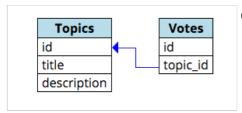
有 0 位同学正在浏览当前页面

# 3-8 把投票记录和 Topics 接起来

(/favorites?post\_id=47)

# 目标



(https://s3-ap-northeast-1.amazonaws.com/ontrackapp-

 $production/XPaKQENjQ1OBk75V8e4x\_\%E8\%9E\%A2\%E5\%B9\%95\%E5\%BF\%AB\%E7\%85\%A7\%202016-07-10\%20\%E4\%B8\%8B\%E5\%8D\%8810.20.26.png)$ 

资料库里面有了关联号码,接下来我们要来告诉 Rails, topic 和投票记录之间有明确的关系。

## 步骤

### 步骤 1: 告诉 Topic model 有 Vote (投票记录) 的存在

编辑 app/models/topic.rb 让它长得像这样:

app/models/topic.rb

class Topic < ApplicationRecord
 has\_many :votes, dependent: :destroy
end</pre>

```
topic.rb — ~/railsbridge/suggestotron

topic.rb

topic.rb

topic.rb

concerns
application_record
topic.rb

app/models/topic.rb

topic.rb

topic.rb

topic.rb

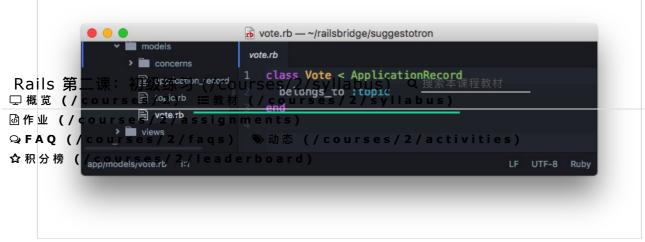
LF UTF-8 Ruby
```

(http://ww1.sinaimg.cn/large/006y8lVajw1faqvu2tazdj30lk08idgy.jpg)

#### 步骤 2: 告诉 Vote model 有 Topic 的存在

编辑 app/models/vote.rb 让它长得像这样:

```
app/models/vote.rb
class Vote < ApplicationRecord
  belongs_to :topic
end</pre>
```



(http://ww4.sinaimg.cn/large/006y8lVajw1faqvv8wpbzj30lk08iwfh.jpg)

## 步骤 3: 进 Rails console 随便玩玩看 Topic 和 Vote model

首先, 先确定你已经开了至少一篇 topic。

然后,从终端机视窗进入 Rails console:

rails c

预期的结果:

Running via Spring preloader in process 11071 Loading development environment (Rails 5.0.0) 2.3.1 :001 >

在 console 里面试试看这些指令:

检查总共有多少 topics:

继续在终端输入 Topic.count

把第一篇 topic 存入变数

my\_topic = Topic.first

my\_topic 可以是任何的变数名称,不过现在我们一律用 my\_topic。

把那个 topic 的 title 改成别的:

my\_topic.update\_attributes(title: 'Edited in the console')

新增一笔投票记录给那个 topic:

my\_topic.votes.create

检查 topic 总共有多少笔投票:

my\_topic.votes.count

删除一笔那个 topic 的投票记录:

my\_topic.votes.first.destroy

最后,记得输入 exit 退出console。

请注意,你在 **Model class**(例如 **Topic、Vote**)和 **Model instance**(例如此例的my\_topic)可以做的事是不一样的。my\_topic.votes称作 association,在这里的行为比较像是 model class。

#### Model class / association 的 methods

Topic.first

Topic.last

Topic.all

Topic.count

Topic.find\_by\_id(5)

Topic.destroy\_all

my\_topic.votes.count

my\_topic.votes.create

my\_topic.votes.destroy\_all

#### Model instance 的 methods

my\_topic.title

my\_topic.title = 'New title'

my\_topic.update\_attributes(title: 'New title')

my\_topic.save

my\_topic.save!

my\_topic.destroy

my\_topic.votes.first.destroy

(https://s3-ap-northeast-1.amazonaws.com/ontrackapp-

production/GmS6BOkCRJagJJkU7fx0\_%E8%9E%A2%E5%B9%95%E5%BF%AB%E7%85%A7%202016-07-10%20%E4%B8%8B%E5%8D%8810.27.27.png)

要看完整的列表的话, 你可以上 Active Record Query Interface RailsGuide (http://guides.rubyonrails.org/active\_record\_querying.html)

## 对本页内容的感受如何?





So easy



6 崩溃了

**←** 上一页 (/posts/46)

♀可以使用 ← → 键进行翻页

下一页→ (/posts/48)

全栈营 课程资源 关于我们

课程介绍 学习中心 公司介绍 (/pages/course\_intro)(/dashboard) (/pages/about)

教学团队帮助文档常见问题(/pages/teachers) (http://docs.qzy.camp(//)pages/faq)学员心得交流论坛联系方式

(/pages/students) (http://forum.qzy.ca(n/ppp/ges/contact)

♀ 在线客服 (非技术答疑, 工作日10:00-19:00)



新生大学 - 软件学院是李笑来对未来世界的实验计划,旨在改变中国的计算机教育。

(http://www.xinshengdaxue.com/)

© 2017 情非得已(北京)科技有限公司

本站聘请孙雁飞律师为法律顾问