

MODEL

Il **model** e' la parte mancante dell'MVC (Model View Control) su cui e' basato Laravel. Tramite il **model** possiamo istruire il progetto a creare e memorizzare i **record**.

php artisan make:model Article . Creiamo un file *Article.php* in *app/Models/* dalla bash.

Nel controller:

```
class ArticleController extends Controller{
    public function submit(Request $request){

        $article = Article::create([
            'title' => $request->input('title'),
            'author' => $request->input('author'),
            'body' => $request->input('body')
        ]),

        // $article = new Article();
        // $article->title = $request->input('title');
        // $article->author = $request->input('author');
        // $article->body = $request->input('body');
        // $article->save();

        return redirect(route()->back()->with('status', 'messaggio
                                                    inviato!'));
    }
}
```

save() . L'articolo viene processato da *Eloquent* che vede i *fillable* e in automatico mette gli articoli nella colonna corrispondente grazie alla Convention Over Configuration.

back() . Redirect alla pagina precedente.

with() . Aggiungiamo all'HTML della pagina un *if/endif* dalla documentazione (*Redirecting with Flashed Session Data*) che verrà richiamato dalla funzione *view* e mostrerà all'utente la stringa nel secondo parametro.

Article::create . Con il *Mass Assignment* possiamo salvare un nuovo modello di classe *Article* senza aver paura di incorrere in errori. Se in input vi e' un valore non e' presente nel *fillable* non verrà preso in considerazione.

Article.php :

```
class Article extends Model{
    use HasFactory;
    protected $fillable = [
        'title',
        'author',
        'body'
    ];
}
```