

Conectarse a una base de datos a través de PHP

Una de las maneras de gestionar una conexión a una base de datos en PHP es a través de la clase **PDO**. En su constructor, puede recibir 4 parámetros.

- DSN. Obligatorio. Es un string que indica datos de la conexión.
- Username. Optativo. Un string que indica nombre de usuario con el que tratará de conectarse.
- Password. Optativo. Un string que indica la contraseña con la que tratará de conectarse.
- Options. Optativo. Es un array que puede recibir diversas de la conexión en un array asociativo.

Dado esto, una conexión tendría la siguiente sintaxis:

```
$db = new PDO($dsn, $username, $password, $options);
```

El DSN es un string que podría tener por ejemplo la siguiente información

```
'mysql:host=localhost;dbname=testdb;charset=utf8mb4;port:3307'
```

1. Generar una conexión a la última base de datos utilizada durante las clases de MySQL aclarando en el DSN únicamente **host** y **dbname**.
2. ¿Qué sucede si utilizamos un **host** o un **dbname** erróneo? ¿Qué sucede si utilizamos credenciales erróneas?
3. Agregar dentro del DSN el puerto deseado.
4. Agregar dentro del DSN el charset deseado.
5. Modificar el código con una estructura **try/catch** para que presente errores en la conexión de forma más amigable.
6. Teniendo esta nueva estructura, ¿que podemos visualizar ahora dentro del **catch** si hay un error?
7. (optativo) Pedirle a un docente que apague el servidor de MySQL. ¿Qué sucede en tal caso?

Prácticas con queries.

Utilizando el método **query** de la clase **PDO**:

1. Armar un archivo `generos.php` que arme un `` con todos los géneros de nuestra base de datos.
2. Armar un archivo `peliculas.php` que arme un `` con todas las películas de nuestra base de datos.

Statements

1. Realizar un archivo serie.php que muestre los datos de **una** serie. Para esto, tendremos una variable \$idSerie que será modificada directamente en el código (ejemplo: \$idSerie = 5;). La consulta debe hacerse a través de un statement.
2. Partiendo de agregarPelicula.php modificarlo para que al enviarse el formulario inserte una nueva película en la base de datos utilizando statements.
3. Sumarle a agregarPelicula.php un <select> con todos los géneros para poder elegir el mismo.

Ejercicios integradores

1. Armar un archivo actores.php que arme un con todos los actores de nuestra base de datos.
2. Modificar generos.php para que por cada uno de los géneros, además, arme un sector en donde indique todas las películas de dicho género.
3. Modificar generos.php para que el título de cada película sea además un link a pelicula.php enviando por GET el id de dicha película. pelicula.php debe mostrar la información de dicha película.
4. Modificar generos.php para que los títulos de películas de los géneros solamente aparezcan luego de clickear en el nombre del género. Al clickear de nuevo, estos deberían ocultarse.
5. Realizar un sitio que se llame favoritas.php que liste los actores que tienen película preferidas e indique cuáles son.
6. Modificar favoritas.php para que muestre todos los actores y si no tiene película preferida diga "No tiene".
7. Modificar agregarPelicula.php para que valide que el título junto con la fecha de estreno de la película no se encuentre repetido.
8. Agregarle a agregarPelicula.php validación y persistencia de los datos (en caso de error)

