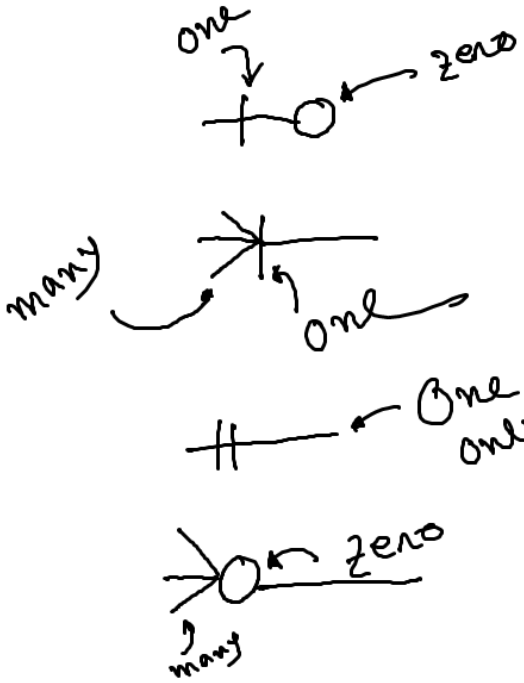
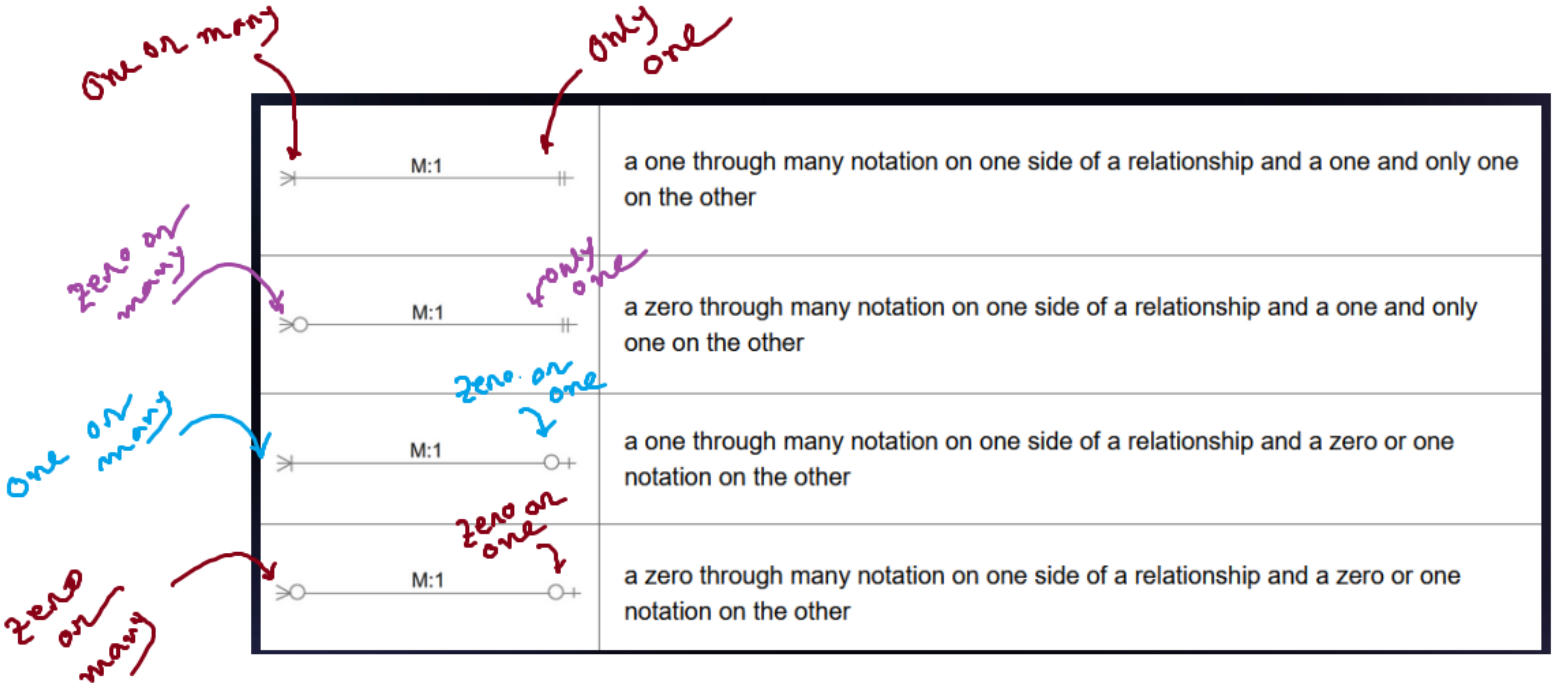
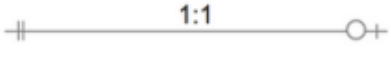
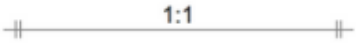
## RELATIONSHIP SIGN

| | |
|---|---|
|  | Zero or One |
|  | One or More |
|  | One and only One |
|  | Zero or More |



| | |
|---|---|
| | a zero through many on both sides of a relationship |
| | a zero through many on one side and a one through many on the other |
| | a one through many on both sides of a relationship |
|   | a one and only one notation on one side of a relationship and a zero or one on the other |
|   | a one and only one notation on both sides |

# Relationship Constraint:

## Foreign Key with Cascade on Delete

```php
$table->foreignId('user_id')
    ->constrained('users') // Foreign key reference to the `users` table
    ->cascadeOnDelete(); // Deletes related orders when a user is deleted
```

## Foreign Key with Restrict on Delete

```php
$table->foreignId('user_id')
    ->constrained('users')
    ->restrictOnDelete(); // Prevents deletion of user if related orders exist
```

## Foreign Key with Set Null on Delete (must be nullable)

```php
$table->foreignId('user_id')
    ->nullable()
    ->constrained('users')
    ->setNullOnDelete(); // Sets user_id to NULL when the user is deleted
```

## Foreign Key with No Action on Delete

```php
$table->foreignId('user_id')
    ->constrained('users')
    ->noActionOnDelete(); // No action (manual handling needed)
```

## Foreign Key with Cascade on Update

```php
$table->foreignId('user_id')
    ->constrained('users')
    ->cascadeOnUpdate(); // Updates related orders when a user's ID is updated
```

## Foreign Key with Restrict on Update

```php
$table->foreignId('user_id')
    ->constrained('users')
    ->restrictOnUpdate(); // Prevents updating user ID if related orders exist
```

## Foreign Key with Set Null on Update

```php
$table->foreignId('user_id')
    ->nullable()
    ->constrained('users')
    ->setNullOnUpdate(); // Sets user_id to NULL when the user ID is updated
```

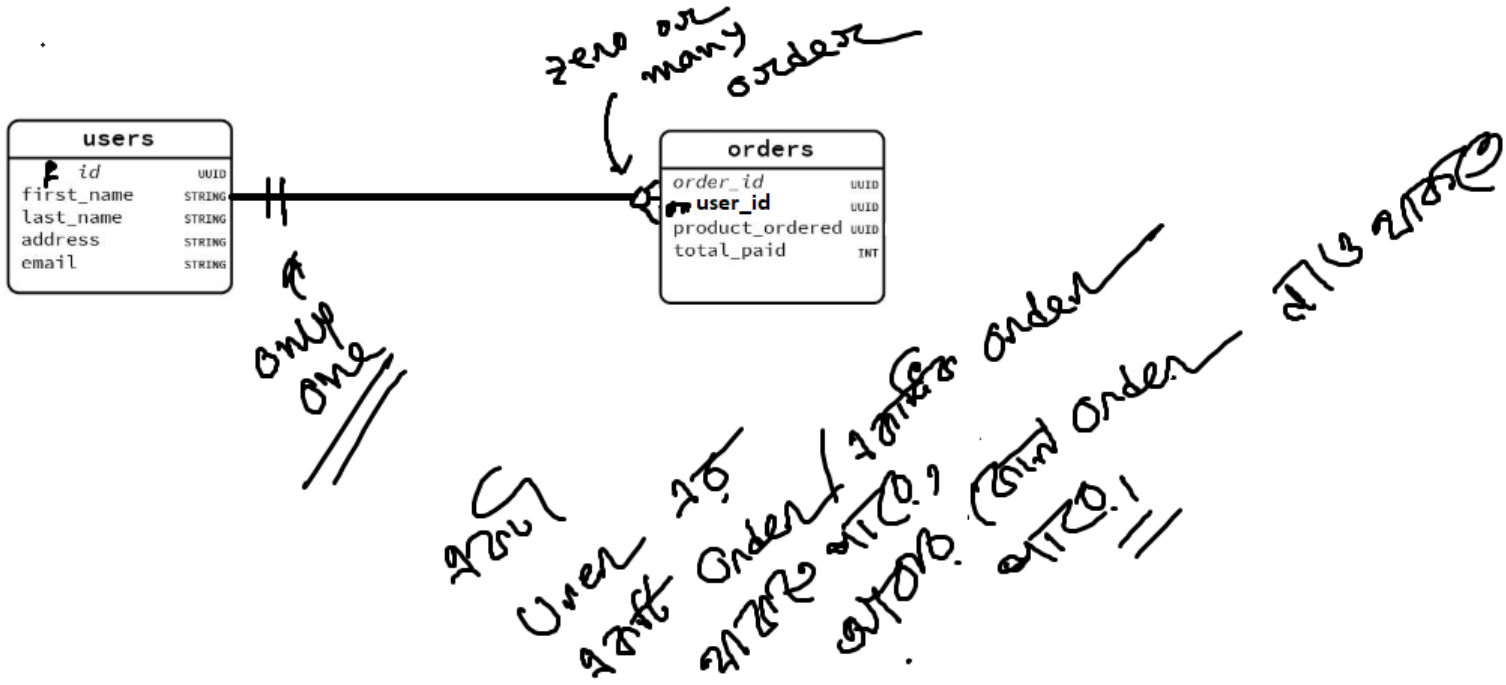
## Foreign Key with No Action on Update

```php
$table->foreignId('user_id')
    ->constrained('users')
    ->noActionOnUpdate(); // No action (manual handling needed)
```

# Relation এর জন্য Foreign Key কিভাবে সেট করব:

১.যদি একটা টেবিল Id column(Unsigned Big Integer Primary Key) অন্য টেবিল এর Foreign Key হিসেবে ব্যবহৃত হয় সেক্ষেত্রে তার Syntax:

**$table->foreignId('user_id') ->constrained('users_infos') ->restrictOnDelete()->cascadeOnUpdate()  এটা ব্যবহার করব**




এখানে users table এর id column(primary key)  orders table এ foreign key হিসেবে ব্যবহৃত হচ্ছে এখানে কিভাবে টেবিল দুটি বানাব।

১. প্রথমে Migration বানাব:

```
User@DESKTOP-TSJIMAQ MINGW64 /c/xampp/htdocs/module_17_migration
$ php artisan make:migration create_users_infos_table
php] created successfully.


User@DESKTOP-TSJIMAQ MINGW64 /c/xampp/htdocs/module_17_migration
$ php artisan make:migration create_users_infos_table
```

২. Migration Code বানালাম:

Users_infos

```php
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

return new class extends Migration {
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('users_infos', function (Blueprint $table) {
            $table->id();
            $table->string('first_name', 50);
            $table->string('last_name', 50);
            $table->string('email')->unique(); // Unique constraint
            $table->string('address', 255);
            $table->timestamp('created_at')->useCurrent(); // Default to CURRENT_TIMESTAMP
            $table->timestamp('updated_at')->useCurrent()->useCurrentOnUpdate(); // Updates to CURRENT_TIMESTAMP on update
        });
    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('users_infos');
    }
};

```

Stores Table(Foreign Key) foreignId('column name') এটা ব্যবহার করব

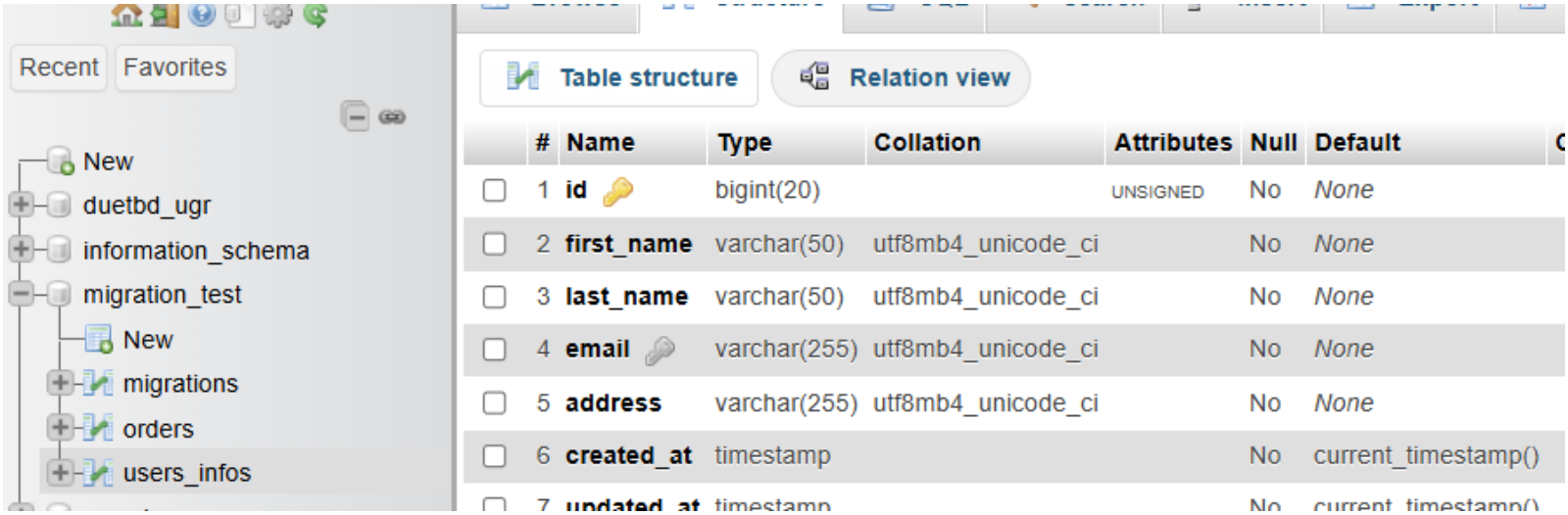
```
// Foreign key referencing the users_infos table
        $table
            ->foreignId('user_id')
            ->constrained('users_infos') // Reference to the users_infos table
            ->restrictOnDelete() // Prevent deletion of the parent while child rows exist
            ->cascadeOnUpdate(); // Update child rows when the parent row is updated
```

```php
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

return new class extends Migration {
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('orders', function (Blueprint $table) {
            $table->id(); // Primary key for the orders table

            // Foreign key referencing the users_infos table
            $table
                ->foreignId('user_id')
                ->constrained('users_infos') // Reference to the users_infos table
                ->restrictOnDelete() // Prevent deletion of the parent while child rows exist
                ->cascadeOnUpdate(); // Update child rows when the parent row is updated

            $table->string('product_order', 255); // Name of the product
            $table->decimal('total_paid', 10, 2); // Decimal field for monetary values
            $table->timestamp('created_at')->useCurrent(); // Default to CURRENT_TIMESTAMP
            $table->timestamp('updated_at')->useCurrent()->useCurrentOnUpdate();// Updates to CURRENT_TIMESTAMP
        });

    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('orders');
    }
};

```

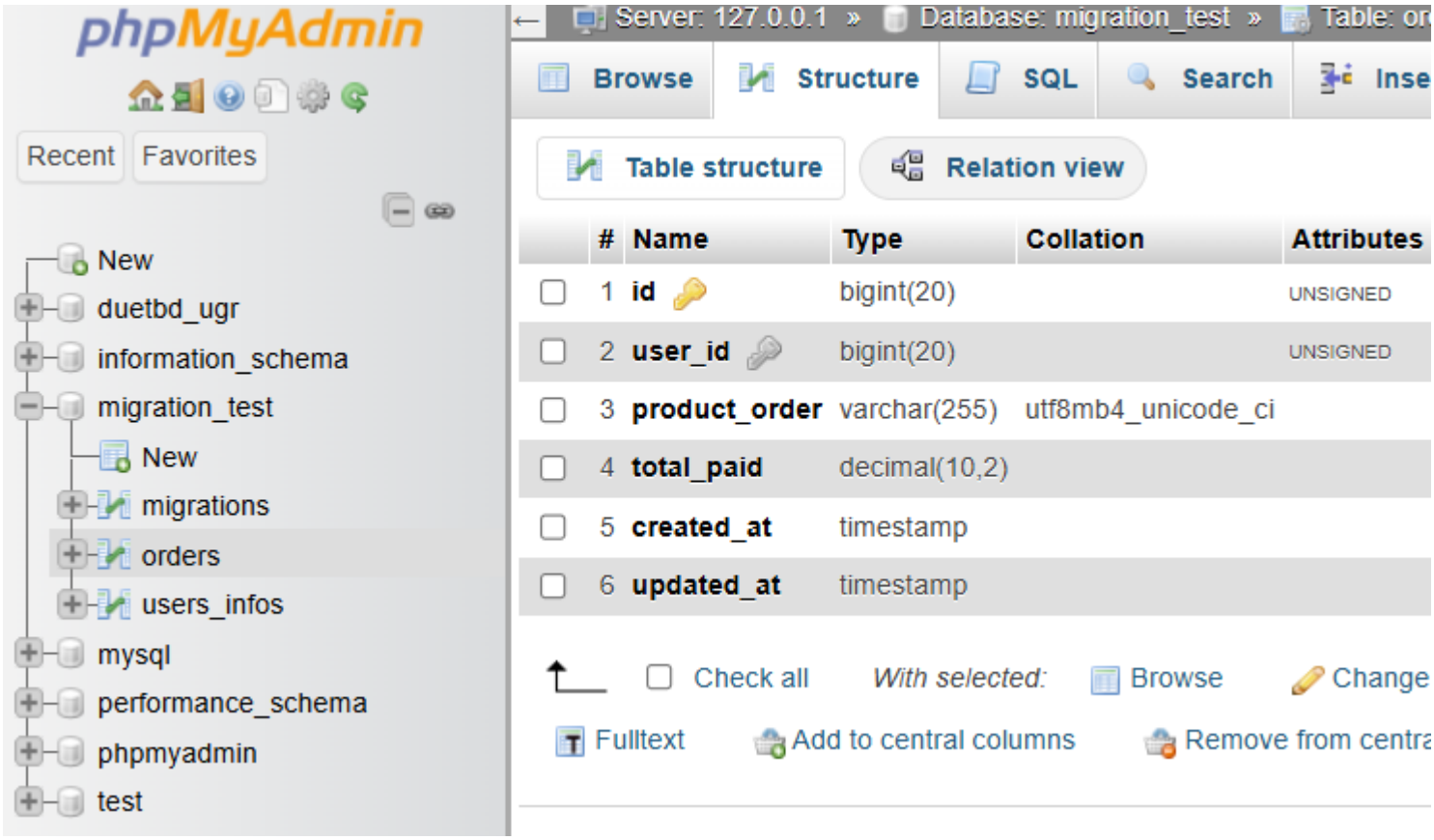
৩.ডাটাবেসে Store করলাম:

```
User@DESKTOP-TSJIMAQ MINGW64 /c/xampp/htdocs/module_17_migration
$ php artisan migrate

  INFO  Running migrations.

  2024_12_14_065431_create_users_infos_table ..... 44.08ms DONE
  2024_12_14_065432_create_orders_table ..... 95.88ms DONE

  INFO  Running migrations.

  2024_12_14_065431_create_users_infos_table ..... 44.08ms DONE
  2024_12_14_065432_create_orders_table ..... 95.88ms DONE

  2024_12_14_065431_create_users_infos_table ..... 44.08ms DONE
  2024_12_14_065432_create_orders_table ..... 95.88ms DONE
```

User_infos

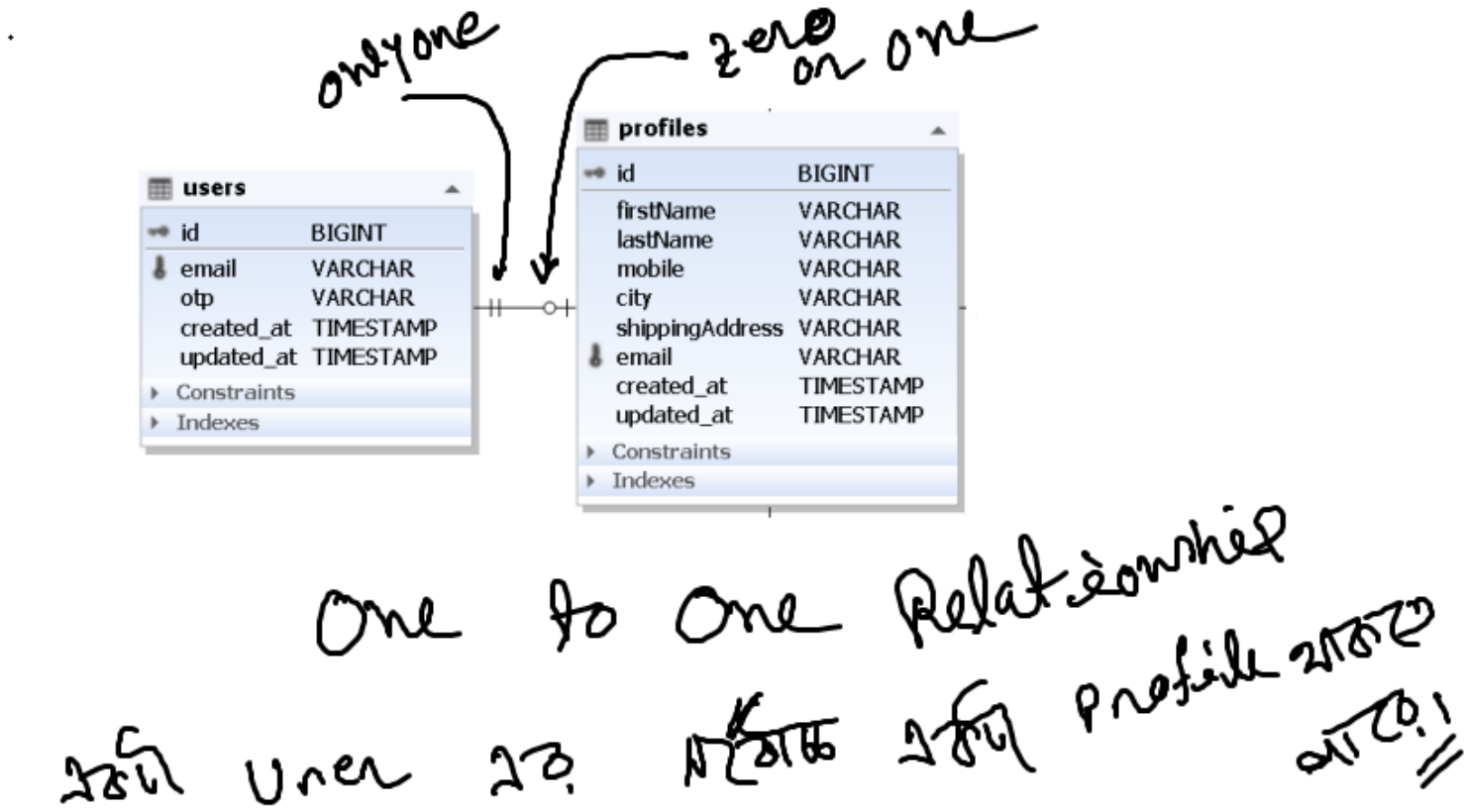



Orders



# Relation এর জন্য Foreign Key কিভাবে সেট করব:

২.যদি একটা টেবিল Id column Unsigned Big Integer/Numeric না হয়ে অন্য কিছু যেমন String(Email) অন্য টেবিল এর Foreign Key হিসেবে ব্যবহৃত হয় সেক্ষেত্রে তার Syntax:




**//same as parent table**

**$table->string('email',50)->unique();**

**foreign(child_table_column)->references('parent_table_column')->on(parent_table)**

***$table->foreign('email')->references('email')->on('users')***

***->restrictOnDelete()***

***->cascadeOnUpdate();***

১.Migration Create

```
User@DESKTOP-TSJIMAQ MINGW64 /c/xampp/htdocs/module_17_migration
$ php artisan make:migration create_profiles_table

  INFO  Migration [C:\xampp\htdocs\module_17_migration\database\migrations/2024_12_14_083532_create_profiles_table.php] created successfully.

User@DESKTOP-TSJIMAQ MINGW64 /c/xampp/htdocs/module_17_migration
$ php artisan make:migration create_profiles_table

  INFO  Migration [C:\xampp\htdocs\module_17_migration\database\migrations/2024_12_14_083532_create_profiles_table.php] created successfully.

  INFO  Migration [C:\xampp\htdocs\module_17_migration\database\migrations/2024_12_14_083532_create_profiles_table.php] created successfully.
.php] created successfully.
```

Activate Windows

২.code create

Users table

```php
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('users', function (Blueprint $table) {
            $table->id();
            $table->string('email',50)->unique();
            $table->string('otp',10);
            $table->timestamp('created_at')->useCurrent();
            $table->timestamp('updated_at')->useCurrent()->useCurrentOnUpdate();
        });
    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('users');
    }
};

```

Profiles table(Email is unique কারণ email একবার ব্যবহার করা যাবে পুরো টেবিল এ তাই Child Table এ unique রাখতে হবে।)

```php
<?php

use Illuminate\Database\Migrations\Migration;
use Illuminate\Database\Schema\Blueprint;
use Illuminate\Support\Facades\Schema;

return new class extends Migration
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('profiles', function (Blueprint $table) {
            $table->id();
            $table->string('firstName',50);
            $table->string('lastName',50);
            $table->string('mobile',50);
            $table->string('city',50);
            $table->string('shippingAddress',1000);
            //same as parent table
            //here unique means one unique email can hold in full table
            $table->string('email',50)->unique();

            // Relationship
            //foreign(child_table_column)->references('parent_table_column')->on(parent_table)
            $table->foreign('email')->references('email')->on('users')
                ->restrictOnDelete()
                ->cascadeOnUpdate();

            $table->timestamp('created_at')->useCurrent();
            $table->timestamp('updated_at')->useCurrent()->useCurrentOnUpdate();
        });
    }

    /**
     * Reverse the migrations.
     */
    public function down(): void
    {
        Schema::dropIfExists('profiles');
    }
};

```

Database এ store

```
User@DESKTOP-TSJIMAQ MINGW64 /c/xampp/htdocs/module_17_migration
$ php artisan migrate

  INFO  Running migrations.

  2024_12_14_082911_create_users_table ..... 362.86ms DONE
  2024_12_14_083532_create_profiles_table ..... 139.48ms DONE
```

আউটপুট:

Users

Table structure | Relation view

| # | Name | Type | Collation | Attributes | Null | Default | Comments | Extra | Action |
|---|---|---|---|---|---|---|---|---|---|
| 1 | **id** | bigint(20) | | UNSIGNED | No | *None* | | AUTO_INCREMENT | Change |
| 2 | **email** | varchar(50) | utf8mb4_unicode_ci | | No | *None* | | | Change |
| 3 | **otp** | varchar(10) | utf8mb4_unicode_ci | | No | *None* | | | Change |
| 4 | **created_at** | timestamp | | | No | current_timestamp() | | | Change |
| 5 | **updated_at** | timestamp | | | No | current_timestamp() | | ON UPDATE CURRENT_TIMESTAMP() | Change |

Profiles

| # | Name | Type | Collation | Attributes | Null | Default |
|---|---|---|---|---|---|---|
| 1 | **id** | bigint(20) | | UNSIGNED | No | *None* |
| 2 | **firstName** | varchar(50) | utf8mb4_unicode_ci | | No | *None* |
| 3 | **lastName** | varchar(50) | utf8mb4_unicode_ci | | No | *None* |
| 4 | **mobile** | varchar(50) | utf8mb4_unicode_ci | | No | *None* |
| 5 | **city** | varchar(50) | utf8mb4_unicode_ci | | No | *None* |
| 6 | **shippingAddress** | varchar(1000) | utf8mb4_unicode_ci | | No | *None* |
| 7 | **email** | varchar(50) | utf8mb4_unicode_ci | | No | *None* |
| 8 | **created_at** | timestamp | | | No | current_timestamp() |
| 9 | **updated_at** | timestamp | | | No | current_timestamp() |