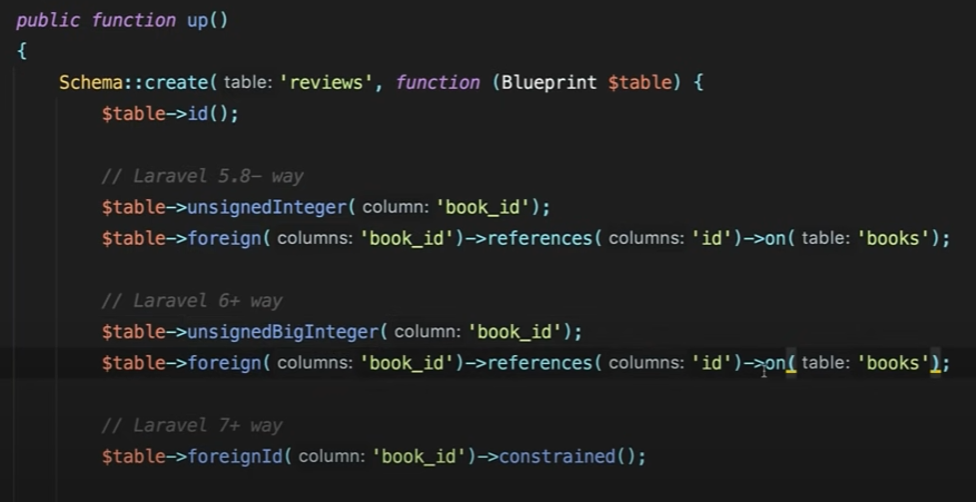
**App -> config -> database.php** contains the configuration for database connection. It also contains migrations table which tracks all the migrations that run for the application. It also contains a **Redis** database (it is an in-memory database).

**Migration:**

* In general migration means to migrate data from source database/s to target database/s by using a migration service. It is like the blueprint of the table.
* To migrate in Laravel we can write command **php artisan migrate.** This will create the database table according to the schema we defined. To see all the options for migrate command we can type -h/-help after the command. Some examples below-
  + **php artisan migrate:fresh** this command recreates (drops then create again) the existing table in case we edited some of them.
  + **php artisan make:migration CreateBlogTable** this example is for creating new migration.

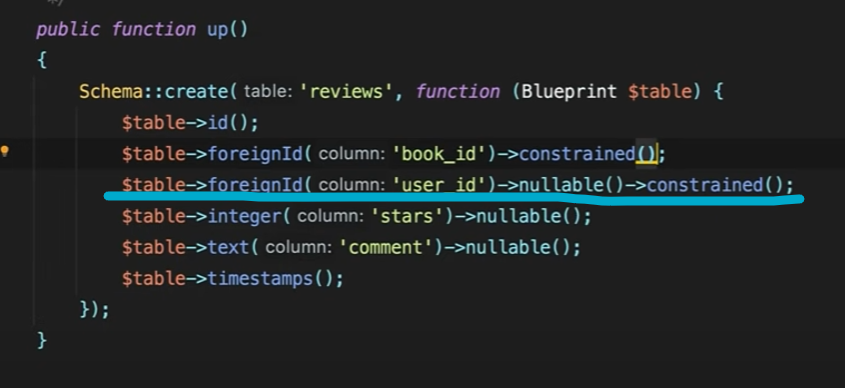
**Basic steps:** create database -> configure the database in the .env file -> run migrate command.

## Foreign Key:



1. **5.7 and below:** used to be just unsigned integer type, and the syntax is more specific
2. **Till 7.**0: it had a change to unsigned big integer type. Rest syntax is same
3. **7.0 and above:** shorter syntax.

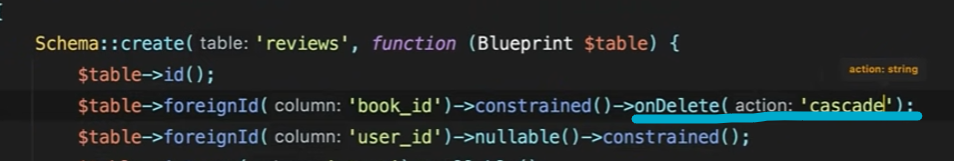
## Nullable () and Constrained ():



The order must be nullable first then constrained, otherwise nullable property doesn’t get migrated to database.

## Delete parent, restrict or cascade?

By default, it is restricted. In case we want to make it cascase, we need to add **onDelete()**

****

## Change order of migration:

It’s more of a hack, since the ordering is based on timestamp. So, we can just rename the file timestamp to an older timestamp according to our needs to re order the migration.

## Timestamps with default value:





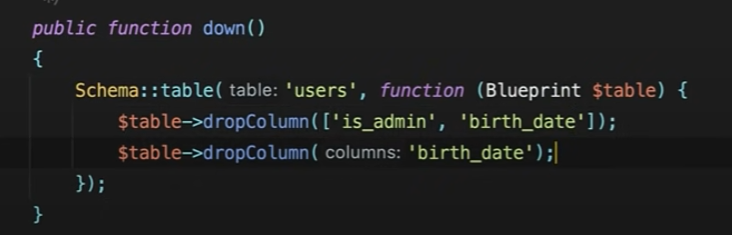
## Customize default migration file:

To customize, we can run command **php artisan stub:publish.** This will publish all the **stugs** in the **stug** folder. There we can find migration.create.stub and edit according to our need.

## Export Migration into one SQL file:

Command is **php artisan schema:dump.** It will dump the sql file in database-> schema folder.

## Drop multiple columns:



Here is two ways of doing it, we can specify one column at a time or use an array of columns.

## Rollback or refresh X steps:

**php artisan migrate:rollback –steps=3**

**php artisan migrate:refresh –steps=3**

## Auto-increment starting value:

