

## DATABASE

Il database e' un archivio organizzato.

Sono divisi in due macrofamiglie, **database organizzati relazionali** (che useremo) e **database non relazionali**.

### MySQL

MySQL è un *RDMS (Relational Database Management System)* sarà il sistema di gestione dei database che utilizzeremo.

Al suo interno ci sono avremo tanti database al cui interno sono contenute delle tabelle divise in colonne e righe.

Le **colonne** sono le **caratteristiche** di un'entità (*utente singolo*) come nome, email, pass ecc.

Le **righe** sono gli *utenti singoli* o **record**.

La colonna **id** e' un tipo di campo che permette di avere ogni *record univoco*, accetta valori numerici integrali e si incrementa da solo.

In caso di eliminazione di un utente, quell'*id* non viene piu' utilizzato dal database, non viene rimpiazzato dal valore successivo o precedente.

Per parlare la stessa lingua del database si utilizza il **SQL**.

Laravel funziona sulla **Convention Over Configuration**. Grazie all' **ORM (object relational mapping)** utilizza e gestisce gli oggetti che creeremo e si interfacerà con il database tramite dei Query.

L'ORM di Laravel e' **Eloquent**.

## CREARE UN DATABASE

Nella bash utilizziamo il linguaggio MySql per creare un nuovo database:

<code>mysql -u root -p</code>	<i>Ci interfacciamo con MySql (password rootroot)</i>
<code>mysql&gt; show databases;</code>	Mostra i database
<code>mysql&gt; use blog;</code>	Entra nel database chiamato (per esempio) blog
<code>mysql&gt; use tables;</code>	Mostrami le tabelle.
<code>mysql&gt; select * from users;</code>	Seleziona tutto* dalla tabella users
<code>mysql&gt; create database article;</code>	Crea database di nome <i>article</i>
<code>mysql&gt; exit / quit;</code>	Uscire da MySql o tieni TAB aperta

## SETTING DEL DATABASE IN LARAVEL

Nel file `.env` modifichiamo i setting del DB (*database*) con i dati corrispondenti appena creati (*database, username e password*).

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=nomeDatabase
DB_USERNAME=root
DB_PASSWORD=root
```

## MIGRATION

Se lavoriamo su un progetto con altre persone dovremo condividere insieme ad altri la struttura del database: per farlo creiamo una **migration**.

La *migration* e' il file che mi permette di *creare* e *modificare* le tabelle, e lo creiamo dalla bash:

```
php artisan make:migration create_articles_table
```

**NB:** I nomi delle tabelle si usano al plurale, e la sintassi va eseguita come nell'esempio, altrimenti non segui la Convention Over Configuration, e tutti i benefici che porta.

Abbiamo creato una cartella in `app/database/migrations/` con dei file per le tabelle di Users e Password, e il nostro file.

```
class CreateArticlesTable extends Migration{

    public function up(){
        Schema::create('articles', function (Blueprint $table){
            $table->id();
            $table->string('name');
            $table->string('name');
            $table->text('name');
            $table->timestamps();
        });
    }

    public function down(){
        Schema::dropIfExists('articles');
    }
}
```