



# Práctica evaluable 11

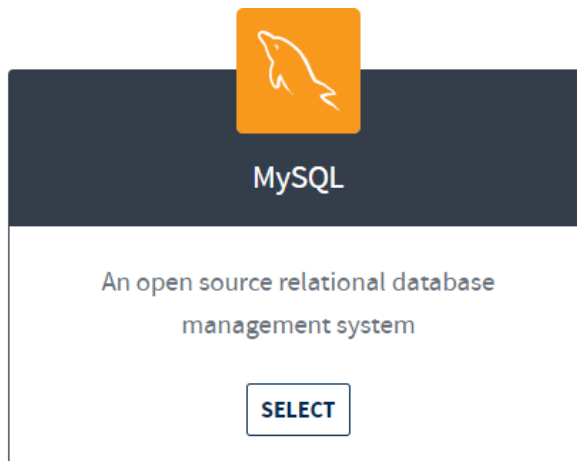
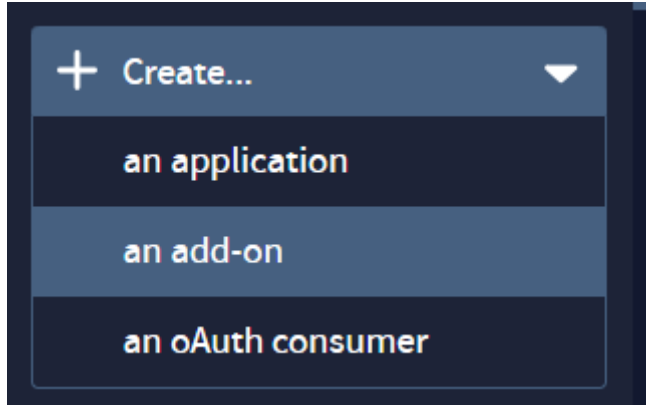
## Utilización de técnicas de acceso a datos

## ***Índice:***

1. Creación de la Base de Datos en Clever Cloud
  1. Registro y Creación de la Aplicación
  2. Configuración de la Base de Datos
  3. Despliegue de la Aplicación
2. Incorporación de Datos a la Base de Datos
  1. Creación de Tablas
  2. Inserción de Datos
3. Obtención de Datos mediante Consultas SQL
  1. Consulta de Datos
  2. Procesamiento de Datos
4. Representación Gráfica de Datos
  1. Selección de Biblioteca de Visualización
  2. Generación de Gráficos
  3. Integración en la Aplicación Web

## 1. Creación de la Base de Datos en Clever Cloud

1. **Registro y Creación de la Aplicación:** Se accede a la plataforma de Clever Cloud y se procede al registro en el servicio. Luego, se crea una nueva aplicación y se elige el plan de base de datos deseado.



2. **Configuración de la Base de Datos:** Se proporcionan los detalles de configuración necesarios, como el tipo de base de datos, la ubicación del servidor, las credenciales de acceso y cualquier otra configuración específica requerida.

<input checked="" type="radio"/>	DEV	Daily - 7 Retained	No	5
----------------------------------	-----	-----------------------	----	---

What is the name of your MySQL add-on? In which region should it be located?

NAME: \*

Practica

ZONE: \*

	<div><div> Paris France</div><div>infra:clever-cloud</div></div> <div> Montreal Canada</div>	<div></div> <div></div>
--	--	-------------------------

3. **Despliegue de la Aplicación:** Una vez configurada la base de datos, se despliega la aplicación en el entorno de Clever Cloud. Esto permite que la base de datos esté disponible y lista para su uso.

## Database Credentials

Get credentials for manual connections to this database.

Host

bcunajgas8gtokcubdxx-mysql.services.clever-cloud.com

Database Name

bcunajgas8gtokcubdxx

User

## 2. Incorporación de Datos a la Base de Datos

1. **Creación de Tablas:** Se diseñan las tablas de la base de datos de acuerdo con los requisitos del proyecto. Esto implica definir la estructura de las tablas, incluidos los tipos de datos y las relaciones entre ellas.

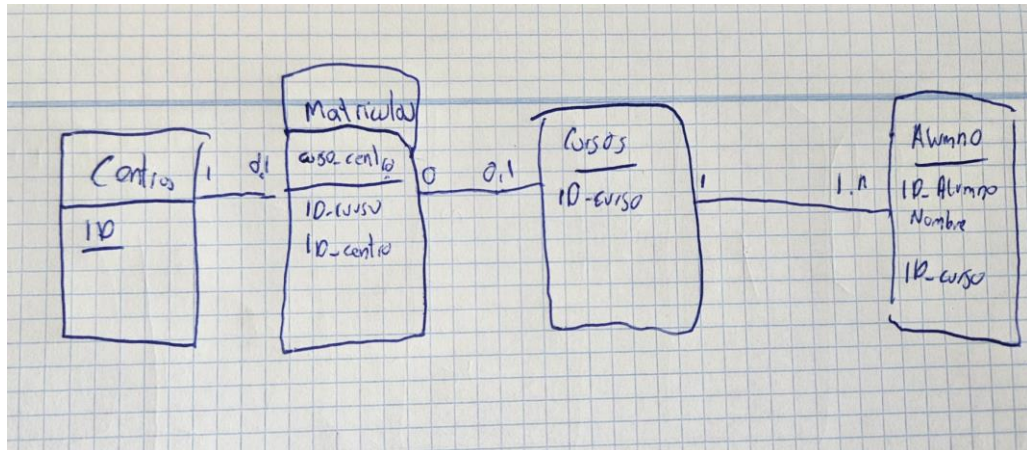
*Tabla de Cursos: id (clave primaria) fecha\_importacion nivel descripcion lugar nombre*

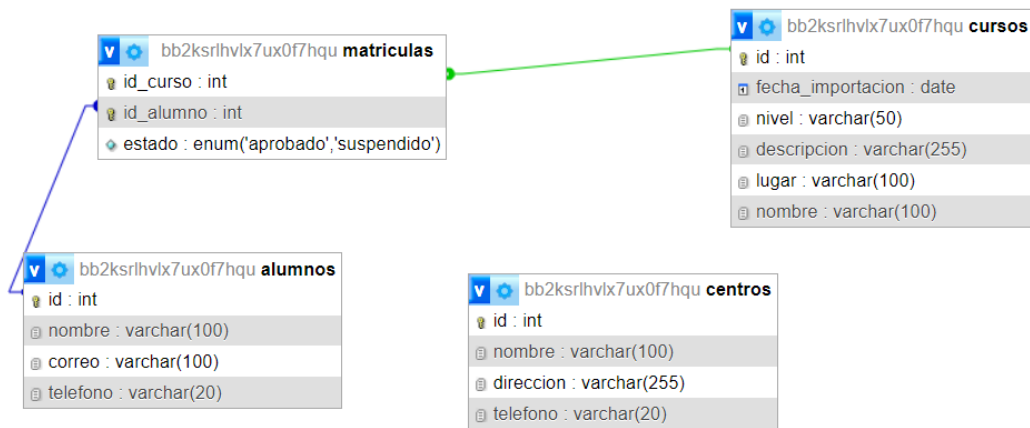
*Tabla de Centros: id (clave primaria) nombre direccion telefono*

*Tabla de Alumnos: id (clave primaria) nombre correo telefono*

*Tabla de Matrículas: id\_curso (clave foránea referenciando a la tabla de Cursos)*

*id\_alumno (clave foránea referenciando a la tabla de Alumnos)*





#### -- Tabla de Cursos

```
CREATE TABLE cursos (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    fecha_importacion DATE,  
    nivel VARCHAR(50),  
    descripcion VARCHAR(255),  
    lugar VARCHAR(100),  
    nombre VARCHAR(100)  
);
```

#### -- Tabla de Centros

```
CREATE TABLE centros (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nombre VARCHAR(100),  
    direccion VARCHAR(255),  
    telefono VARCHAR(20)  
);
```

#### -- Tabla de Alumnos

```
CREATE TABLE alumnos (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nombre VARCHAR(100),
```

Rodrigo Lopez Alfaro

```
correo VARCHAR(100),  
telefono VARCHAR(20)  
);
```

#### -- **Tabla de Matrículas**

```
CREATE TABLE matriculas (  
    id_curso INT,  
    id_alumno INT,  
    estado ENUM('aprobado', 'suspendido'),  
    FOREIGN KEY (id_curso) REFERENCES cursos(id),  
    FOREIGN KEY (id_alumno) REFERENCES alumnos(id),  
    PRIMARY KEY (id_curso, id_alumno)  
);
```

2. **Inserción de Datos:** Yo he incorporado los datos en las tablas recién creadas utilizando consultas SQL de inserción. Estos datos pueden ser generados manualmente o mediante la importación desde archivos CSV u otras fuentes de datos.

#### -- **Datos aleatorios en la tabla de Cursos**

```
INSERT INTO cursos (fecha_importacion, nivel, descripcion, lugar, nombre)  
VALUES  
    ('2023-01-15', 'Nivel Avanzado', 'Curso de Programación Avanzada', 'Aula 101', 'Programación Avanzada'),  
    ('2023-02-20', 'Nivel Intermedio', 'Curso de Inglés Intermedio', 'Aula de Idiomas', 'Inglés Intermedio'),  
    ('2023-03-25', 'Nivel Básico', 'Curso de Fotografía Básica', 'Estudio de Fotografía', 'Fotografía Básica'),  
    ('2023-04-10', 'Nivel Avanzado', 'Curso de Diseño Gráfico Avanzado', 'Taller de Diseño', 'Diseño Gráfico Avanzado'),  
    ('2023-05-05', 'Nivel Intermedio', 'Curso de Marketing Digital Intermedio', 'Sala de Conferencias', 'Marketing Digital Intermedio'),  
    ('2023-06-12', 'Nivel Avanzado', 'Curso de Desarrollo Web Avanzado', 'Laboratorio de Computación', 'Desarrollo Web Avanzado'),  
    ('2023-07-18', 'Nivel Básico', 'Curso de Cocina Básica', 'Cocina Gourmet', 'Cocina Básica'),  
    ('2023-08-22', 'Nivel Intermedio', 'Curso de Francés Intermedio', 'Aula de Idiomas', 'Francés Intermedio'),
```

('2023-09-30', 'Nivel Avanzado', 'Curso de Inteligencia Artificial Avanzada', 'Sala de Seminarios', 'Inteligencia Artificial Avanzada'),

('2023-10-10', 'Nivel Básico', 'Curso de Yoga Básico', 'Salón de Yoga', 'Yoga Básico');

**-- Datos aleatorios en la tabla de Centros**

INSERT INTO centros (nombre, direccion, telefono)

VALUES

('Academia XYZ', 'Calle Principal 123', '123-456-7890'),

('Instituto ABC', 'Avenida Central 456', '098-765-4321'),

('Escuela DEF', 'Plaza Mayor 789', '456-789-0123'),

('Centro de Formación GHI', 'Avenida Secundaria 234', '789-012-3456'),

('Academia JKL', 'Calle Secundaria 567', '012-345-6789');

**-- Datos aleatorios en la tabla de Alumnos**

INSERT INTO alumnos (nombre, correo, telefono)

VALUES

('Juan Pérez', 'juan@example.com', '555-123-4567'),

('María González', 'maria@example.com', '555-234-5678'),

('Carlos Rodríguez', 'carlos@example.com', '555-345-6789'),

('Ana Martínez', 'ana@example.com', '555-456-7890'),

('Luisa Sánchez', 'luisa@example.com', '555-567-8901');

**-- Datos aleatorios en la tabla de Matrículas**

INSERT INTO matriculas (id\_curso, id\_alumno, estado)

SELECT

(SELECT id FROM cursos ORDER BY RAND() LIMIT 1),

(SELECT id FROM alumnos ORDER BY RAND() LIMIT 1),

IF(RAND() < 0.8, 'aprobado', 'suspendido')

FROM

information\_schema.tables

LIMIT

10; -- Limite de cantidad

### 3. Obtención de Datos mediante Consultas SQL

1. **Consulta de Datos:** Se utilizan consultas SQL para recuperar los datos necesarios de la base de datos. Esto puede incluir consultas simples como SELECT para obtener todos los registros de una tabla o consultas más complejas que involucren JOINS y condiciones.

```
// Consulta SQL para obtener los alumnos matriculados en el curso
const sql = `SELECT alumnos.nombre, matriculas.estado
              FROM matriculas
              INNER JOIN alumnos ON matriculas.id_alumno = alumnos.id
              WHERE matriculas.id_curso = ?`;
```

```
get('/centros', (req, res) => {
  pool.query('SELECT * FROM centros', (error, results) => {
    if (error) {
```

2. **Procesamiento de Datos:** Una vez que se obtienen los datos, se pueden procesar según sea necesario para su análisis o visualización. Esto puede implicar la agregación de datos, el cálculo de métricas específicas o la transformación de los datos en un formato adecuado para la representación gráfica.

### 4. Representación Gráfica de Datos

1. **Selección de Biblioteca de Visualización:** Se elige una biblioteca de visualización adecuada para representar los datos de manera gráfica. En este caso, se optó por utilizar Chart.js debido a su facilidad de uso y flexibilidad.

```
C:\Users\Personal\Documents\DAW\2DAW\Desarrollo_Servidor\2trimestre\nodeSQL>npm install chart.js
```

2. **Generación de Gráficos:** Se utilizan los datos obtenidos de la base de datos para generar gráficos utilizando la biblioteca de visualización seleccionada. Esto implica configurar el tipo de gráfico, los datos a representar y cualquier otra configuración necesaria.

```
app.get('/graficas', (req, res) => {
  // Consulta SQL para obtener el ratio de alumnos aprobados por el total de matriculados para cada curso
  const sql = `
    SELECT
      cursos.nombre AS nombre_curso,
      COUNT(CASE WHEN matriculas.estado = 'aprobado' THEN 1 ELSE NULL END) AS aprobados,
      COUNT(*) AS total_matriculados
    FROM
      cursos
    LEFT JOIN
      matriculas ON cursos.id = matriculas.id_curso
    GROUP BY
      cursos.nombre;
  `;
```



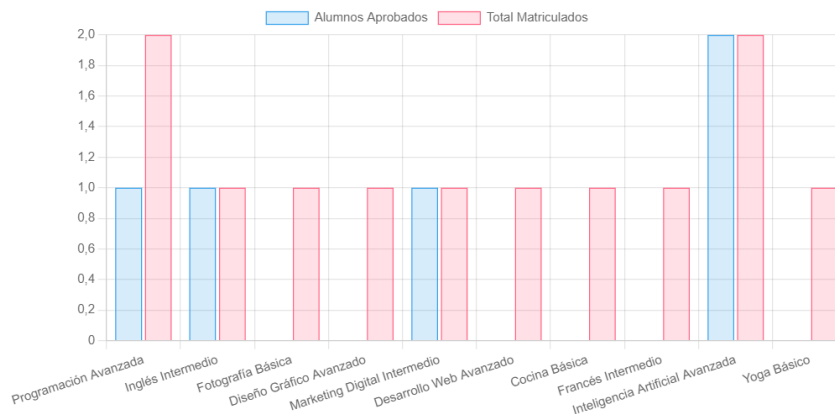
```
// Ejecutar la consulta en la base de datos
pool.query(sql, (error, results) => {
  if (error) {
    // Si hay un error, redirigir a una página de error o mostrar un mens
    res.status(500).send('Error al obtener datos para las gráficas.');
```

```
    return;
  }

  // Extraer Los datos del resultado de la consulta
  const datosGrafica = results.map(row => ({
    nombreCurso: row.nombre_curso,
    alumnosAprobados: row.aprobados,
    totalMatriculados: row.total_matriculados
  }));

  // Enviar los datos al cliente en formato JSON
  res.json(datosGrafica);
});
```

3. **Integración en la Aplicación Web:** Los gráficos generados se integran en la aplicación web para que los usuarios puedan acceder a ellos y visualizar los datos de manera interactiva.



## Conclusiones

El proceso de creación de una base de datos en Clever Cloud, la incorporación de datos, la obtención de datos mediante consultas SQL y la representación gráfica de los mismos es fundamental para el desarrollo de aplicaciones web robustas y con capacidades de análisis de datos avanzadas. Clever Cloud proporciona una plataforma confiable y escalable para gestionar bases de datos en la nube, mientras que las bibliotecas de visualización como Chart.js permiten crear visualizaciones atractivas y significativas de los datos. Este proceso es importante para tomar decisiones informadas basadas en datos y proporcionar a los usuarios una experiencia enriquecida.