

Ejemplo Flask y BD

1.No te olvides de **instalar flask**:

```
pip install flask
```

2.Creamos la base de datos:

```
CREATE DATABASE escuela;
```

3.Creamos nuestra tabla:

```
USE escuela;
```

```
CREATE TABLE alumno (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nombre VARCHAR(100) NOT NULL,  
    edad INT,  
    curso VARCHAR(50)  
);
```

4.Desde phpMyAdmin insertamos algunos datos en la tabla.

5.Instalar el conector de MySQL en Python desde la terminal.

```
pip install mysqlclient
```



Cofinanciado por
la Unión Europea



MINISTERIO
DE TRABAJO
Y ECONOMÍA SOCIAL



Fondos Europeos



Ayuda
en Acción
Impulsa Empleo Joven

6. código de app.py

```
# Importamos la clase Flask y la función render_template de Flask.
# render_template se usa para devolver el HTML renderizado desde el
# archivo de plantilla.
from flask import Flask, render_template
# Importamos pymysql para poder trabajar con bases de datos MySQL.
import pymysql
# Creamos una instancia de la clase Flask. Esto es necesario para
# configurar nuestra aplicación web.
app = Flask(__name__)
# Intentamos conectar con la base de datos usando un bloque try-except
# para manejar errores de forma segura.
try:
    print("Intentando conectar a la base de datos...")
# Establecemos la conexión a la base de datos MySQL utilizando PyMySQL.
# Debemos indicar el host, usuario, contraseña y la base de datos a
# la que nos queremos conectar.
    db = pymysql.connect(
        host="localhost",
        user="root",
        password="", # Deja vacío
        database="escuela"
    )
    print("Conexión exitosa con la base de datos.")
# Si ocurre un error al intentar conectar, el bloque except lo captura y
# muestra el error.
except Exception as e:
    print(f"Error al conectar con la base de datos: {e}")

# Definimos la ruta principal ("/") que es la que se abrirá cuando el
# usuario acceda a la aplicación web.
@app.route('/')
def mostrar_alumnos():
# Creamos un cursor para ejecutar las consultas en la base de datos.
    cursor = db.cursor()
    # Ejecutamos una consulta SQL para obtener todos los nombres de los
    # alumnos de la tabla "alumno".
    cursor.execute("SELECT nombre FROM alumno")
    # Obtenemos todos los resultados de la consulta. El resultado es una
    # lista de tuplas.
    alumnos = cursor.fetchall() # Lista de tuplas [(nombre1,),
    (nombre2,)]
    cursor.close()
    return render_template('index.html', alumnos=alumnos)
```



REGISTRO NACIONAL DE ASOCIACIONES N°611922
DECLARADA ENTIDAD DE UTILIDAD PÚBLICA ESTATAL
AGENCIA DE COLOCACIÓN: ID 0100000017

CENTRO EN MÁLAGA
C/DOS ACERAS 23, 29012
MÁLAGA | (+34) 952 300 500
ARRABAL@ARRABALEMPLEO.ORG

CENTRO EN CÁDIZ
TR.º ALAMEDA DE SOLANO, 32, 11130
CHICLANA DE LA FRONTERA | (+34) 956 900 312
CHICLANA@ARRABALEMPLEO.ORG



Cofinanciado por
la Unión Europea



MINISTERIO
DE TRABAJO
Y ECONOMÍA SOCIAL



Fondos Europeos



Junta de Andalucía
Fondo de Empleo Juvenil
Ayuda al Empleo Juvenil
REINTEGRO AL EMPLEO



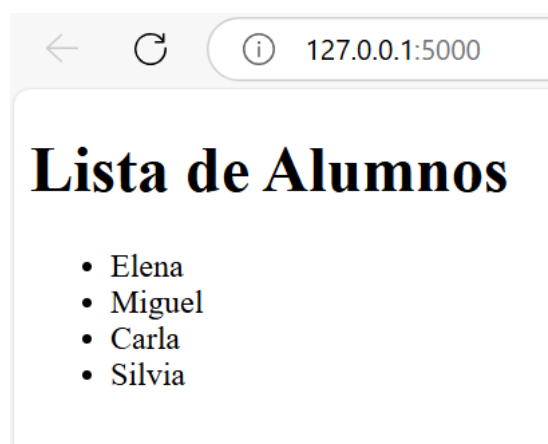
Ayuda
en Acción
Impulsa Empleo Joven

```
if __name__ == '__main__':  
    app.run(debug=True)
```

7. Html (no olvides que debe estar en la carpeta templates para que flask lo encuentre)

```
<!DOCTYPE html>  
<html lang="es">  
<head>  
    <meta charset="UTF-8">  
    <title>Alumnos</title>  
</head>  
<body>  
    <h1>Lista de Alumnos</h1>  
    <ul>  
        {% for alumno in alumnos %}  
            <li>{{ alumno[0] }}</li> <!-- Mostramos el nombre del alumno -->  
        {% endfor %}  
    </ul>  
</body>  
</html>
```

Y ahora cuando ejecutemos Python app.py tenemos el siguiente resultado:



REGISTRO NACIONAL DE ASOCIACIONES N°611922
DECLARADA ENTIDAD DE UTILIDAD PÚBLICA ESTATAL
AGENCIA DE COLOCACIÓN: ID 0100000017

CENTRO EN MÁLAGA
C/DOS ACERAS 23, 29012
MÁLAGA | (+34) 952 300 500
ARRABAL@ARRABALEMPLEO.ORG

CENTRO EN CÁDIZ
TR.ª ALAMEDA DE SOLANO, 32, 11130
CHICLANA DE LA FRONTERA | (+34) 956 900 312
CHICLANA@ARRABALEMPLEO.ORG