Generating Your Starter Files

Model, view, and controller files go in specific places within the Rails project folder structure.

We can use the `rails` command-line tool to generate all sorts of MVC starter files in the correct places.

In some cases `rails` can generate all the MVC source code required to perform CRUD action on a specific database table.

Table of Contents

1 Scaffolding a Rails MVC Bundle
2 Generating a Controller
3 Generating a Model
4 Removing Generated “Things”

Scaffolding a Rails MVC Bundle

We can have Rails generate a Model View Controller bundle for us by describing the data we which to store in our database. For example, to scaffold a products database table:

```
rails g scaffold product name:string description:text price:decimal stock_quantity:integer
```

If you’ve gone through the Getting Started Guide or the Rails Blogging Tutorial notes, you’ll know that this command creates all the routes, controllers, migrations, models and views needed to perform CRUD tasks on a specified resource by way of a web browser.

The scaffold command doesn’t create the required `products` table in the database. To create that table we need to run:

```
rails db:migrate
```

Generating a Controller

To generate an empty controller:

```
rails g controller controller_name
```

To specific the actions in the controller upfront (and have Rails generated the associated views):

```
rails g controller controller_name action1 action2 action3
```

Where action1, 2 and 3 are the names of actions you want to include in your controller.

Generating a Model

Generating a model is similar to generated a scaffold, except only the model and the migration in built:

```
rails g model product name:string description:text price:decimal stock_quantity:integer
```

Remember that when generating either a scaffold or a model you also have to prepare the database by running:

```
rails db:migrate
```

Removing Generated “Things”

But what if you want to remove a generated scaffold, model, or controller?

If you’ve been diligent with your git commits and your branching you can rollback the changes using version control.

You can also remove generated “things” using the `rails destroy` command:

```
rails d scaffold Product

rails d model Product
rails d controller products
```