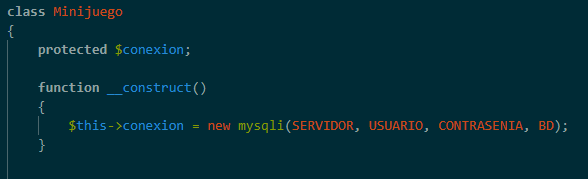
**Consultas preparadas**

Las consultas preparadas se utilizan para ejecutar una sentencia de forma repetida y con eficacia, haciendo exactamente lo que pretendemos y evitando que pueda inyectar código indeseado.

**Inserciones**

**Cómo hacer una consulta preparada para inserción**

1º Crear la conexión con la BD.



2.0º Crear consulta almacenada en variable (aconsejable hacerlo así).



2º Almacenar en una variable el resultado o statement del método prepare sobre la conexión. El método toma como argumento la consulta.



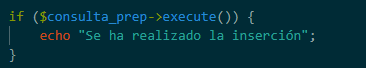
También se crean las variables que que se usarán en el siguiente paso.



3º Asociar a la variable anterior el tipo de datos y las varaibles definidas mediante bind\_param().



4º Ejecutar la consulta.



5º Cerrar el prepare().



**Inserción masiva preparada**

Funciona igual que el apartado anterior hasta el punto 2.

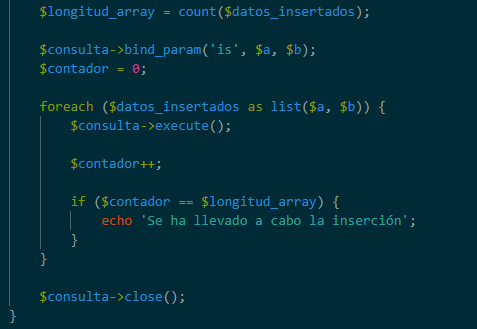
3º Se crea una variable que almacene varios conjuntos de datos por ejemplo; en un array.



4º Se crea un bucle para recorrer el array. En este caso, se usa un foreach con el método list(), que recoge cada variable de cada array en una variable.

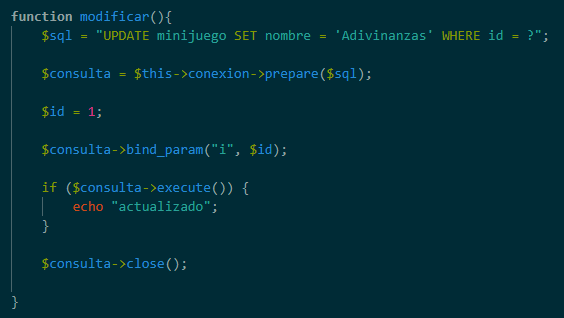
$a guarda el primer ítem de los arrays contenidos en $datos\_insertados ($id)

$b guarda el segundo ($nombre)



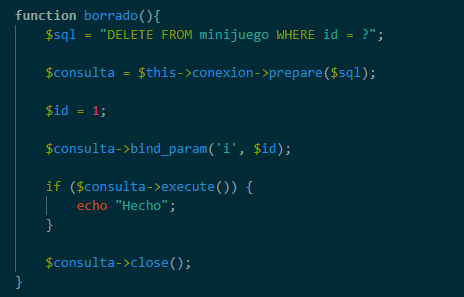
**Actualizar**

Lo único que cambia con respecto a la inserción es la consulta sql, y las variables de bind\_param() si es necesario.

****

**Borrar**

Lo mismo en este caso que para actualizar.

****

**Select**

Esta parte es algo más difícil.

1º Se crea la consulta



2º Se prepara la consulta

$consulta = $this->conexion->prepare($sql);

3º Se asocian parámetros

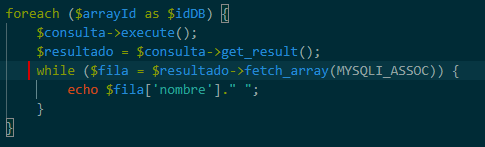


Hasta aquí todo igual.

4º Se va a crear una variable para este ejemplo que tenga los valores del array.



5º Para recorrer dicho array se usará un foreach, aunque podría usarse un for u otro tipo de bucle.



El foreach ejecuta la consulta con cada valor del array, es decir; hace 4 consultas y almacena con get\_result() el resultado de las consultas en la variable resultado. Por último se hace el fetch\_array() como de costumbre y se hace lo que se desee con el resultado.