

23) Eloquent Relationship & Serialization

Database Table နှစ်ခု ချိတ်ဆက်တဲ့အခါ တခုနဲ့တခုမီခိုနေတဲ့ပုံစံဖြစ်ပါတယ်။

Database Foreign Key

Foreign key ကတော့ တခြား tableတခုမှ primary key ဖြစ်နေတဲ့ Column field ကို reference ယူထားတဲ့ key ဖြစ်ပါတယ်။

Advantage of Foreign Key

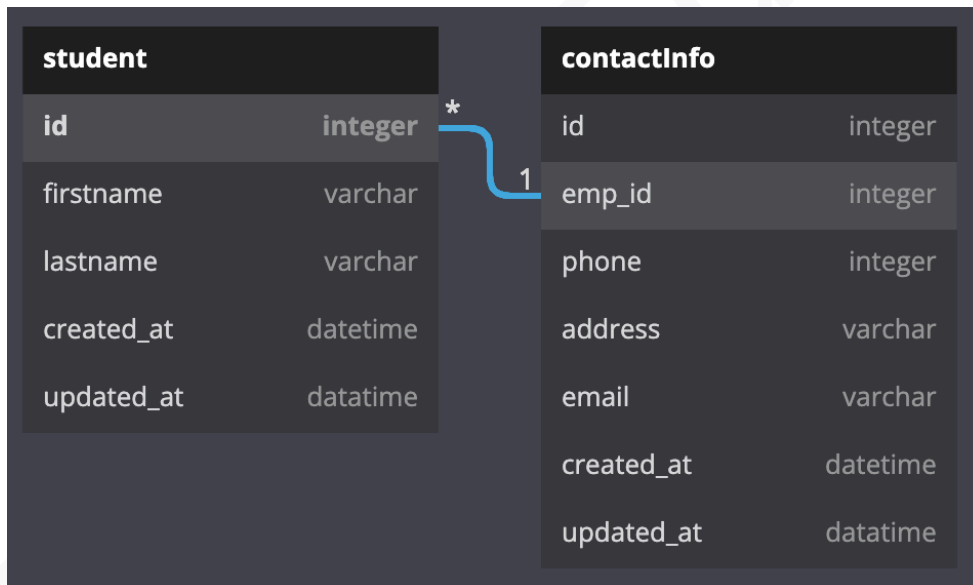
orphan record တွေမဖြစ်အောင် ကာကွယ်ပေးပါတယ်။

Disadvantage of Foreign Key

Database CPU Loading

**Note>> Restrict,Cascade,Set null
Must be order**

One to One Relationship



Student table နဲ့ ContactInfo table ရှိတယ်လို့ဆိုကြပါစို့။

Student ဆိုတာ ContactInfo တခုတည်းရှိသလို ContactInfo ဆိုတာလည်း Student တယောက်ပဲရှိပါတယ်။ ဒီလိုမျိုး ဆက်ဆက်နေတဲ့ relationship ကို one to one relationship လို့ခေါ်ပါတယ်။

Create Student Table (Migration)

```
Schema::create(students, function (Blueprint $table) {  
    $table->id();  
    $table->string("firstname");  
    $table->string("lastname");  
    $table->timestamps();  
});
```

Create ContactInfo Table (Migration)

```
Schema::create(contact_infos, function (Blueprint $table) {  
    $table->id();  
    $table->unsignedBigInteger("student_id");  
    $table->integer("phone");  
    $table->string("address");  
    $table->string("email");  
    $table->timestamps();  
    $table->foreign('student_id')->references('id')->on(students)  
        ->onDelete('cascade');  
});
```

ContactInfo Model

ContactInfo မှာ Student ရဲ့ ID နဲ့ချိတ်ဆက်ထားတာကြောင့် တနည်းအားဖြင့်

ContactInfo data ဟာ Student ရှိမှ တည်ဆောက်လို့ရတာကြောင့် belongTo နဲ့ Student

Model ကို ချိတ်ဆက်ပေးရပါတယ်။

```
public function student()  
{  
    return $this->belongsTo('App\Models\Student');  
}
```

Student Model

Student မှာ Contact တခုပဲရှိတာကြောင့် hasOne နဲ့ Contact ကို ချိတ်ပေးရပါတယ်။

```
public function conntactInfo()  
{  
    return $this->hasOne('App\Models\ContactInfo');  
}
```

Inserting Query with eloquent

```
//insert first student
$student = new Student();
$student->firstname = "John";
$student->lastname = "Smith";
$student->age = 23;
$student->save();

//prepare insert contactInfo
$contact = new ContactInfo();
$contact ->phone = 91234567;
$contact ->email = "john@gmail.com";
$contact ->address = "Yangon";

//save contactInfo and add student id in foreign key
$student->contactInfo()->save($contact);
```

Retrieving Data

```
public function show($id)
{
    $student = Student::find($id)->contactInfo;

    return $student;
}
```

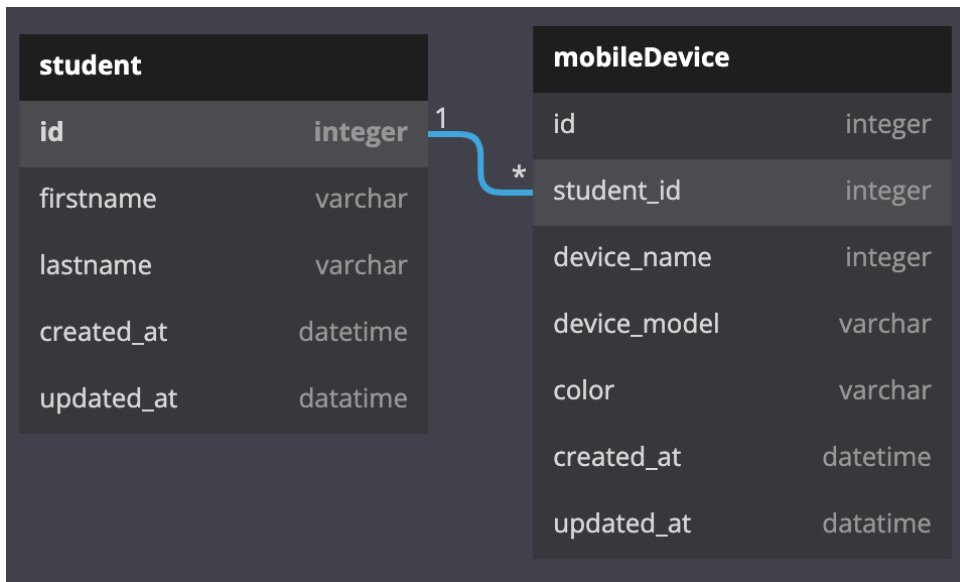
Updating Data

```
$student = Student::find($id);

$student->age = 20;
$student->contactInfo->address = "Yangon";

$student->save();
$student->contactInfo->save();
```

One to Many Relationship



Student table နဲ့ mobileDevice table ရှိတဲ့အခါ Student တယောက်မှာ အသုံးပြုတဲ့ device ဟာ တခုထက်မက ပိုရှိနိုင်ပါတယ်။ ဒီလိုမျိုး relationship ကို one to many relationship လို့ သတ်မှတ်ပါတယ်။

MobileDevice Model

MobileDevice မှာတော့ Student ကို belongsTo နဲ့ချိတ်ပေးရပါတယ်။

```
public function student()  
{  
    return $this->belongsTo('App\Models\Student');  
}
```

Student Model

Student မှာ MobileDevice တခုထက်ပိုပြီးရှိနိုင်တာကြောင့် hasMany နဲ့ Device ကို ချိတ်ပေးရပါတယ်။

```
public function device()  
{  
    return $this->hasMany('App\Models\Device');  
}
```

Inserting Query with eloquent

```
//insert first student
$student = new Student();
$student->firstname = "John";
$student->lastname = "Smith";
$student->age = 23;
$student->save();

//prepare insert device1
$device1 = new MobileDevice();
$device1->deviceName = "iphone";
$device1->deviceModel = "7";
$device1->color = "Red";

//prepare insert device2
$device2 = new MobileDevice();
$device2->deviceName = "Samsung";
$device2->deviceModel = "Fold";
$device2->color = "White";

//save devices and add student id in foreign key
$student->mobileDevice()->saveMany([$device1,$device2]);
```

Retrieving Data

```
public function show($id)
{
    $student = Student::find($id)->mobileDevice;
}
```