

- ❖ Controllers are the best place to initiate database (model) operations
- ❖ Models represent real-world things, and they also act as the sole gateway to the application database.
- ❖ HTTP provides different *methods* or *verbs* that combine with each *URL*
- ❖ Think in terms of *resources*, not web "pages"
- ❖ Rails provides view helpers that synthesize common HTML for you.
- ❖ Rails view helpers use HTML5 and provide best-practices code.
- ❖ HTTP CRUD is achieved via: **GET**, **POST**, **PATCH**, and **DELETE**
- ❖ HTML only supports **GET** (`<a>` and `<form>`) and **POST** (`<form>`)
- ❖ Rails view helpers also provide "extra" functionality that simulates full HTTP support over HTML.
- ❖ Each form you display will require two actions: one to display the form, plus one to receive the submitted data.
- ❖ Use **rake routes** to see a cheat sheet of your defined routes
- ❖ The best way to understand your routes is with **rake routes**, not `routes.rb`
- ❖ Use `form_tag` to generate the necessary security required for non-**GET** forms
- ❖ Use `link_to` instead of `<a>` tags, so you can use `:method => :delete` when appropriate
- ❖ Warning! Do not use `.create` or `.update` with the `params` hash until we learn about strong parameters
- ❖ One fully-defined RESTful resource will require 7 routes, 7 actions, and 4 views.



index
show
new
create
edit
update
delete