

레거시 애플리케이션에 Eloquent ORM 사용하기

Modern PHP User Group 3월 정기 모임
이현석

ORM

- Object-Relational Mapping
- OOP 언어와 데이터를 다루는 RDBMS 와의 상이한 시스템을 매핑하여, 쉽게 데이터 관련 OOP 프로그래밍을 쉽게 하도록 하기 위한 프로그래밍 기법
- ORM은 RDB를 사용해야 한다는 것을 인정하고 RDB를 객체 지향적으로 사용할 수 있도록 중간 역할을 하기 위해 만들어 짐.

Eloquent ORM

라라벨에 포함된 Eloquent ORM 은 아름답고, 심플한 액티브 레코드를 제공합니다. 각 데이터베이스 테이블은 테이블과 상호작용하는 “Model” 을 갖습니다.

테이블 - 모델

설치

- `composer require illuminate/database`
- `composer require jeremeamia/superclosure doctrine/dbal illuminate/console illuminate/events illuminate/filesystem`

레거시 애플리케이션

- **오토로드 설정이 안되어 있다.**
- 테이블명이 제각각이다.
- Primary key 컬럼명이 제각각이다.
- Timestamp에 대한 컬럼명이 제각각이다.

엘로퀀트 ORM 셋업

아래 내용을 셋업 파일에 추가해주세요. (메뉴얼 참고)

별도의 셋업 파일이 없는 경우 만들어줍니다.

여기서는 루트폴더에 setup.php 파일을 만드는 것으로 가정하겠습니다.

```
<?php
use Illuminate\Database\Capsule\Manager;

include_once "vendor/autoload.php";

$capsule = new Manager();

$capsule->addConnection([
    'driver'    => 'mysql',
    'host'      => 'localhost',
    'database'  => 'pug',
    'username'  => 'root',
    'password'  => 'root',
    'charset'   => 'utf8',
    'collation' => 'utf8_unicode_ci',
    'prefix'    => '',
]);

// Set the event dispatcher used by Eloquent models... (optional)
use Illuminate\Events\Dispatcher;
$capsule->setEventDispatcher(new Dispatcher(new Container));

// Make this Capsule instance available globally via static methods... (optional)
$capsule->setAsGlobal();

// Setup the Eloquent ORM... (optional; unless you've used setEventDispatcher())
$capsule->bootEloquent();
```

셋업 파일 오토로드

엘로퀀트 ORM을 셋업하는 코드가 포함된 파일을 오토로드 합니다.
이를 통해 어디서든 엘로퀀트 ORM을 사용할 수 있게 됩니다.
composer.json 에 autoload 키를 아래와 같이 추가해주세요.

```
{
  "require": {
    "illuminate/database": "~5.0",
    "jeremeamia/superclosure": "~2.0",
    "doctrine/dbal": "~2.5",
    "illuminate/console": "~5.0",
    "illuminate/events": "~5.0",
    "illuminate/filesystem": "~5.0"
  },
  "autoload": {
    "files": [
      "setup.php"
    ]
  }
}
```


클래스 오토로드

모델 클래스를 모아두는 폴더를 오토로드 하세요.
이를 통해 모델 클래스를 어디서든 사용할 수 있게 됩니다.
여기서는 모델 클래스를 classes 폴더에 모아둔다고 가정합니다.

```
{
  "require": {
    "illuminate/database": "~5.0",
    "jeremiamia/superclosure": "~2.0",
    "doctrine/dbal": "~2.5",
    "illuminate/console": "~5.0",
    "illuminate/events": "~5.0",
    "illuminate/filesystem": "~5.0"
  },
  "autoload": {
    "psr-4": {
      "": "classes/"
    },
    "files": [
      "setup.php"
    ]
  }
}
```

Composer 오토로드 갱신

- composer.json 의 autoload 키가 수정되었으므로 오토로드 정보를 갱신해줍니다.
- `php composer.phar dump`

레거시 애플리케이션

- 오토로드 설정이 안되어 있다.
- 테이블명이 제각각이다.
- Primary key 컬럼명이 제각각이다.
- Timestamp에 대한 컬럼명이 제각각이다.

기본 테이블명 규칙

- Eloquent ORM 은 기본적으로 모델명이 복수형을 테이블명으로 가정합니다.
- User 모델은 users 테이블을 사용한다고 가정
- UserCondition 모델은 user_conditions 테이블

테이블명 명시

테이블명을 명시해줌으로써 기존에 사용하던 DB 테이블명 수정없이 Eloquent ORM 을 사용할 수 있습니다.

```
class User extends Model {  
  
    protected $table = 'my_users';  
  
}
```

레거시 애플리케이션

- 오토로드 설정이 안되어 있다.
- 테이블명이 제각각이다.
- **Primary key 컬럼명이 제각각이다.**
- Timestamp에 대한 컬럼명이 제각각이다.

기본 primary key 규칙

- Eloquent ORM 은 기본적으로 primary key 컬럼명을 id 로 가정합니다.

primary key 컬럼명 명시

primary Key 컬럼명을 명시해줌으로써 기존에 사용하던 DB 테이블의 컬럼명 수정없이 Eloquent ORM 을 사용할 수 있습니다.

```
class User extends Model {  
  
    protected $table = 'my_users';  
  
    protected $primaryKey = 'idx';  
  
}
```


레거시 애플리케이션

- 오토로드 설정이 안되어 있다.
- 테이블명이 제각각이다.
- Primary key 컬럼명이 제각각이다.
- Timestamp에 대한 컬럼명이 제각각이다.

기본 Timestamp 컬럼명 규칙

- Eloquent ORM 은 created_at 컬럼과 updated_at 컬럼을 이용합니다.

Timestamp 컬럼명 명시(권장)

timestamp 컬럼명을 명시해줌으로써 자동으로 생성시점 및 갱신시점을 관리할 수 있습니다.

```
class User extends Model {  
  
    const CREATED_AT = 'reg_date';  
  
    const UPDATED_AT = 'mod_date';  
  
    const DELETED_AT = 'del_date';  
  
}
```

Timestamp 끄기

timestamp 옵션을 끌 수 있습니다.

이 경우 생성시점이나 업데이트 시점을 수동으로 관리해야 합니다.

```
class User extends Model {  
  
    protected $timestamps = false;  
  
}
```

Timestamp 컬럼명 추가

Eloquent ORM이 사용하는 기본 timestamp 컬럼명을 추가해주는 방법을 사용할 수도 있습니다.

기존에 timestamp를 관리하지 않았을 경우에 적합하다고 생각됩니다.

감사합니다.

Modern PHP 사용자 그룹

이현석

트위터 @smartbosslee

이메일 smartbosslee@gmail.com