

An Introduction to Rails

Episode #1



This content is available free on
<http://RailsForZombies.org>

created by
envylabs



Prerequisites:
stay alive!

Episode #1

Deep in the CRUI



FOR ZOMBIES!!

What's happening?

140



greggpollack
1,520 tweets

148 3,028 357
following followers listed

Who to follow view all



mojombo
Tom Preston-Werner
Follow



bryanl
Bryan Liles (ツ)
Follow

Home

@greggpollack

Direct Messages 113

Favorites

Retweets

Search

Lists

greggs-favorites

New list View all

Trending: Worldwide

Change

Taylor Hotner

Teen Choice Awards

Ted Stevens

#wheniwaslittle

#followerquestion

Ramadhan

#annoyingquestion

Create our Zombie Database

CRUD

```
$ rails new twitter_for_zombies
```

```
~$ rails new twitter_for_zombies
create
create  README
...
create  vendor/plugins
create  vendor/plugins/.gitkeep
```

Create our Zombie Database

CRUD

```
$ bundle
```

```
~/twitter_for_zombies$ bundle
Fetching source index for http://rubygems.org/
Installing rake (0.9.1)
Using abstract (1.0.0)
Using activesupport (3.0.7)
Using builder (2.1.2)
...
Using thor (0.14.6)
Using railties (3.0.7)
Using rails (3.0.7)
Installing sqlite3 (1.3.3) with native extensions
Your bundle is complete! Use 'bundle show [gemname]' to see where a bundled gem is installed.
```

```
$ rails new twitter_for_zombies
```

Create our Zombie Database

CRUD

```
$ rails generate scaffold Tweet status:string name:string
```

```
~/twitter_for_zombies$ rails generate scaffold Tweet status:string name:string
invoke  active_record
create  db/migrate/20110605030231_create_tweets.rb
create  app/models/tweet.rb
invoke  test_unit
create  test/unit/tweet_test.rb
create  test/fixtures/tweets.yml

...

invoke  test_unit
create  test/unit/helpers/tweets_helper_test.rb
invoke  stylesheets
create  public/stylesheets/scaffold.css
```

```
$ rails new twitter_for_zombies
```

```
$ bundle
```

Create our Zombie Database

CRUD

```
$ rake db:migrate
```

```
~/twitter_for_zombies$ rake db:migrate
(in /Users/wpdavenport/twitter_for_zombies)
== CreateTweets: migrating =====
-- create_table(:tweets)
   => 0.0009s
== CreateTweets: migrated (0.0010s) =====

~/twitter_for_zombies$ █
```

```
$ rails new twitter_for_zombies
```

```
$ bundle
```

```
$ rails generate scaffold Tweet status:string name:string
```


CRUD

Columns

(we have 3)

tweets

Rows

(we have 4)

	A	B
1	Where can I get a good bite to eat?	Ash
2	My left arm is missing, but I don't care.	Bob
3	I just ate some delicious brains.	Jim
4	OMG, my fingers turned green. #FML	Ash

id

status

name

	A	B
1	Where can I get a good bite to eat?	Ash
2	My left arm is missing, but I don't care.	Bob
3	I just ate some delicious brains.	Jim
4	OMG, my fingers turned green. #FML	Ash

Console

\$ rails s

(browser) <http://localhost:3000/tweets>

ADD SOME ZOMBIE TWEETS TO YOUR APP!

	A	B
1	Where can I get a good bite to eat?	Ash
2	My left arm is missing, but I don't care.	Bob
3	I just ate some delicious brains.	Jim
4	OMG, my fingers turned green. #FML	Ash

Console

CRUD

\$ rails c



Hash

Series of key value pairs

CRUD

```
b = { :status => "I just ate some delicious brains",  
      :name => "Jim" }
```

```
puts b[:status]
```

"I just ate some delicious brains"

```
puts b[:name]
```

"Jim"

```
puts b[:name] + " said " + b[:status]
```

"Jim said I just ate some delicious brains"



tweets

id	status	name
1	Where can I get a good bite to eat?	Ash
2	My left arm is missing, but I don't care.	Bob
3	I just ate some delicious brains.	Jim
4	OMG, my fingers turned green. #FML	Ash

ZOMBIE SURVIVAL LESSON #1

Learn to retrieve a hash of the Zombie tweet with id = 3

```
ANSWER t = Tweet.find(3)

puts t[:id]      3

puts t[:status]  "I just ate some delicious brains."

puts t[:name]    "Jim"
```

CRUD

```
t = Tweet.find(3)
```

```
puts t[:id]
```

```
puts t.id
```

```
puts t[:status]
```

```
puts t.status
```

```
puts t[:name]
```

```
puts t.name
```

Same As

ANSWER

```
puts t.id      3
```

```
puts t.status  "I just ate some delicious brains."
```

```
puts t.name    "Jim"
```

C

R

U

D

Create

Read

Update

Delete

Create

```
t = Tweet.new  
t.status = "I <3 brains."  
t.save
```

Read

```
Tweet.find(3)
```

Update

```
t = Tweet.find(3)  
t.name = "EyeballChomper"  
t.save
```

Delete

```
t = Tweet.find(3)  
t.destroy
```

```
t = Tweet.new  
t.status = "I <3 brains."  
t.zombie = "Jim"  
t.save
```

— Notice we don't set the id.
The id gets set manually for us.

Alternative Syntax

```
t = Tweet.new(:status => "I <3 brains", :zombie => "Jim")  
t.save
```

```
Tweet.create(:status => "I <3 brains", :zombie => "Jim")
```

Read

Syntax

`Tweet.find(2)` # Returns a single item

`Tweet.find(3, 4, 5)` # Returns an array

`Tweet.first` # Returns the first tweet

`Tweet.last` # Returns the last tweet

`Tweet.all` # Returns all the tweets

`Tweet.count` # Returns number of tweets

`Tweet.order(:name)` # All ordered by zombie

`Tweet.limit(10)` # Only 10 tweets

`Tweet.where(:name => "ash")` # Only tweets by Ash

`Tweet.where(:name => "ash").order(:name).limit(10)`

method chaining

```
t = Tweet.find(3)
t.name = "EyeballChomper"
t.save
```

```
t = Tweet.find(2)
t.attributes = {
  :status => "Can I munch your eyeballs?",
  :name => "EyeballChomper"
}
t.save
```

*Saves without
calling save!*

```
t = Tweet.find(2)
t.update_attributes(
  :status => "Can I munch your eyeballs?",
  :name => "EyeballChomper"
)
```

```
t = Tweet.find(2)  
t.destroy
```

```
Tweet.find(2).destroy
```

```
Tweet.destroy_all
```



ZOMBIE LAB 1

<http://railsforzombies.org>

Stop when you reach the video for Lab 2

Don't worry about watching any of the videos

Download the slides so you can remember the syntax