



SETUP & CLI

```
gem install rails
rails new myapp --api
--database=postgresql
rails server (or rails s)
rails generate scaffold User name:string
email:string
rails db:migrate
rails console (REPL)
```

MVC STRUCTURE

app/models/	ActiveRecord models
app/controllers/	ApplicationController subclasses
app/views/	ERB templates
app/helpers/	view helper methods
config/routes.rb	routing config
db/migrate/	migration files

ROUTES

```
resources :users # CRUD routes
resources :posts, only: [:index, :show, :create]
namespace :api do
  resources :users
end
get '/login', to: 'sessions#new', as: :login
root 'home#index'
rails routes # list all routes
```

ACTIVE RECORD MODELS

```
class User < ApplicationRecord
  validates :email, presence: true,
    uniqueness: true
  validates :name, length: { minimum: 2 }
  has_many :orders, dependent: :destroy
  belongs_to :team, optional: true
  before_save :downcase_email
end
```

MIGRATIONS

```
rails generate migration AddPhoneToUsers
phone:string
class AddPhoneToUsers <
  ActiveRecord::Migration[7.0]
  def change
    add_column :users, :phone, :string
    add_index :users, :email, unique: true
  end
end
rails db:migrate / db:rollback /
db:schema:load
```

VIEWS (ERB)

```
<%= @user.name %> <## comment %>
<% @users.each do |u| %><%= u.name %><%
end %>
<%= render 'shared/header' %>
<%= link_to 'Edit',
  edit_user_path(@user) %>
<%= form_with model: @user do |f| %>
  <%= f.text_field :name, class:
    'form-control' %>
  <% end %>
```

ACTIVE RECORD ASSOCIATIONS

```
has_one has_many belongs_to
has_and_belongs_to_many
has_many :comments, through: :posts #
join model
# Eager loading:
User.includes(:orders, :role).all
User.eager_load(:orders).where(orders:
{status: 'paid'})
User.preload(:orders)
```

VALIDATIONS

```
validates :name, presence: true
validates :email, format: { with:
  URI::MailTo::EMAIL_REGEXP }
validates :age, numericality: {
  greater_than: 0 }
validates :role, inclusion: { in:
  %w[admin user] }
validate :custom_validation
def custom_validation
  errors.add(:base, 'Custom error') if
  condition
end
```

QUERY INTERFACE

```
User.all / User.find(1) /
User.find_by(email: 'a@b.com')
User.where('active = ?',
  true).order(:name).limit(10)
User.where(active: true, role: 'admin')
User.joins(:orders).where(orders:
{status: 'paid'})
User.select(:id, :name).distinct
User.count / User.sum(:balance) /
User.average(:age)
```

RAILS GENERATORS

```
rails g controller Api::Users index show
create
rails g model Post title:string
body:text user:references
rails g migration
CreateJoinTableUsersRoles user role
rails g job SendEmail
rails g mailer UserMailer welcome_email
bundle exec rails test (or rspec with
rspec-rails gem)
```

COMMON GOTCHAS

N+1: use includes/eager_load never loop + lazy load
Strong parameters: use permit() to whitelist params
SECRET_KEY_BASE in credentials use rails credentials:edit
Time zones: always use Time.current, not Time.now
Convention over configuration
follow Rails naming conventions