

CRUD

La base de la gestión de los
datos

Que es CRUD:

- El concepto CRUD está estrechamente vinculado a la gestión de datos digitales. CRUD hace referencia a **un acrónimo** en el que se reúnen las primeras letras de las cuatro operaciones fundamentales de aplicaciones persistentes en sistemas de bases de datos:
 - **Create** (Crear registros)
 - **Read** bzw. **Retrieve** (Leer registros)
 - **Update** (Actualizar registros)
 - **Delete** bzw. **Destroy** (Borrar registros)

Que es CRUD:

- En pocas palabras, CRUD resume las funciones requeridas por un usuario para crear y gestionar datos. Varios procesos de gestión de datos están basados en CRUD, en los que **dichas operaciones están específicamente adaptadas a los requisitos del sistema y de usuario**, ya sea para la gestión de bases de datos o para el uso de aplicaciones.

Que es CRUD:

- Para los expertos, las operaciones son las herramientas de acceso típicas e indispensables para comprobar, por ejemplo, los problemas de la base de datos, mientras que para los usuarios, CRUD significa crear una cuenta (créate) y utilizarla (read), actualizarla (update) o borrarla (delete) en cualquier momento. Dependiendo de la configuración regional, las operaciones CRUD pueden implementarse de diferentes maneras, como lo muestra la siguiente tabla:

Que es CRUD:

CRUD-Operation	SQL	RESTful HTTP	XQuery
Create	INSERT	POST, PUT	insert
Read	SELECT	GET, HEAD	copy/modify/return
Update	UPDATE	PUT, PATCH	replace, rename
Delete	DELETE	DELETE	delete

Frameworks CRUD: capa de acceso a las bases de datos

- Si los objetos individuales son **visualizados por medio de una interfaz gráfica** y modificados con las llamadas operaciones CRUD, entonces se habla de un framework CRUD o de un CRUD grid. Por lo general, se trata de **interfaces HTML**. Un framework CRUD demanda varios pasos de transacción, de forma que los datos no se recogen una vez se han introducido, sino que es necesario pulsar la opción “Guardar” o “Continuar”.

Frameworks CRUD: capa de acceso a las bases de datos

- Los frameworks CRUD facilitan el acceso al sistema de bases de datos y son utilizados tanto en el desarrollo como en el uso de aplicaciones. Existen numerosos frameworks con un concepto CRUD basados en diferentes lenguajes y plataformas. A continuación, presentamos algunos ejemplos:

Frameworks CRUD: capa de acceso a las bases de datos

Lenguaje o plataforma	Framework
Java	JDBC (The Java Database Connectivity), Hibernate, JBoss Seam, Isis
PHP	Yii, CakePHP, Zikula, Symfony, TYPO3 Flow
Perl	Catalyst, Gantry
Python	Django, SQLAlchemy, web2py
Groovy	Grails
.NET	NHibernate, ADO.NET/Entity Framework
Ruby	Ruby on Rails
JavaScript	Backbone.js, AngularJS

Cómo desarrollar un CRUD PHP grid para tu base de datos

- 1. El primer paso consiste en crear una tabla de base de datos simple que pueda ser manipulada con operaciones CRUD a lo largo de este tutorial. Para ello, importa la siguiente tabla de ejemplo en tu base de datos MySQL

Cómo desarrollar un CRUD PHP grid para tu base de datos

```
1 CREATE TABLE `customers` (  
2   `id` INT NOT NULL AUTO_INCREMENT PRIMARY KEY ,  
3   `name` VARCHAR( 100 ) NOT NULL ,  
4   `email` VARCHAR( 100 ) NOT NULL ,  
5   `mobile` VARCHAR( 100 ) NOT NULL  
6 ) ENGINE = INNODB;
```

- La tabla se utiliza para recopilar información del usuario como nombre, correo electrónico y número de teléfono. A cada entrada se le asigna automáticamente una **clave principal** (*AUTO_INCREMENT PRIMARY KEY*), es decir un identificador único.

Cómo desarrollar un CRUD PHP grid para tu base de datos

- 2. A continuación, es necesario regular las conexiones de apertura y de cierre de la base de datos. Crea un archivo PHP con el nombre *database.php* y añade la siguiente secuencia de comandos con la clase “Database” para gestionar las conexiones de la base de datos.

Cómo desarrollar un CRUD PHP grid para tu base de datos

```
public static function connect() {  
    // Permitir solo una conexión para la totalidad del acceso  
    if ( null == self::$cont )  
    {  
        try  
        {  
            self::$cont = new  
PDO("mysql:host=".self::$dbHost.";".self::$dbName,  
self::$dbUsername, self::$dbUserPassword);  
        }  
        Catch (PDOException $e)  
        {  
            die($e->getMessage());  
        }  
    }  
    return self::$cont;  
}  
  
public static function disconnect()  
{  
    self::$cont = null;  
}  
}
```

Cómo desarrollar un CRUD PHP grid para tu base de datos

- Para que puedas utilizar la clase definida en el documento y lograr el acceso a la base de datos con PDO, necesitas los valores exactos de los elementos ***\$dbName*** (nombre de la base de datos utilizada), ***\$dbHost*** (nombre del host donde se ejecutará la base de datos, por lo general [localhost](#) como en el ejemplo), ***\$dbUsername*** (nombre del usuario que accede) y ***\$dbUserPassword*** (contraseña del usuario que accede).