

## Task 1:

Create a new Laravel project named "MigrationAssignment" using the Laravel command-line interface.

```
hp@LAPTOP-A30EFPRI MINGW64 /d/Ostad php/module-16
$ composer global require laravel/installer
Changed current directory to C:/Users/hp/AppData/Roaming/Composer
Info from https://repo.packagist.org: #StandWithUkraine
./composer.json has been updated
Running composer update laravel/installer
Loading composer repositories with package information
Updating dependencies
Nothing to modify in lock file
Writing lock file
Installing dependencies from lock file (including require-dev)
Nothing to install, update or remove
Generating autoload files
9 packages you are using are looking for funding.
Use the `composer fund` command to find out more!
No security vulnerability advisories found
Using version ^4.5 for laravel/installer

hp@LAPTOP-A30EFPRI MINGW64 /d/Ostad php/module-16
$ laravel new MigrationAssignment

  Laravel

  Creating a "laravel/laravel" project at "./MigrationAssignment"
  Installing laravel/laravel (v10.2.2)
    - Installing laravel/laravel (v10.2.2): Extracting archive
  Created project in D:\Ostad php\module-16\MigrationAssignment
> @php -r "file_exists('.env') || copy('.env.example', '.env');"
  Loading composer repositories with package information
  Updating dependencies
Lock file operations: 108 installs, 0 updates, 0 removals
  - Locking brick/math (0.11.0)
  - Locking dflydev/dot-access-data (v3.0.2)
  - Locking doctrine/inferno (2.0.6)
  - Locking doctrine/lexer (3.0.0)
```

## Task 2:

Within the project, create a new migration file named "create\_products\_table" that will be responsible for creating a table called "products" in the database. The "products" table should have the following columns:

id: an auto-incrementing integer and primary key.

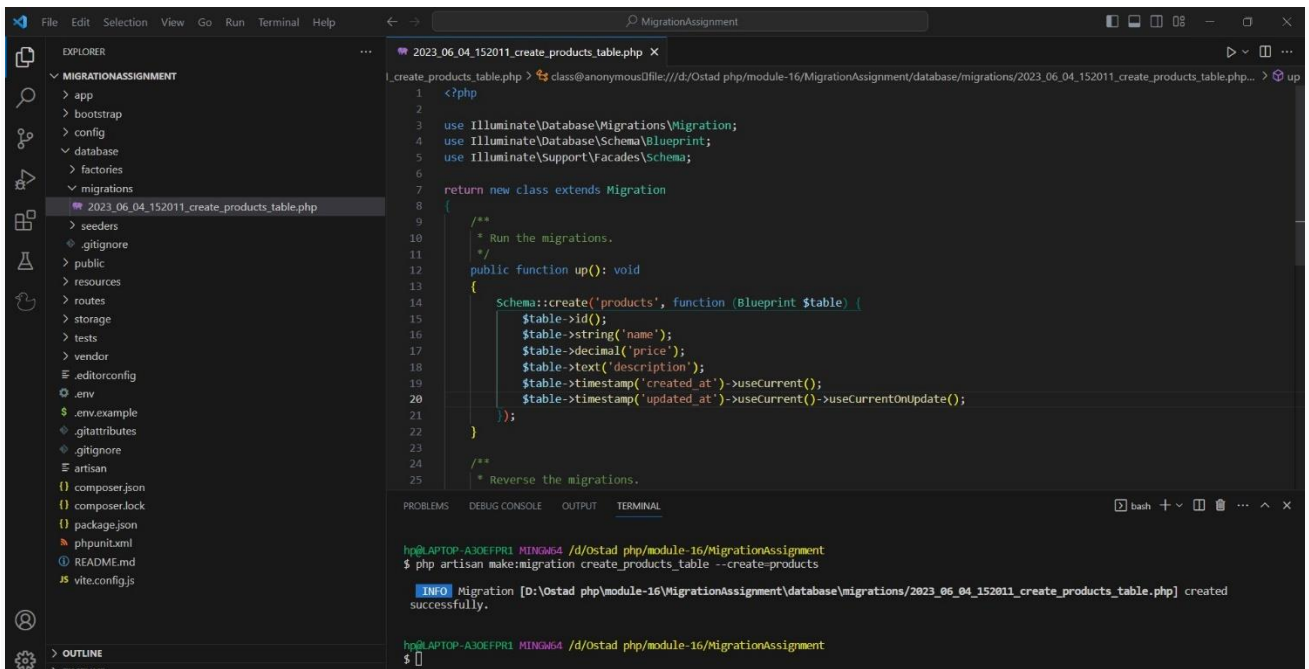
name: a string column to store the product name.

price: a decimal column to store the product price.

description: a text column to store the product description.

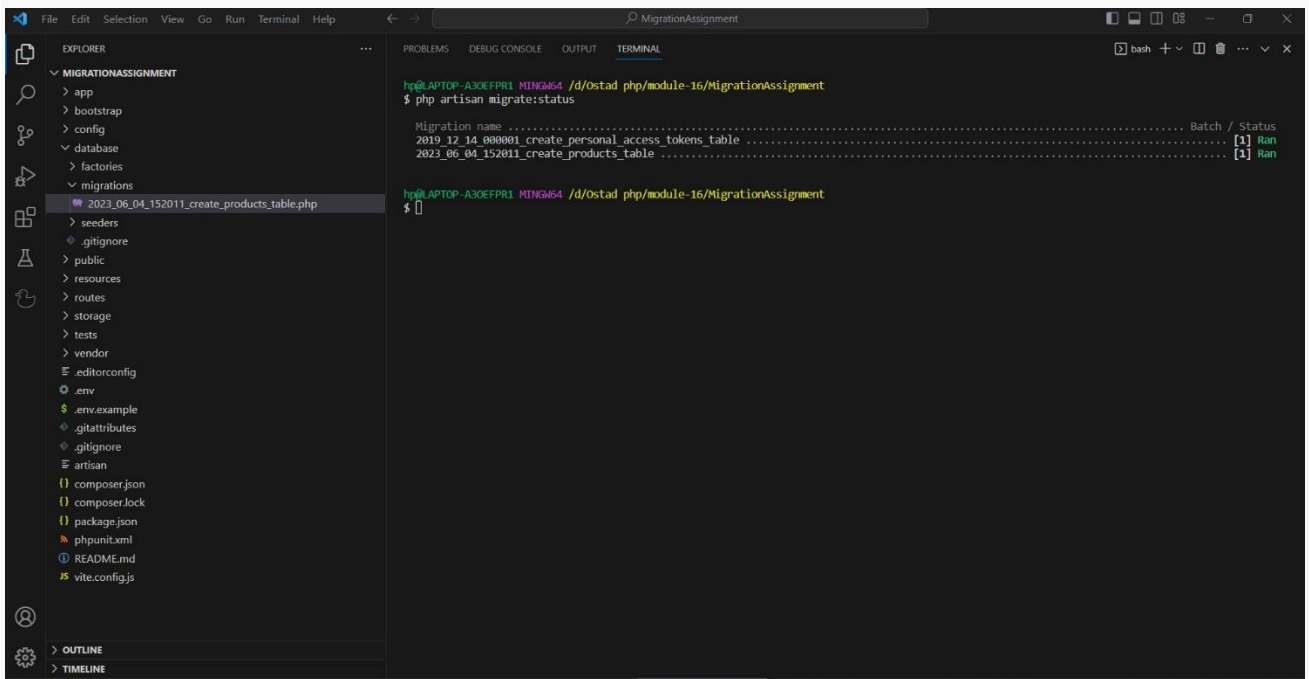
created\_at: a timestamp column to store the creation date and time.

updated\_at: a timestamp column to store the last update date and time.



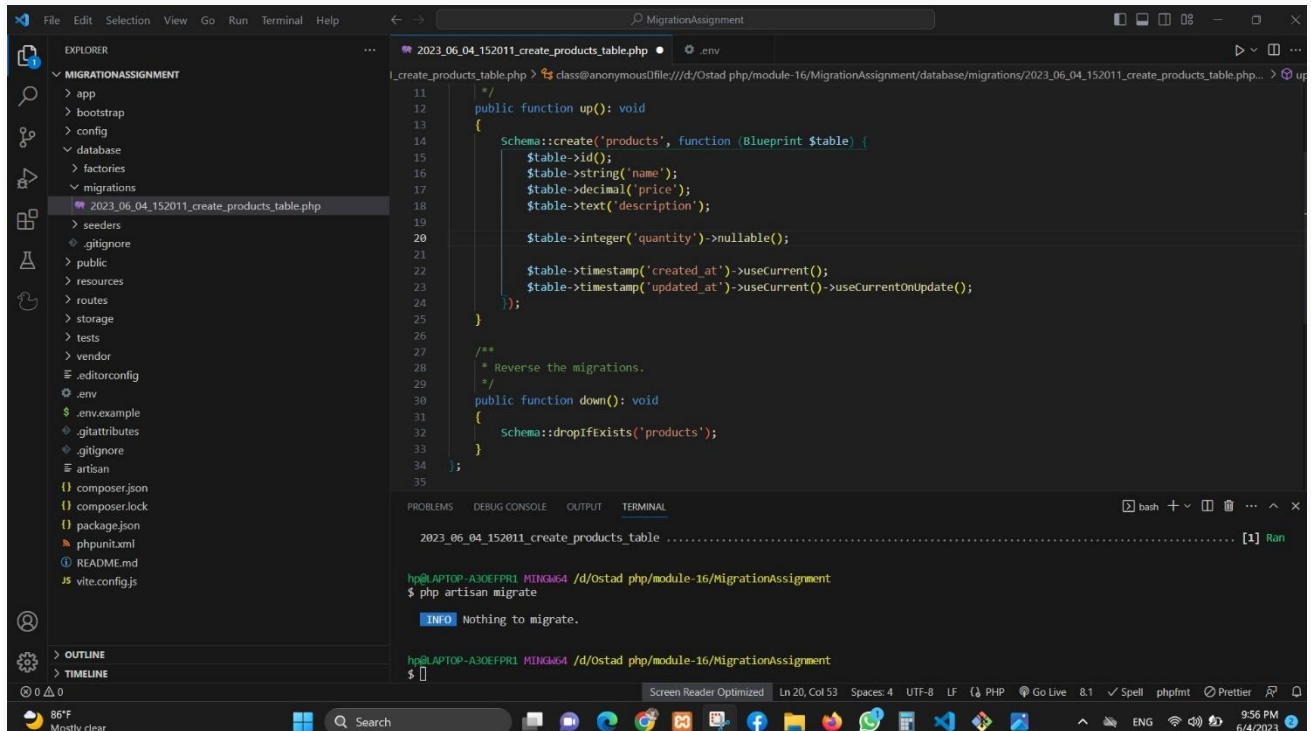
### Task 3:

After creating the migration file, run the migration to create the "products" table in the database.



## Task 4:

Modify the existing migration file "create\_products\_table" to add a new column called "quantity" to the "products" table. The "quantity" column should be an integer column and allow null values.



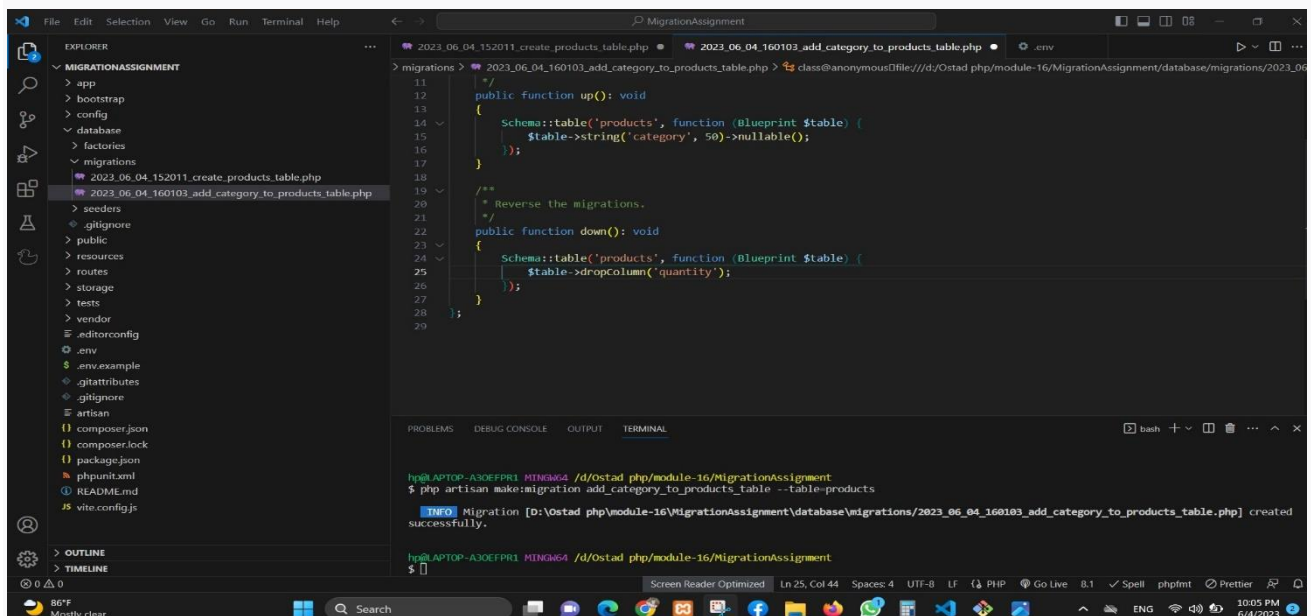
The screenshot shows the VS Code editor with the file explorer on the left displaying the project structure. The main editor window shows the file `2023_06_04_152011_create_products_table.php` with the following code:

```
11  /*
12  13  public function up(): void
14  {
15      Schema::create('products', function (Blueprint $table) {
16          $table->id();
17          $table->string('name');
18          $table->decimal('price');
19          $table->text('description');
20
21          $table->integer('quantity')->nullable();
22
23          $table->timestamp('created_at')->useCurrent();
24          $table->timestamp('updated_at')->useCurrent()->useCurrentOnUpdate();
25      });
26
27  /**
28   * Reverse the migrations.
29   */
30  public function down(): void
31  {
32      Schema::dropIfExists('products');
33  }
34  }
35
```

The terminal at the bottom shows the command `php artisan migrate` being executed, resulting in the output: `Nothing to migrate.`

## Task 5:

Create a new migration file named "add\_category\_to\_products\_table" that will be responsible for adding a new column called "category" to the "products" table. The "category" column should be a string column with a maximum length of 50 characters.



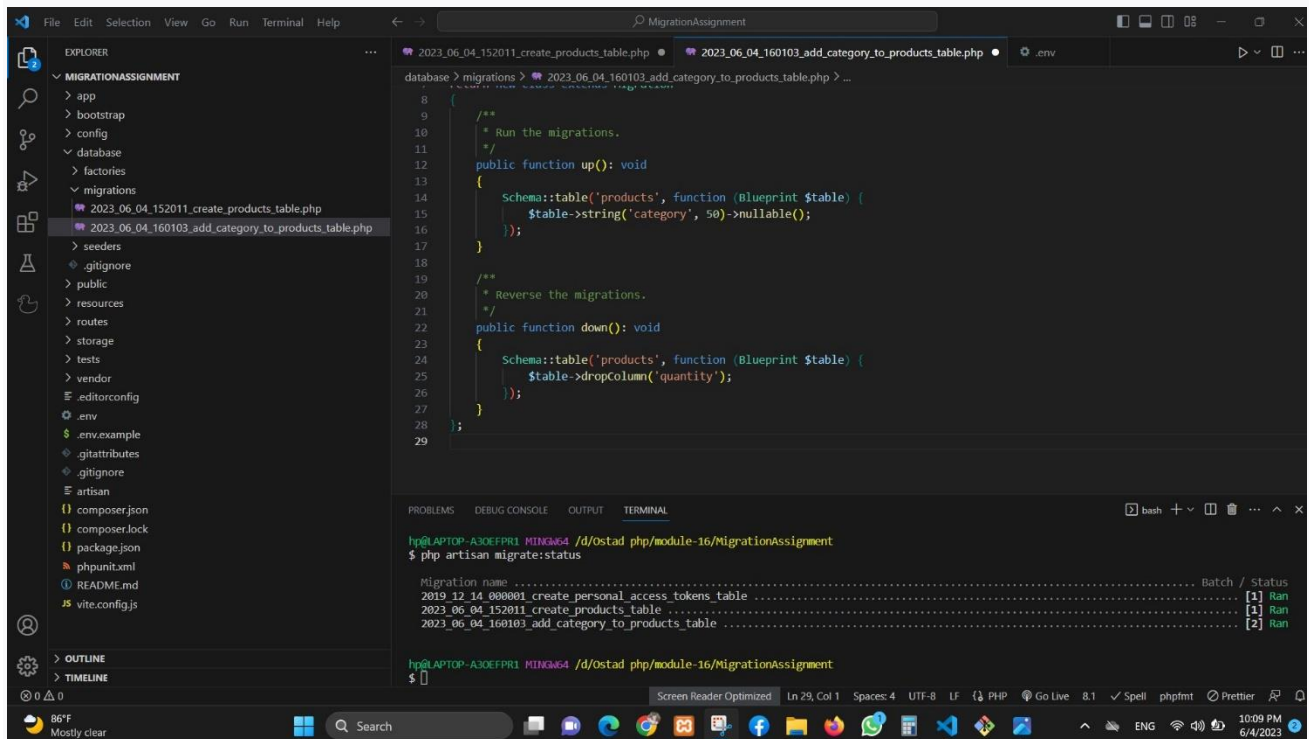
The screenshot shows the VS Code editor with the file explorer on the left. The main editor window shows the new file `2023_06_04_160103_add_category_to_products_table.php` with the following code:

```
11  /*
12  13  public function up(): void
14  {
15      Schema::table('products', function (Blueprint $table) {
16          $table->string('category', 50)->nullable();
17      });
18
19  /**
20   * Reverse the migrations.
21   */
22  public function down(): void
23  {
24      Schema::table('products', function (Blueprint $table) {
25          $table->dropColumn('category');
26      });
27  }
28  }
29
```

The terminal at the bottom shows the command `php artisan make:migration add_category_to_products_table --table=products` being executed, resulting in the output: `Migration [D:\Ostad php\module-16\MigrationAssignment\database\migrations\2023_06_04_160103_add_category_to_products_table.php] created successfully.`

## Task 6:

After creating the new migration file, run the migration to add the "category" column to the "products" table.



The screenshot shows a code editor with a file explorer on the left. The file explorer shows a project structure with folders like 'app', 'bootstrap', 'config', 'database', 'factories', 'migrations', 'seeds', and 'resources'. The 'migrations' folder is expanded, showing two files: '2023\_06\_04\_152011\_create\_products\_table.php' and '2023\_06\_04\_160103\_add\_category\_to\_products\_table.php'. The second file is selected, and its content is displayed in the editor. The code is a Laravel migration that adds a 'category' column to the 'products' table. The code is as follows:

```
8 database > migrations > 2023_06_04_160103_add_category_to_products_table.php > ...
9 {
10     /**
11      * Run the migrations.
12      */
13     public function up(): void
14     {
15         Schema::table('products', function (Blueprint $table) {
16             $table->string('category', 50)->nullable();
17         });
18     }
19
20     /**
21      * Reverse the migrations.
22      */
23     public function down(): void
24     {
25         Schema::table('products', function (Blueprint $table) {
26             $table->dropColumn('category');
27         });
28     }
29 }
```

Below the code editor, the terminal output shows the command 'php artisan migrate:status' and its output:

```
hp@LAPTOP-A30EFPR1 MINGW64 /d/Ostad php/module-16/MigrationAssignment
$ php artisan migrate:status

Migration name ..... Batch / Status
2019_12_14_000001_create_personal_access_tokens_table ..... [1] Ran
2023_06_04_152011_create_products_table ..... [1] Ran
2023_06_04_160103_add_category_to_products_table ..... [2] Ran

hp@LAPTOP-A30EFPR1 MINGW64 /d/Ostad php/module-16/MigrationAssignment
$
```

## Task 7:

Create a new migration file named "create\_orders\_table" that will be responsible for creating a table called "orders" in the database. The "orders" table should have the following columns:

id: an auto-incrementing integer and primary key.

product\_id: an unsigned integer column to establish a foreign key relationship with the "id" column of the "products" table.

quantity: an integer column to store the quantity of products ordered.

created\_at: a timestamp column to store the creation date and time.

updated\_at: a timestamp column to store the last update date and time.

The screenshot shows the VS Code editor with the Explorer sidebar on the left. The 'migrations' folder is expanded, showing three files: '2023\_06\_04\_152011\_create\_products\_table.php', '2023\_06\_04\_160103\_add\_category\_to\_products\_table.php', and '2023\_06\_04\_161450\_create\_orders\_table.php'. The third file is selected and its content is visible in the editor. The code defines an 'up' method for creating the 'orders' table with a foreign key to the 'products' table and a 'down' method to drop the table. The terminal at the bottom shows the command 'php artisan make:migration create\_orders\_table' and its output.

```
150_create_orders_table.php > class<anonymous>File:///d:/Ostad_php/module-16/MigrationAssignment/database/migrations/2023_06_04_161450_create_orders_table.php
8
9
10 /**
11  * Run the migrations.
12  */
13 public function up(): void
14 {
15     Schema::create('orders', function (Blueprint $table) {
16         $table->id();
17
18         // F - K
19         $table->unsignedBigInteger('product_id')->unique();
20
21         // Relationship
22         $table->foreign('product_id')->references('id')->on('products')
23         ->restrictOnDelete()
24         ->restrictOnUpdate();
25
26         $table->integer('quantity');
27
28         $table->timestamp('created_at')->useCurrent();
29         $table->timestamp('updated_at')->useCurrent()->useCurrentOnUpdate();
30     });
31 }
32 /**
33  * Reverse the migrations.
34  */
35 public function down(): void
36 {
37     Schema::dropIfExists('orders');
38 }
```

```
hp@LAPTOP-A30EFPRI MINGW64 /d/Ostad_php/module-16/MigrationAssignment
$ php artisan make:migration create_orders_table

[INFO] Migration [D:/Ostad_php/module-16/MigrationAssignment/database/migrations/2023_06_04_161450_create_orders_table.php] created.
```

## Task 8:

After creating the migration file for the "orders" table, run the migration to create the "orders" table in the database.

The screenshot shows the VS Code editor with the same migration file selected. The terminal at the bottom shows the command 'php artisan migrate' and its output, which indicates that the migration was successful. Below the command output, there is a table showing the status of all migrations.

```
hp@LAPTOP-A30EFPRI MINGW64 /d/Ostad_php/module-16/MigrationAssignment
$ php artisan migrate

[INFO] Nothing to migrate.

hp@LAPTOP-A30EFPRI MINGW64 /d/Ostad_php/module-16/MigrationAssignment
$ php artisan migrate:status

Migration name ..... Batch / Status
2019_12_14_000001_create_personal_access_tokens_table ..... [1] Ran
2023_06_04_152011_create_products_table ..... [1] Ran
2023_06_04_160103_add_category_to_products_table ..... [2] Ran
2023_06_04_161450_create_orders_table ..... [3] Ran
```