Laravel Cheat Sheet

Method 1: Docker

0. Preparation of the environment

1. Launch WSL (Windows users)
2. Launch Docker and login using your Docker credentials

1. Creating a new project

1. Make sure you are in the correct directory of your WSL where you want to create your project
2. Execute `curl https://laravel.build/project-name | bash` from the command line

2. Launching your project

1. Navigate to the directory using `cd project-name`
2. Execute `./vendor/bin/sail up`

3. Creating new stuff

3.1. Migrations (example: create table for products)

```
./vendor/bin/sail artisan make:migration create_products_table
```

3.2. Models (example: create associated Product model)

```
./vendor/bin/sail artisan make:model Product
```

3.3. Controllers (example: create ShopController)

```
./vendor/bin/sail artisan make:controller ShopController
```

3.4. Mailables (example: create confirmation mail)

```
./vendor/bin/sail artisan make:mail ConfirmationMail
```

4. Executing database migrations

`./vendor/bin/sail artisan migrate`

5. Connecting to the MySQL-database

1. Make sure `sail` is `up` (cfr. §2)
2. Launch SQLyog/DataGrip/...
3. Connect to `localhost` on port `3306` with username `sail` and password `password`

→ Not working? Check in `services.msc` whether a service called `MySQL` is running. If so, disable it, reboot your computer and try again.

Method 2: Homestead

Source: https://laravel.com/docs/9.x/homestead

0. Installation of Homestead (to be done once)

1. Install Vagrant through https://www.vagrantup.com/downloads.html
2. Install VirtualBox through https://www.virtualbox.org/wiki/Downloads
3. Open a Terminal
4. From your homedir, execute `git clone https://github.com/laravel/homestead.git .\Homestead`
5. Execute `cd Homestead`
6. Execute `git checkout release`
7. Execute `init.bat`

1. Configuration of Homestead (to be done once)

1. Generate your SSH keys using `ssh-keygen -t rsa -C "you@homestead"` from within Terminal
2. Update the `Homestead.yaml` file in the `Homestead` directory as follows:

```yaml
keys:
    - C:/Users/yourusername/.ssh/id_rsa
```

3. Create a new local directory in `C:\Users\yourusername` called `code`
4. Update the `Homestead.yaml` file as follows:

```yaml
folders:
    - map: C:/Users/yourusername/code
      to: /home/vagrant/code
```

2. Launching Homestead

1. Execute `vagrant up` from within your Homestead directory
2. Note: this can take quite some time in the beginning

3. Creating a new project called project-name

1. Add the site to the `Homestad.yaml` file:

```yaml
sites:
    - map: project-name.test
      to: /home/vagrant/code/project-name/public
```

2. Add the database as well, choose a name (here `database-name`):

```yaml
databases:
    - database-name
```

3. Reload the provisions by executing `vagrant reload --provision`
4. Update your hosts file (in `C:\Windows\system32\drivers\etc`) using VSCode by adding a line:

   `192.168.56.56 project-name.test`

   Save as admin when prompted.
5. Open a SSH connection to the Laravel box by executing `vagrant ssh`
6. In the box, execute the following Linux commands:
   - `cd code`
   - `composer create-project laravel/laravel project-name`

(To exit the SSH session at all times, type `exit`).

4. Updating the .env file for your database

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=database-name
DB_USERNAME=homestead
DB_PASSWORD=secret
```

5. Creating new stuff (from within SSH)

5.1. Migrations (example: create table for products)

`php artisan make:migration create_products_table`

5.2. Models (example: create associated Product model)

`php artisan make:model Product`

5.3. Controllers (example: create ShopController)

`php artisan make:controller ShopController`

5.4. Mailables (example: create confirmation mail)

`php artisan make:mail ConfirmationMail`

6. Executing database migrations

`php artisan migrate`

7. Connecting to the database from within your Windows OS

1. Make sure `vagrant` is `up` (cfr. §2)
2. Launch SQLyog/DataGrip/...
3. Connect to `localhost` on port `33060` with username `homestead` and password `secret`

8. Working on your code from within your Windows OS

The mapping specified in step 1 should allow you to access all your projects from within `C:\Users\yourusername\code`.

9. Halting the Vagrant box

From within your `Homestead` directory, execute `vagrant halt`.