**Task 1:**

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

❖ Command line for creating MigrationAssignment:

✓ **composer global require laravel/installer**

✓ **laravel new MigrationAssignment**

**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.

❖ In order To Create new migration file I runned following command:

✓ **php artisan make:migration create_products_table**

**The product table migration file:**

```php
<?php

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

return new class extends Migration
```

```php
{
    /**
     * Run the migrations.
     */
    public function up(): void
    {
        Schema::create('products', function (Blueprint $table) {
            $table->id();
            $table->string('name');
            $table->decimal('price');
            $table->text('description', 1000);
            $table->timestamps();
        });
    }

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

**Task 3:**

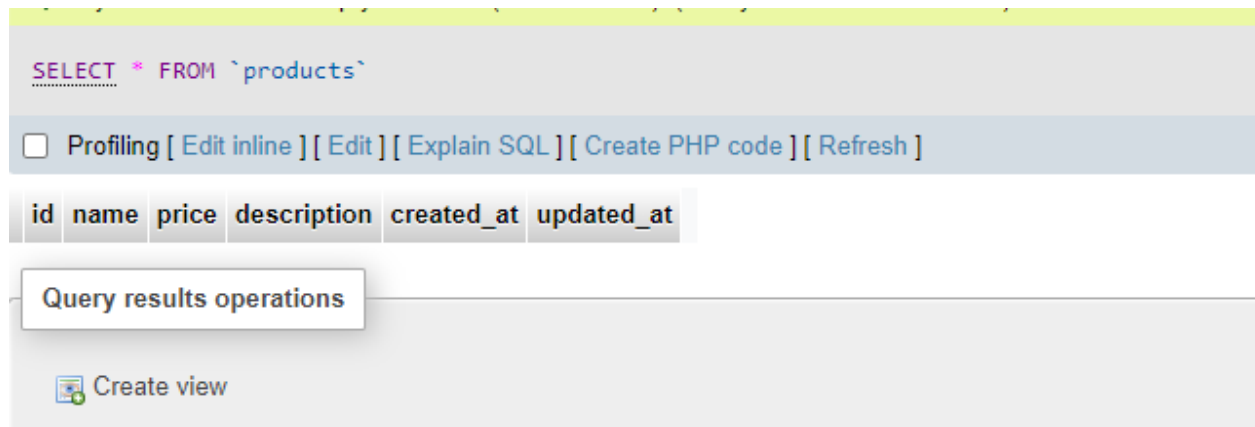After creating the migration file, run the migration to create the **"products"** table in the database.

- ❖ Migration Command:
- ✓ **php artisan migrate**

```
2014_10_12_000000_create_users_table ................................................. 573ms DONE
2014_10_12_100000_create_password_reset_tokens_table ................................. 1,103ms DONE
2019_08_19_000000_create_failed_jobs_table ........................................... 529ms DONE
2019_12_14_000001_create_personal_access_tokens_table ................................ 814ms DONE
2023_06_03_175852_create_products_table .............................................. 371ms DONE
```

```
SELECT * FROM `products`
```

☐ Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

| id | name | price | description | created_at | updated_at |

**Query results operations**
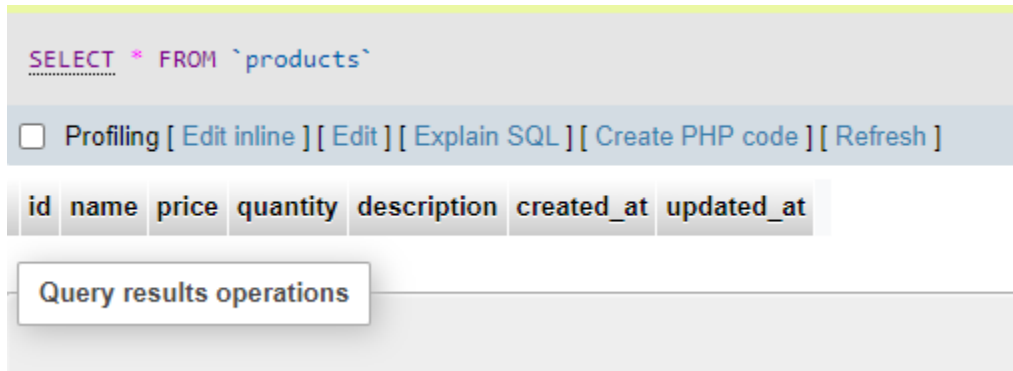
🗗 Create view

**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**.

For adding Quantity column I wrote this statements:

```php
public function up(): void
{
    Schema::table('products', function (Blueprint $table) {
        $table->after('price', function (Blueprint $table) {
            $table->integer('quantity')->nullable();
        });
    });
}
```

❖ Then again run migration command:
✓ **Php artisan migrate**

Now we can see that **quantity** column is added in the database.

**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.

- ❖ Migration Command For creating migration file named **"add_category_to_products_table"** :

- ✓ **php artisan make:migration add_category_to_products_table -- table=products**

  - ❖ In **add_category_to_products_table** migration file I wrote this:

    - ➢ I wanted to show category column after name column so write the statement like this

```php
public function up(): void
{
    Schema::table('products', function (Blueprint $table) {
        $table->after('name', function (Blueprint $table) {
            $table->string('category', 50);
        });
    });
}
```
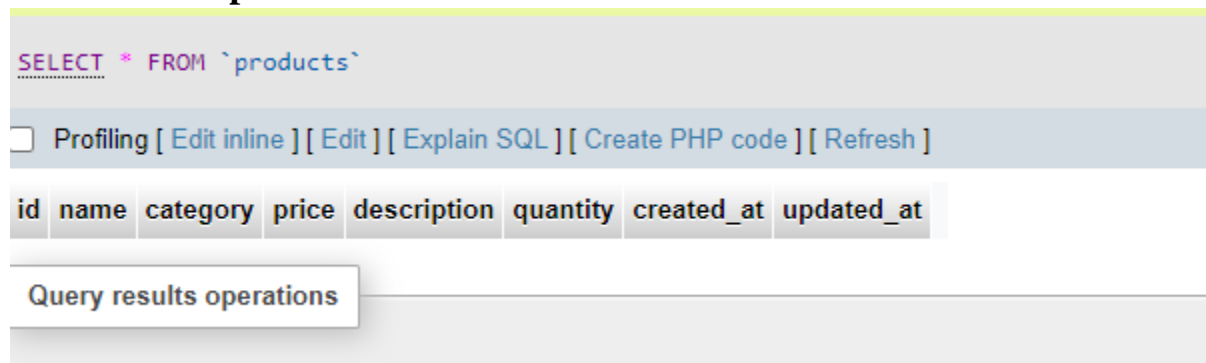
**Task 6:**
After creating the new migration file, run the migration to add the **"category"** column to the **"products"** table.

I wanted to show category column after name column, so category is showing after name column. So to achieve this I run below artisan command:

✓ **php artisan migrate**

**Database Output:**

```
SELECT * FROM `products`
```

☐ Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

| id | name | category | price | description | quantity | created_at | updated_at |

Query results operations

**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.

❖ In order To Create new migration file I runned following command:

**php artisan make:migration create_orders_table**

**Added necessary statements for adding mentioned columns:**

```php
public function up(): void
{
    Schema::create('orders', function (Blueprint $table) {
        $table->id();
        $table->unsignedBigInteger('product_id');
        $table->integer('quantity');
        //relationship
        $table->foreign('product_id')->references('id')->on('products')
            ->restrictOnDelete()
            ->cascadeOnUpdate();

        $table->timestamps();
    });
}
```

**Task 8:**
After creating the migration file for the "orders" table, run the migration to create the "orders" table in the database

❖ **Migration Command:**

✓ **php artisan migrate**

**Database Output:**

SELECT * FROM `orders`

☐ Profiling [ Edit inline ] [ Edit ] [ Explain SQL ] [ Create PHP code ] [ Refresh ]

| id | product_id | quantity | created_at | updated_at |
|----|-----------|----------|-----------|-----------|

Query results operations

🖳 Create view