# RAILS (2/3)

- Rails CLI
- Console
- Server
- Migration
- Logs
- Request
- Response

# RAILS CLI

“There are a few commands that are absolutely critical to your everyday usage of Rails.”
https://guides.rubyonrails.org/command_line.html

```
bin/rails console
bin/rails server
bin/rails test
bin/rails generate
bin/rails db:migrate
bin/rails db:create
bin/rails routes
bin/rails dbconsole
rails new app_name
```

# Rails console

“Rails console is like IRB but for Rails, it support code completion.”

```
bin/rails console
> # test a rails request
> app.post '/messages', params: {"text" => "Hello", "user_id" => 1}
> # fetch db
> Message.all
> # create/update/delete db
> Message.last.update(text: 'Hello, World')
> # reload the console if you changes your code
> reload!
```

# RAILS SERVER

“Rails include a built-in server (for development) ” https://guides.rubyonrails.org/command_line.html#bin-rails-server

```
$ bin/rails server
=> Booting Puma
=> Rails 6.0.0 application starting in development
=> Run `bin/rails server --help` for more startup options
Puma starting in single mode...
* Version 3.12.1 (ruby 2.5.7-p206), codename: Llamas in Pajamas
* Min threads: 5, max threads: 5
* Environment: development
* Listening on tcp://localhost:3000
Use Ctrl-C to stop
```

# MIGRATION

“Migrations are a feature of Active Record that allows you to evolve your database schema over time. Rather than write schema modifications in pure SQL, migrations allow you to use a Ruby DSL to describe changes to your tables.” https://guides.rubyonrails.org/activerecordmigrations.html

- Generate a migration

  ```
  bin/rails generate migration CreateProducts name:string description:text
  ```

- migration are ruby files

```ruby
# db/migrate/20131024214113_create_products
class CreateProducts < ActiveRecord::Migration[6.0]
  def change
    create_table :products do |t|
      t.string :name
      t.text :description

      t.timestamps
    end
  end
end
```

# LOGS

“Rails write logs in a local file”
https://guides.rubyonrails.org/debuggingrailsapplications.html#the-logger

```
Started POST "/articles" for 127.0.0.1 at 2018-10-18 20:09:23 -0400
Processing by ArticlesController#create as HTML
  Parameters: {"utf8"=>"✓", "authenticity_token"=>"XLveDrKzF1SwaiNRPTaMtkrsTzedtebPPkmxEFIU0ordLjICSnXsSNfrdMa4ccyBjuGwnnEiQhEoMN6H1Gtz3A==", "article"=>{"title"=>"Debugging Rails", "body"=>"I'm learning how to print in logs.", "published"=>"0"}, "commit"=>"Create Article"}
New article: {"id"=>nil, "title"=>"Debugging Rails", "body"=>"I'm learning how to print in logs.", "published"=>false, "created_at"=>nil, "updated_at"=>nil}
Article should be valid: true
   (0.0ms)  begin transaction
  ↳ app/controllers/articles_controller.rb:31
  Article Create (0.5ms)  INSERT INTO "articles" ("title", "body", "published", "created_at", "updated_at") VALUES (?, ?, ?, ?, ?)  [["title", "Debugging Rails"], ["body", "I'm learning how to print in logs."], ["published", 0], ["created_at", "2018-10-19 00:09:23.216549"], ["updated_at", "2018-10-19 00:09:23.216549"]]
  ↳ app/controllers/articles_controller.rb:31
   (2.3ms)  commit transaction
  ↳ app/controllers/articles_controller.rb:31
The article was saved and now the user is going to be redirected...
Redirected to http://localhost:3000/articles/1
Completed 302 Found in 4ms (ActiveRecord: 0.8ms)
```

# CUSTOM LOGGING

```
Rails.logger.info "this is a information log"
Rails.logger.debug "this is a debuglog"
Rails.logger.error "this is an error log"
```

# REQUEST

“The request object contains a lot of useful information about the request coming in from the client” https://guides.rubyonrails.org/actioncontrolleroverview.html#the-request-object

```
# ip address of the client
request.remote_ip
# HTTP headers sent by the client
request.headers
# content type requested by the client
request.format
```

# RESPONSE

“The response object is not usually used directly, but is built up during the execution of the action and rendering of the data that is being sent back to the user, but sometimes - like in an after filter - it can be useful to access the response directly”
https://guides.rubyonrails.org/
actioncontrolleroverview.html#the-request-object

```
response.headers["Content-Type"] = "application/pdf"
```