

Acceso a bases de datos desde lenguajes de «script» de servidor (Implantación de aplicaciones web)

Integración de los lenguajes de «script» de servidor con los sistemas gestores de base de datos

Mediante PHP se puede acceder y actualizar información almacenada en una base de datos de MySQL.

Esta sección proporciona una introducción a las opciones disponibles a la hora de desarrollar una aplicación PHP que necesite interactuar con una base de datos MySQL.

Conexión a bases de datos

La forma de realizar las conexiones a bases de datos se realiza mediante un API, una Interfaz de Programación de Aplicaciones, o API, declara las clases, métodos, funciones y variables a las que necesitará llamar una aplicación para llevar a cabo una tarea determinada. En el caso de las aplicaciones PHP que necesiten comunicarse con un servidor de bases de datos, las APIs que se necesitarán se ofrecen generalmente en forma de extensiones de PHP.

La conexión se lleva a cabo mediante conectores, en la documentación de MySQL, el término conector hace referencia al software que permite a una aplicación conectarse a un servidor de bases de datos MySQL. MySQL proporciona conectores para ciertos lenguajes, entre ellos PHP.

Para conectar de forma concreta con un servidor de base de datos se utiliza un driver.

Las conexiones se establecen creando instancias de la clase base PDO. No importa que driver quieras utilizar; usa siempre el nombre de la clase PDO. El constructor acepta parámetros para especificar la fuente de datos (conocido como DSN) y opcionalmente el nombre de usuario y el password (si procede).

Creación de bases de datos y tablas

La creación de bases de datos y tablas se realiza con comandos de SQL.

Recuperación de la información de la base de datos desde una página Web

La recuperación de datos se realiza mediante consultas de SQL.

Modificación de la información almacenada: inserciones, actualizaciones y borrados

La modificación de información se realiza mediante consultas en SQL.

Verificación de la información

Todas las operaciones realizadas sobre las bases de datos tienen que ser verificadas correctamente.

Gestión de errores

El PDO te ofrece 3 diferentes estrategias de manejar errores, para adaptarse al estilo de tu desarrollo de la aplicación.

- `PDO::ERRMODE_SILENT`
- `PDO::ERRMODE_WARNING`
- `PDO::ERRMODE_EXCEPTION`

Mecanismos de seguridad y control de accesos

Es fundamental llevar un control de los usuarios que acceden al sistema gestor de base de dato, los usuarios tienen distintos permisos para realizar operaciones en el gestor.

Para reforzar esta medida también resulta conveniente emplear las funciones que aportan seguridad a los datos insertados, como por ejemplo `mysqli_real_escape_string()` o `crypt()` que se usa para encriptar las contraseñas, y seguir las recomendaciones que se hacen desde el sitio oficial de PHP con el fin de dificultar el uso de técnicas como SQL injection (SQLi).

Verificación del funcionamiento y pruebas de rendimiento

Antes de poner una aplicación en funcionamiento hay que realizar pruebas de todo tipo.