Laravel / Commands

Brainster Web Development Academy

Exercise 01

Set up a new laravel project and create migrations for the following tables:

Roles:

- id
- name
- timestamps

users:

- id,
- name,
- surname,
- email (unique),
- password,
- role_id (foreign key to the roles table),
- timestamps

Exercise 01

Create a command which will enable you to create a new role from command line by calling the command.

Run the command 3 times (php artisan command:name) to create 3 roles.
Ask for the role name in the command line.

The roles that should be created are:

- admin,
- manager
- developer

Exercise 01

Create a command to populate the users table.

From the command line ask for the user’s:

- name - Use $this->anticipate(), auto-complete the choice with the first names already saved in the users table in the DB
- surname - Use $this->anticipate(), auto-complete the choice with the last names already saved in the users table in the DB
- email - Use $this->ask(). After the email is entered, check if it already exists in the DB. If it does show an error message with $this->error().
- password - Use $this->secret()
- role_id - Use $this->choice().

Set the array of choices from the roles table.

Ex. if you choose ‘admin’, change it with the proper id from the DB.

Exercise 01

If all data is valid valid, show the all the information (without the password) in a table in the command line.
Ask the user to confirm that the information is correct, then save the user in the DB and print an info() message in the command line.

Then, add authentication to the application using any of the Laravel starter packs.
Modify the register view and method with so that role can be selected upon registration.

When a user with the role of admin logs in the application a table of all the users should be shown.

If any other user logs in a welcome page with title ‘Welcome [name surname]’ is shown.