The rails new command created a new Rails app named MySite. It generated a number of files and folders that we will use to build the app

1. The browser makes a request for the URL [http://localhost:8000](http://localhost:8000/).
2. The request hits the Rails router in config/routes.rb. The router recognizes the URL and sends the request to the controller.
3. The controller receives the request and processes it.
4. The controller passes the request to the view.
5. The view renders the page as HTML.
6. The controller sends the HTML back to the browser for you to see.

This is called the request/response cycle. It's a useful way to see how a Rails app's files and folders fit together.

we need three parts to build a Rails app: a controller, a route, and a view

The rails generate controller Pages command generated a new controller named Pages. This created a file named app/controllers/pages\_controller.rb.. Inside the new Pages controller, we added a method called home. Methods in Rails controllers are also referred to as controller actions, so here we added the home action to the Pages controller.

1. When you type http://localhost:8000/welcome, the browser makes a request for the URL /welcome.
2. The request hits the Rails router.
3. The router maps the URL to a controller action to handle the request.
4. The controller action recieves the request, and asks the model to fetch data from the database.
5. The model returns data to the controller action.
6. The controller action passes the data on to the view.
7. The view renders the page as HTML.
8. The controller sends the HTML back to the browser.

For dynamic app we need four parts to build a Rails app - a model, a route, a controller, and a view.

The rails generate model command created a new model named Message. In doing so, Rails created two files:

1. a model file in app/models/message.rb. The model represents a table in the database.
2. a migration file in db/migrate/. Migrations are a way to update the database.

The migration file contains a few things:

1. The change method tells Rails what change to make to the database. Here it uses the create\_table method to create a new table in the database for storing messages.
2. Inside create\_table, we added t.text :content. This will create a text column called content in the messages tables.
3. The final line t.timestamps is a Rails command that creates two more columns in the messages table called created\_at and updated\_at. These columns are automatically set when a message is created and updated.

The rake db:migrate command updates the database with the new messages data model.

Finally the rake db:seed command seeds the database with sample data from db/seeds.rb.

rails generate controller Messages // Create controller named messages

get 'messages' => 'messages#index' //routes

Rails provides seven standard controller actions for doing common operations with data. Here we want display a list of all messages, so we used the index action.

Putting it all together:

1. When a user visits http://localhost:8000/messages, the routes file maps this request to the Messages controller's index action.
2. The index action retrieves all messages from the database and stores them in variable @messages.

The @messages variable is passed on to the view. The view should display each message

The file index.html.erb is a web template. Web templates are HTML files that contain variables and control flow statements. Rather than write the same HTML over and over again for each message, we can use web templates to loop through and display data from the database.

In this case:

1. <% @messages.each do |message| %> iterates through each message in @messages array. We created @messages in the Messages controller's index action.
2. For each message, we use <%= message.content %> and <%= message.created\_at %> to display its content and the time when it was created.

The default web templating language in Rails is embedded Ruby, or **ERB.**

Messages controller's seven actions (index, show, new, create, edit, update, and destroy):