

The Simple Guide to Rails Pluralization

This is probably one of the simplest things, but something that I have to stop and ask myself over and over again: when do I use singular or plural when using Rails generators? I've been a long time fan and user of Ruby on Rails. I'm also not a developer, so working with proper coding is actually foreign to me and I keep having to check on syntax and formatting.

Rails has easy to use generators to create the different elements of your application code. The trick I always find is that because Rails also does automatic pluralization, you have to make sure to use singular and plural in different contexts.

Here's my cheatsheet which is basically for my purposes since I keep having to use Evernote to find it. There is a decent chance that someone else may need this info too on occasion.

If you're stumbling on this and wondering "Why does Eric use `generate` instead of just using `g` because it seems unnecessarily long?", that's a great question. I use the full word for folks who (like me) are not developers.

Note that these will do the same thing:

```
rails generate Scaffold Bookmark url:string description:text
rails g Scaffold Bookmark url:string description:text
```

Here's my nifty "Eric is not a developer and keeps having to check this" checklist:

Generators that are Plural

Controller

Example:

```
rails generate controller Bookmarks index show
```

Helper

Example:

```
rails generate helper Bookmarks
```

Migration

Example:

```
rails generate migration AddPublicToBookmarks public:boolean
```

Resource

This is not a generator, but you have to use plural when coding resources:

Example:

```
resources :bookmarks, :only => [:index, :show]
```

Generators that are Singular

Model

Example:

```
rails generate model Bookmark url:string
```

Scaffold

Example:

```
rails generate scaffold Bookmark url:string description:text
```

Mailer

Example:

```
rails generate mailer UserMailer
```

Hopefully this is useful for others. I sure know I use it more than I would have thought.

Podcast Episode 107 - How Grassroots Software Development for Equity is Changing How Startups Build Successful Products at Series Code

This is a startup and product builder's dream episode as we speak with John Steele, founder of Series Code. John shares how his team at Series Code creates what I would almost term a Developers-as-a-Service with a powerful twist on how they operate: they split the cost to their customers between equity and cash. Beyond Series Code as a business model, John shares his lessons on product building, being a successful development team leader, how to motivate and build great teams and products.

Listen to the episode here:

<https://discopossepodcast.com/episode-107-how-grassroots-software-development-for-equity-is-changing-how-startups-build-successful-products-at-series-code/>

Listen and Subscribe on iTunes here:



Podcast Episode 107 -How Grassroots Software Development for Equity is Changing How Startups Build Successful Products at Series Code

PODCAST LINK:

<https://discopossepodcast.com/episode-107-how-grassroots-software-development-for-equity-is-changing-how-startups-build-successful-products-at-series-code/>