

Язык программирования Ruby, занятие 6

CS 212

3 декабря 2018 г.

План

- 1 Миграции
- 2 Запуск миграций
- 3 Связи между таблицами
- 4 Полиморфные связи
- 5 Callbacks

Code First

```
1 class CreateProducts < ActiveRecord::Migration[5.0]
2   def change
3     create_table :products do |t|
4       t.string :name
5       t.text :description
6
7       t.timestamps
8     end
9   end
10 end
```

Листинг 1: Миграции

<http://rusrails.ru/active-record-basics>

Code First

```
1 class ChangeProductsPrice <
  ActiveRecord::Migration[5.0]
2 def up
3   change_table :products do |t|
4     t.change :price, :string
5   end
6 end
7
8 def down
9   change_table :products do |t|
10    t.change :price, :integer
11  end
12 end
13 end
```

Code First

```
1 class CreatePublications <
  ActiveRecord::Migration[5.0]
2 def change
3   create_table :publications do |t|
4     t.string :title
5     t.text :description
6     t.references :publication_type
7     t.integer :publisher_id
8     t.string :publisher_type
9     t.boolean :single_issue
10    t.timestamps
11  end
12  add_index :publications, :publication_type_id
13 end
14 end
```

ALTER TABLE

```
1 bin/rails generate migration
   AddPartNumberToProducts part_number:string
2
3 class AddPartNumberToProducts <
   ActiveRecord::Migration[5.0]
4 def change
   add_column :products, :part_number, :string
6 end
7 end
```

Листинг 4: Добавление столбцов

ALTER TABLE

```
1 bin/rails generate migration AddUserRefToProducts
   user:references
2
3 class AddUserRefToProducts <
   ActiveRecord::Migration[5.0]
4 def change
   add_reference :products, :user, foreign_key: true
6 end
7 end
```

Листинг 5: Добавление ограничений

Многие ко многим

```
1 generate migration CreateJoinTableCustomerProduct
   customer product
2
3 class CreateJoinTableCustomerProduct <
   ActiveRecord::Migration[5.0]
4 def change
5   create_join_table :customers, :products do |t|
6     # t.index [:customer_id, :product_id]
7     # t.index [:product_id, :customer_id]
8   end
9 end
10 end
```

Листинг 6: Добавление ограничений

Запуск миграций

```
1 bin/rails db:migrate
2 bundle exec rake db:migrate
3 bundle exec rake db:rollback
4 db:rollback STEP=3
5 db:migrate:redo STEP=3
6 db:migrate VERSION=20080906120000
7 db:migrate RAILS_ENV=test
```

Листинг 7: Запуск миграций

Подготовка БД

```
1 bundle exec rake db:schema:load
2
3 bundle exec rake db:create
4 bundle exec rake db:migrate
```

Листинг 8: Подготовка БД

Связи между таблицами

```
1 class Author < ApplicationRecord
2 end
3
4 class Book < ApplicationRecord
5 end
6
7 @book = Book.create(published_at: Time.now,
8   author_id: @author.id)
9
10 @books = Book.where(author_id: @author.id)
11 @books.each do |book|
12   book.destroy
13 end
14 @author.destroy
```

Связи между таблицами

```
1 class Author < ApplicationRecord
2   has_many :books, dependent: :destroy
3 end
4
5 class Book < ApplicationRecord
6   belongs_to :author
7 end
8 @book = @author.books.create(published_at:
9   Time.now)
10 @author.destroy
```

Листинг 10: Операции со связями

Связи между таблицами

```
1
2 belongs_to
3 has_one
4 has_many
5 has_many :through
6 has_one :through
7 has_and_belongs_to_many
```

Листинг 11: Типы связей

Belongs To

```
1 class Book < ApplicationRecord
2   belongs_to :author
3 end
4 class CreateBooks < ActiveRecord::Migration[5.0]
5 def change
6   create_table :authors do |t|
7     t.string :name
8   end
9
10  create_table :books do |t|
11    t.belongs_to :author, index: true
12    t.datetime :published_at
13  end
14 end
15 end
```

Has One

```
1 class CreateSuppliers <
  ActiveRecord::Migration[5.0]
2 def change
3   create_table :suppliers do |t|
4     t.string :name
5   end
6
7   create_table :accounts do |t|
8     t.belongs_to :supplier, index: true,
9       foreign_key: true
10    t.string :account_number
11  end
12 end
```

has many

```
1 class CreateAuthors < ActiveRecord::Migration[5.0]
2   def change
3     create_table :authors do |t|
4       t.string :name
5       t.timestamps
6     end
7
8     create_table :books do |t|
9       t.belongs_to :author, index: true
10      t.datetime :published_at
11      t.timestamps
12    end
13  end
14 end
```


has many through

```
1 class Physician < ApplicationRecord
2   has_many :appointments
3   has_many :patients, through: :appointments
4 end
5
6 class Appointment < ApplicationRecord
7   belongs_to :physician
8   belongs_to :patient
9 end
10
11 class Patient < ApplicationRecord
12   has_many :appointments
13   has_many :physicians, through: :appointments
14 end
```

has one through

```
1 class Supplier < ApplicationRecord
2   has_one :account
3   has_one :account_history, through: :account
4 end
5
6 class Account < ApplicationRecord
7   belongs_to :supplier
8   has_one :account_history
9 end
10
11 class AccountHistory < ApplicationRecord
12   belongs_to :account
13 end
```

Листинг 16: has one through

has one through

```
1 class CreateAccountHistories <
  ActiveRecord::Migration[5.0]
2 def change
3   create_table :suppliers do |t|
4     t.string :name
5   end
6   create_table :accounts do |t|
7     t.belongs_to :supplier, index: true
8     t.string :account_number
9   end
10  create_table :account_histories do |t|
11    t.belongs_to :account, index: true
12    t.integer :credit_rating
13  end
14 end
```

has and belongs to many

```
1 class Assembly < ApplicationRecord
2   has_and_belongs_to_many :parts
3 end
4
5 class Part < ApplicationRecord
6   has_and_belongs_to_many :assemblies
7 end
```

Листинг 18: has and belongs to many

has and belongs to many

```
1 class CreateAssembliesAndParts <
  ActiveRecord::Migration[5.0]
2 def change
3   create_table :assemblies do |t|
4     t.string :name
5     t.timestamps
6   end
7
8   create_table :parts do |t|
9     t.string :part_number
10    t.timestamps
11  end
12
13  create_table :assemblies_parts, id: false do |t|
14    t.belongs_to :assembly, index: true
```

Полиморфные связи

```
1 class Picture < ApplicationRecord
2   belongs_to :imageable, polymorphic: true
3 end
4
5 class Employee < ApplicationRecord
6   has_many :pictures, as: :imageable
7 end
8
9 class Product < ApplicationRecord
10  has_many :pictures, as: :imageable
11 end
```

Листинг 20: Полиморфные связи

Полиморфные связи

```
1
2 class CreatePictures < ActiveRecord::Migration[5.0]
3   def change
4     create_table :pictures do |t|
5       t.string :name
6       t.references :imageable, polymorphic: true,
7         index: true
8       t.timestamps
9     end
10  end
```

Листинг 21: Полиморфные связи 2

Join в одной таблице

```
1 class Employee < ApplicationRecord
2   has_many :subordinates, class_name: "Employee",
3     foreign_key: "manager_id"
4
5   belongs_to :manager, class_name: "Employee"
6 end
```

Листинг 22: Join в одной таблице

Join в одной таблице

```
1 class CreateEmployees <
  ActiveRecord::Migration[5.0]
2 def change
3   create_table :employees do |t|
4     t.references :manager, index: true
5     t.timestamps
6   end
7 end
8 end
```

Листинг 23: Join в одной таблице 2

Callbacks

```
1 before_validation
2 after_validation
3 before_save
4 around_save
5 before_create
6 around_create
7 after_create
8 after_save
9 after_commit/after_rollback
```

Листинг 24: Создание объекта

Callbacks

```
1 before_validation
2 after_validation
3 before_save
4 around_save
5 before_update
6 around_update
7 after_update
8 after_save
9 after_commit/after_rollback
```

Листинг 25: Обновление объекта

Callbacks

```
1 before_destroy
2 around_destroy
3 after_destroy
4 after_commit/after_rollback
```

Листинг 26: Удаление объекта

CRUD

```
1 users = User.all
2 User.first
3 User.find(id)
4 User.find_by_name
5 User.find_by_id
6 users = User.where(name: 'David', occupation:
    'Code Artist').order(created_at: :desc)
```

Листинг 27: Чтение записей

CRUD

```
1 users = User.where("name = #{name}") #danger!  
2 users = User.where(age: #danger!
```

Листинг 28: Чтение записей 2

CRUD

```
1 user.delete # no callbacks
2 user.destroy # with callbacks
```

Листинг 29: Удаление записей