***PHP***

# -Basics

* Las variables se crearán usando $ y utilizando camelCase. Ej: $miVarible.
* Las constantes se definen usando define(‘NOMBRE’, valores) y usando SNAKE\_CASE.
* Para unir se usará un punto (.) en vez del más (+).
* Para ejecutar php dentro del HTML usaremos <?php ?> o <?=?>, en el caso del segundo equivale a hacer <?php echo ?>.
* Para crear una función usaremos function nombreFunct(tipo $valor):tipoDevuelto{}.
* El foreach tendrá la siguiente estructura ($variable as $key => $value), donde el =>$value se podrá omitir.

# -Instalación SQLite

Descargamos la última versión de SQLite (binaries y tools) y lo añadimos los archivos del .zip en C:\Windows\System32 para poder usarlo desde la terminal.

Creamos el archivo en caso de que no exista siguiendo los siguientes pasos:

* En la terminal escribimos sqlite3 para iniciarlo.
* Escribimos el nombre de nuestra base de datos, con “.open NOMBRE\_BD.db”, en caso de que no exista SQLite creará el archivo.

# -Instalación MariaDB (Windows)

Instalamos la última versión de MariaDB, asignamos contraseña a root y añadimos el directorio que contenga C:\Program Files\MariaDB 11.2\bin en el PATH.

Para crear una copia de la base de datos deberemos hacer lo siguiente:

*mysqldump -u nombre\_usuario -p nombre\_base\_de\_datos > backup.sql*

Para cargar la copia usando PowerShell usaremos:

*Get-Content backup.sql | mysql -u nombre\_usuario -p ejercicios*

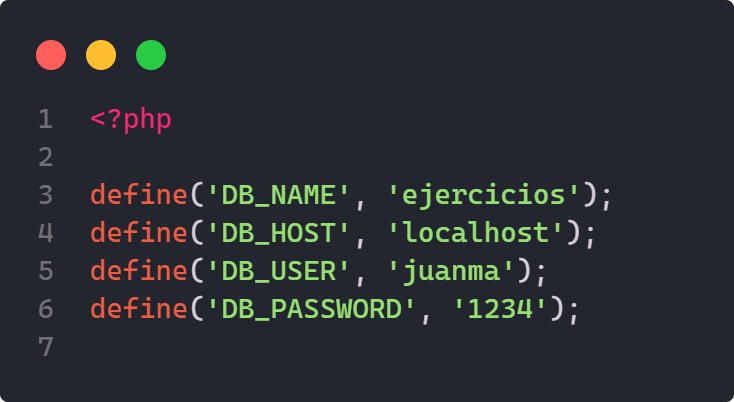
Para usar la base de datos que queramos, usaremos:

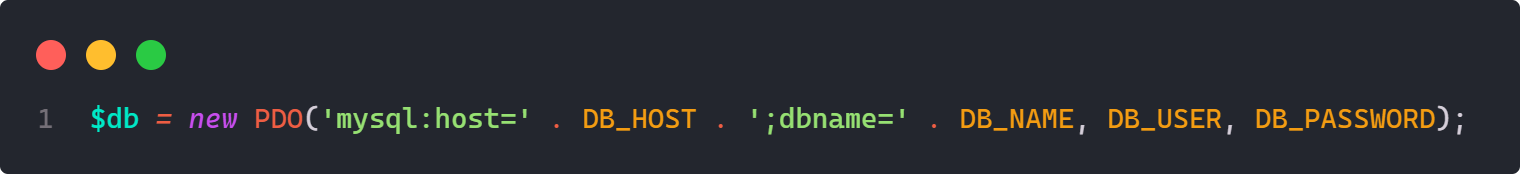
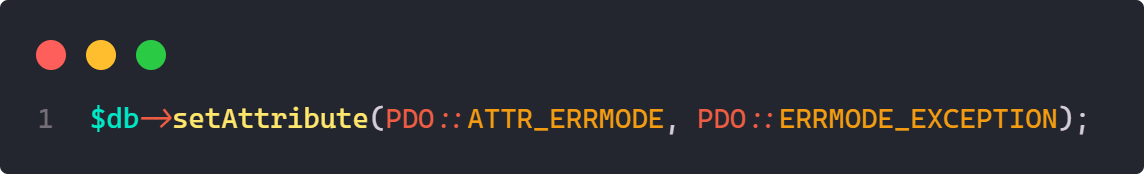
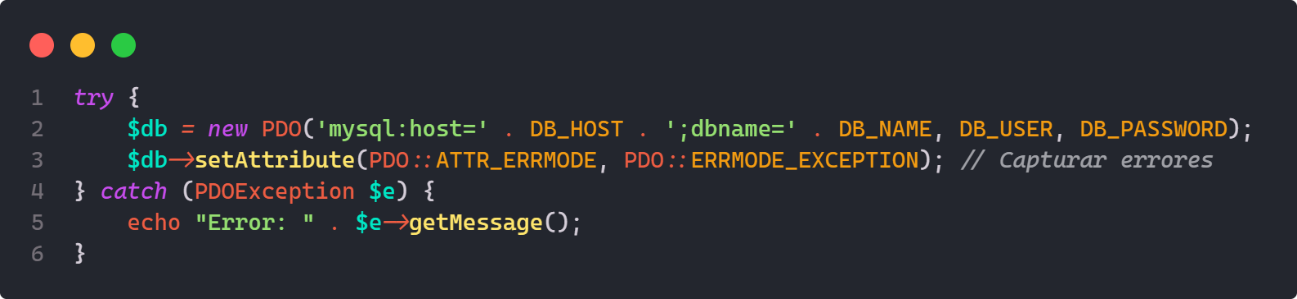
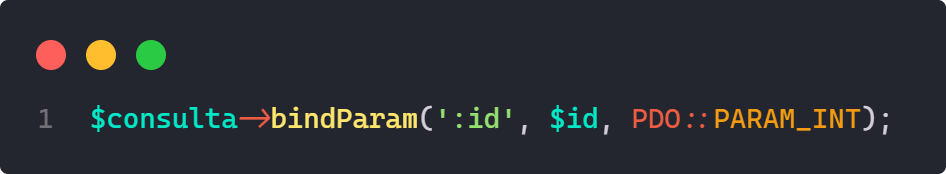
*USE NOMBRE\_DATABASE;*

# -Conexión con PHP a MariaDB

Para conectarnos a la base de datos deberemos seguir los siguientes pasos:

1. Tendremos un archivo data.php con los datos de la conexión.



1. Para conectarnos deberemos incluir el archivo anterior con *require\_once(‘data.php’)* o include(‘data.php’) usando el autoload.
2. Lo siguiente será crear un try-catch para manejar los errores que puedan llegar a surgir.
3. Dentro del try usaremos ***PDO*** para crear la conexión:
4. Podemos activar el manejo de errores mediante:
5. Ejemplo de como quedaría la creación de la conexión:
6. Para cerrar la conexión usaremos el nombre de la variable que contiene la conexión y la igualaremos a *null*.
7. Antes de ejecutar la consulta deberemos asignar a los parámetros (‘:id’) su valor para realizar la consulta mediante bindParam(‘:param’, $valor, tipo)
8. Para crear consultas debemos usar el método *prepare()* y guardaremos la consulta en una variable para más tarde poder ejecutarla con *execute()*.