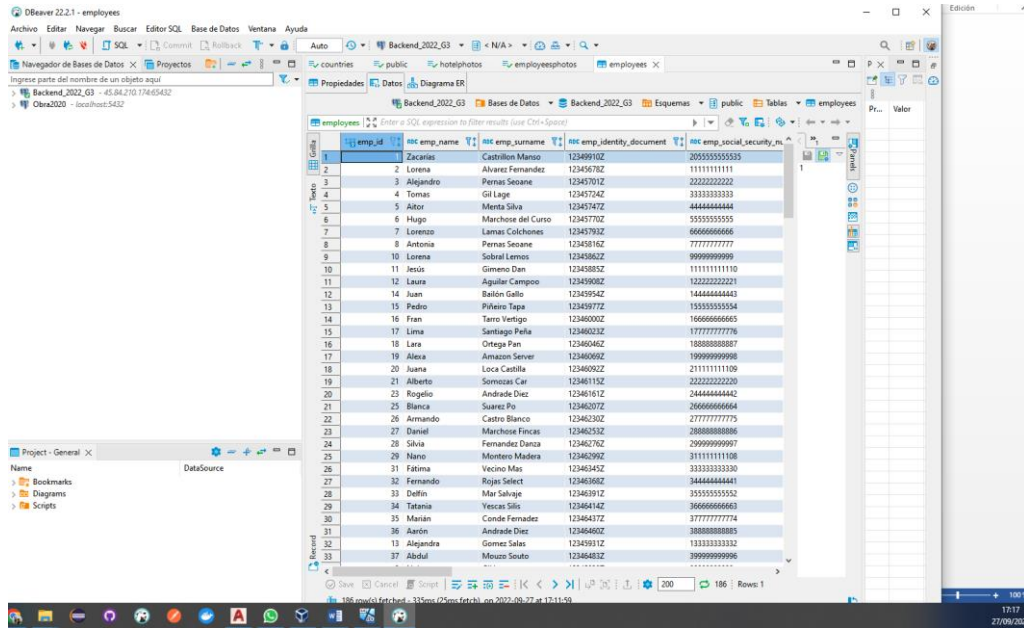


Ilustración 9 Dbeaver

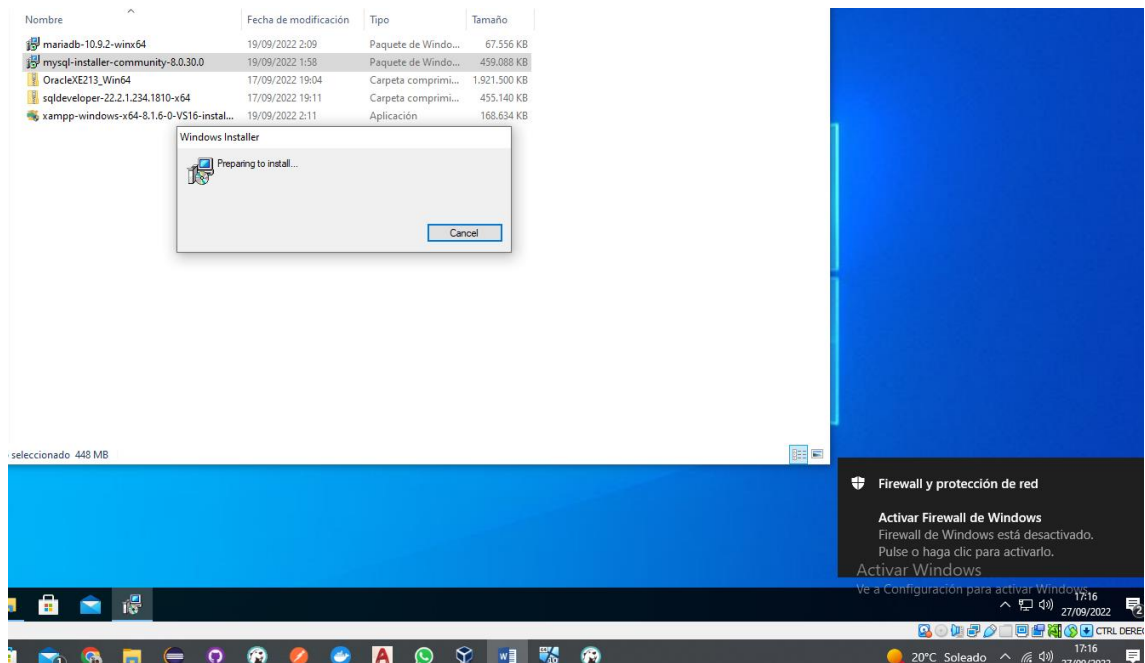
PRÁCTICA 1 ACCESO A DATOS– ALUMNO: CESAR BOUZAS SOTO



2.2.2 Mysql.

Usando la máquina virtual AD2023 ,usuario: admin contraseña: abcd123.

Como sólo necesitamos el servidor instalamos server only.



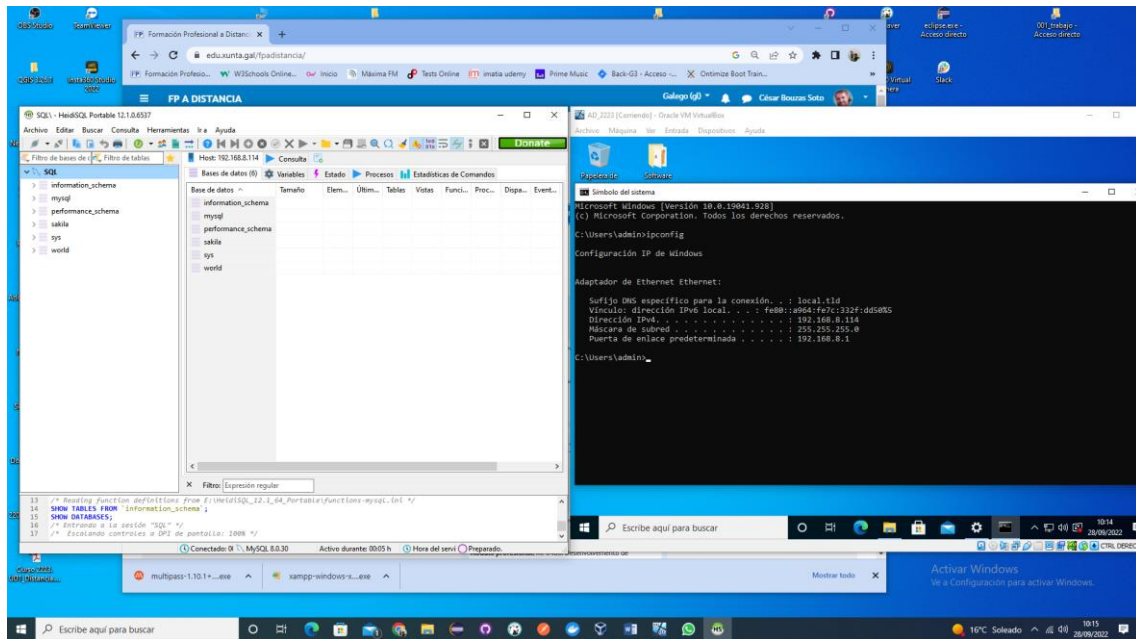
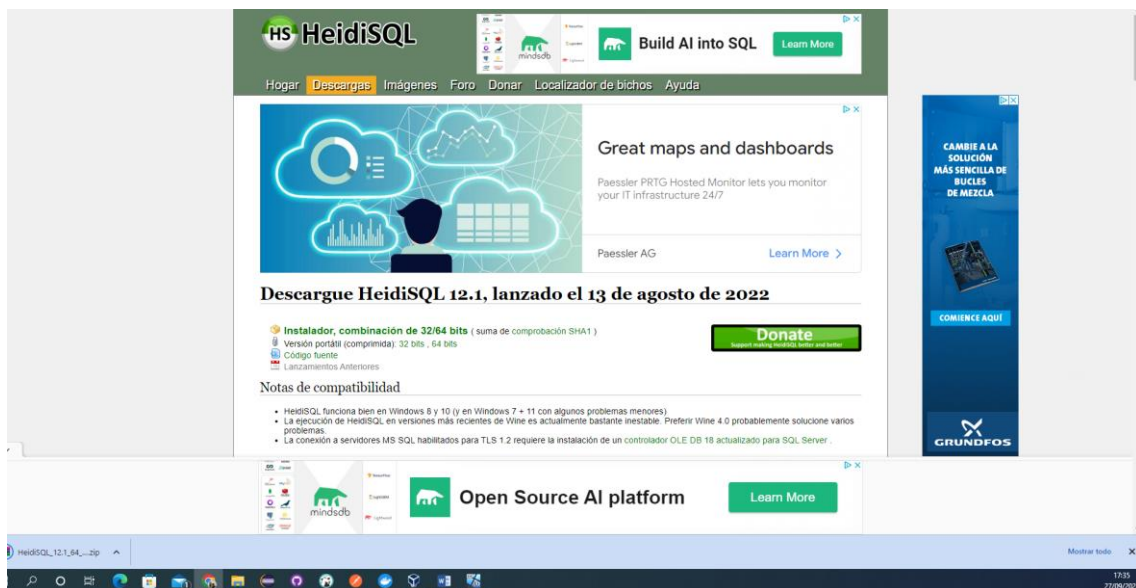


Ilustración 10 máquina virtual y heidi en local

2.2.3 Heidi

Descargamos la versión 64 bits portable en un usb ejecutando HeidiSQL.exe.



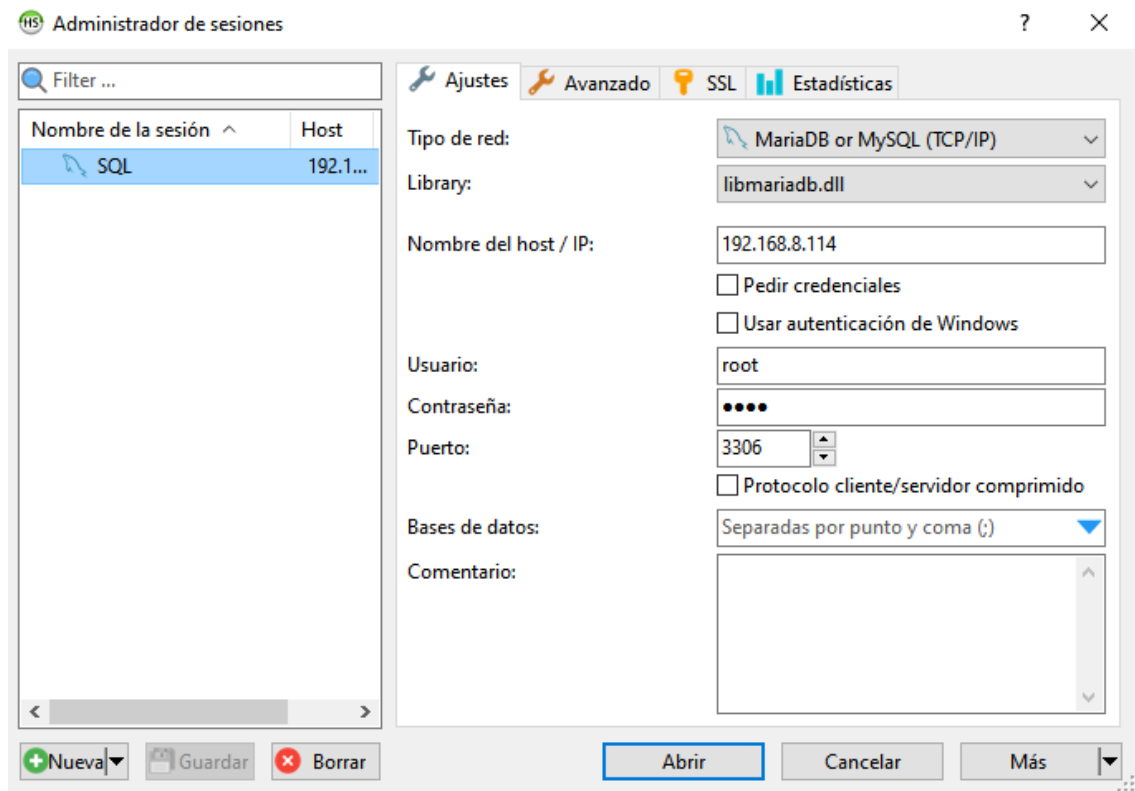
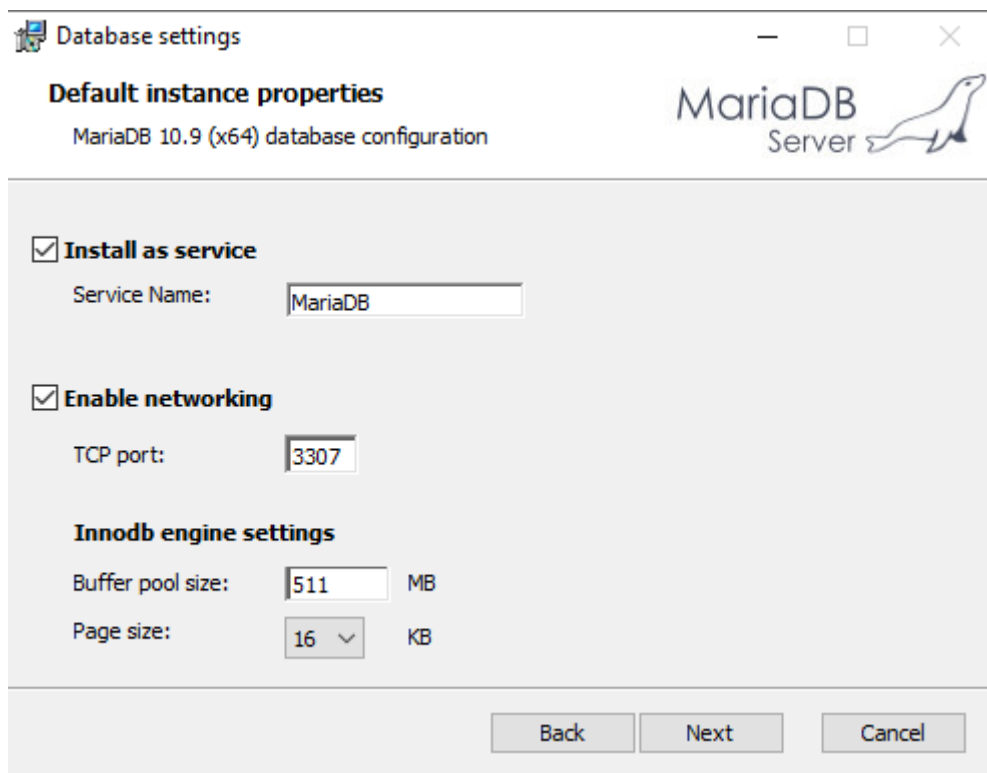
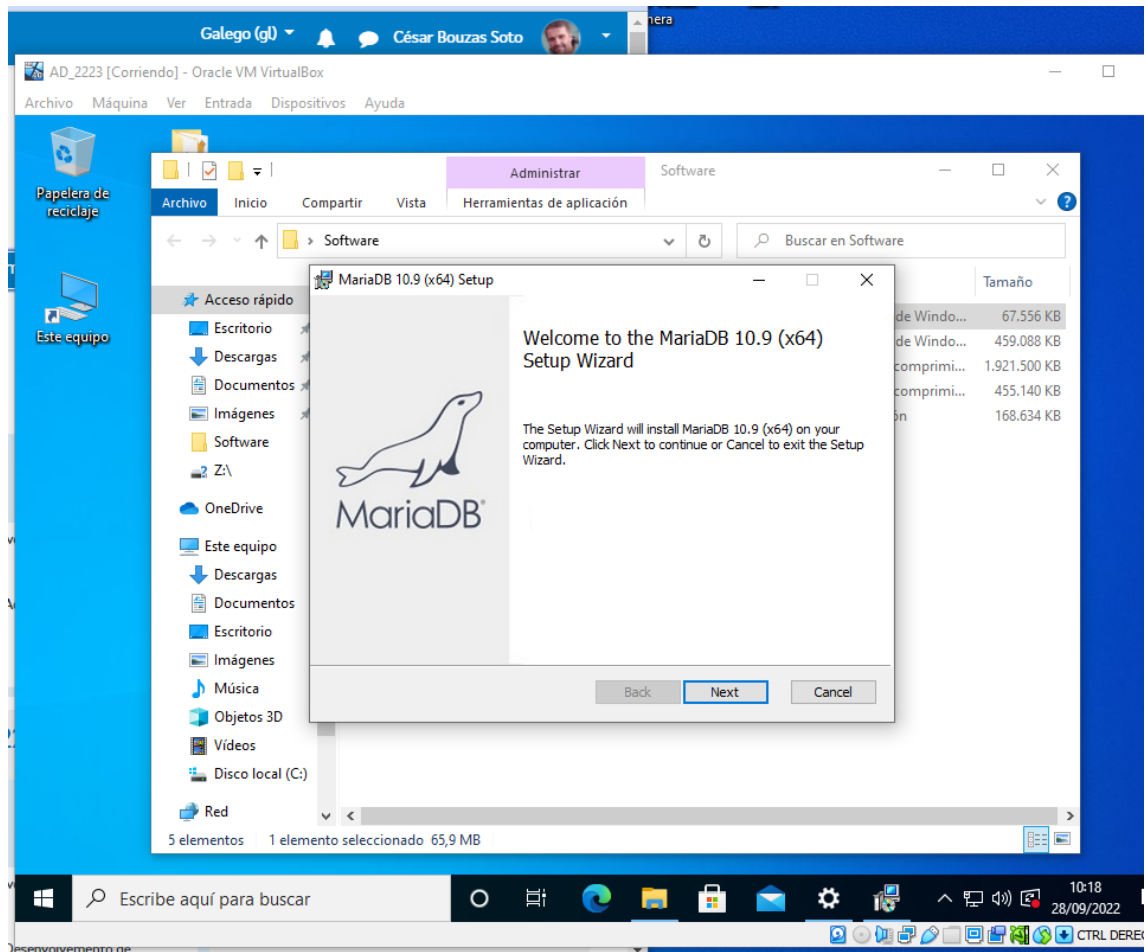


Ilustración 11 ventana de configuración conexión.

2.2.4 mariaDB.



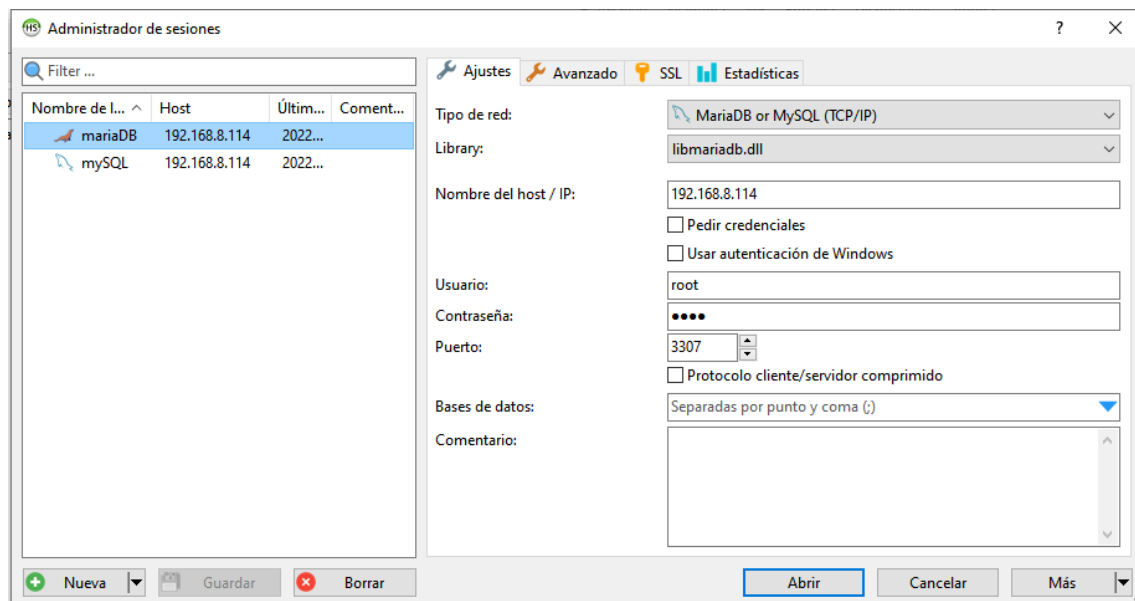
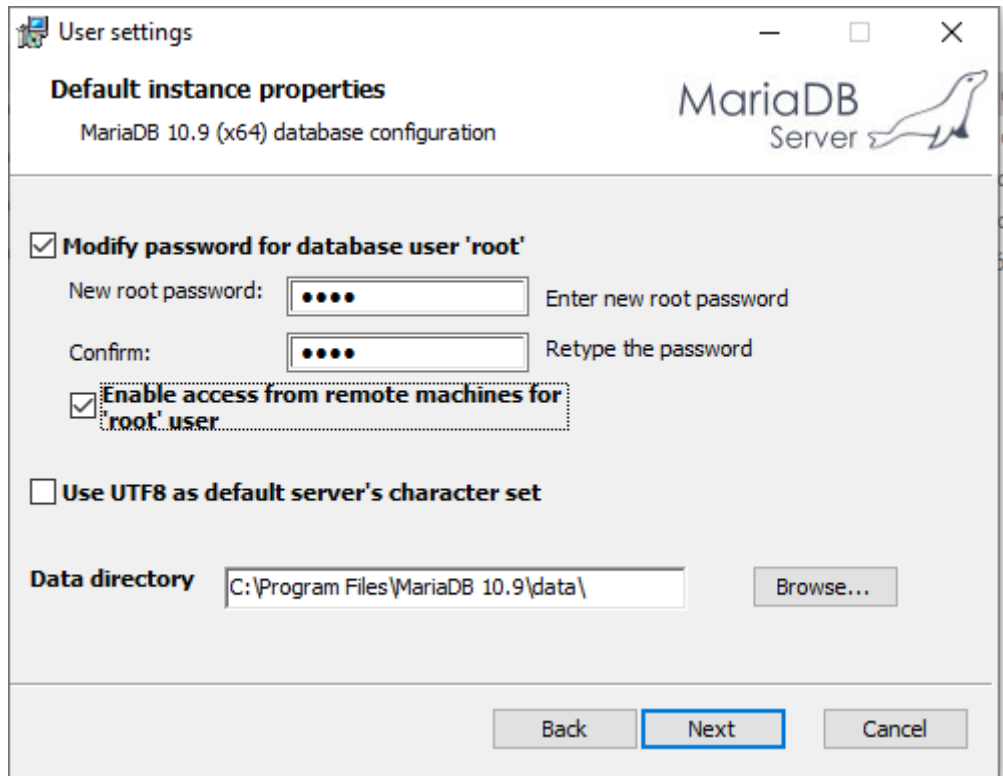


Ilustración 12 conexión mediante Heidi

Ilustración 1 Bd Jerárquica.....	2
Ilustración 2 Bd en Red.	3
Ilustración 3 Bd Relacional.....	3
Ilustración 4 Bd deductivas.....	4
Ilustración 5 Bd multidimensionales.....	4
Ilustración 6 Bd Orientada a Objetos.....	5
Ilustración 7 Bd orientada a Documentos	5
Ilustración 8 Netbeans 14 ejecutándose	8
Ilustración 9 Dbeaver.....	8
Ilustración 10 máquina virtual y heidi en local	10
Ilustración 11 ventana de configuracion conexión.	11
Ilustración 12 conexión mediante Heidi	13