ember
ABOUT
GUIDES
API
COMMUNITY
BLOG
BUILDS
Search the guide
FORK US!
v2.0.0
GETTING STARTED
THE OBJECT MODEL
ROUTING
Introduction
Defining Your Routes
Specifying a Route's Model
Rendering a Template
Redirecting
Preventing and Retrying Transitions
Loading / Error Substates
Query Parameters
Asynchronous Routing
TEMPLATES
COMPONENTS
CONTROLLERS
MODELS
APPLICATION CONCERNS
VIEW v2.12.0

Old Guides - You are viewing the guides for Ember v2.0.0.

INTRODUCTION

Imagine we are writing a web app for managing a blog. At any given time, we should be able to answer questions like *What post are they looking at?* and *Are they editing it?* In Ember.js, the answer to these questions is determined by the URL.

The URL can be set in a few ways:

- The user loads the app for the first time.
- The user changes the URL manually, such as by clicking the back button or by editing the address bar.
- The user clicks a link within the app.
- Some other event in the app causes the URL to change.

Regardless of how the URL becomes set, the Ember router then maps the current URL to one or more route handlers. A route handler can do several things:

- It can render a template.
- It can load a model that is then available to the template.
- It can redirect to a new route, such as if the user isn't allowed to visit that part of the app.
- It can handle actions that involve changing a model or transitioning to a new route.

ember
ABOUT
GUIDES
API
COMMUNITY
BLOG
BUILDS
Search the guide
FORK US!
v2.0.0
GETTING STARTED
THE OBJECT MODEL
ROUTING
Introduction
Defining Your Routes
Specifying a Route's Model
Rendering a Template
Redirecting
Preventing and Retrying Transitions
Loading / Error Substates
Query Parameters
Asynchronous Routing
TEMPLATES
COMPONENTS
CONTROLLERS
MODELS
VIEW v2.12.0

Old Guides - You are viewing the guides for Ember v2.0.0.

INTRODUCTION

Imagine we are writing a web app for managing a blog. At any given time, we should be able to answer questions like *What post are they looking at?* and *Are they editing it?* In Ember.js, the answer to these questions is determined by the URL.

The URL can be set in a few ways:

- The user loads the app for the first time.
- The user changes the URL manually, such as by clicking the back button or by editing the address bar.
- The user clicks a link within the app.
- Some other event in the app causes the URL to change.

Regardless of how the URL becomes set, the Ember router then maps the current URL to one or more route handlers. A route handler can do several things:

- It can render a template.
- It can load a model that is then available to the template.
- It can redirect to a new route, such as if the user isn't allowed to visit that part of the app.
- It can handle actions that involve changing a model or transitioning to a new route.

apps
ember
ABOUT
GUIDES
API
COMMUNITY
BLOG
BUILDS
Search the guide
FORK US!
v1.13.0
GETTING STARTED
THE OBJECT MODEL
TEMPLATES
ROUTING
COMPONENTS
CONTROLLERS
MODELS
TESTING
EMBER INSPECTOR
CONFIGURING EMBER.JS
UNDERSTANDING EMBER.JS
CONTRIBUTING TO EMBER.JS

GUIDES AND TUTORIALS

Welcome to the Ember.js guides! This documentation will take you from total beginner to Ember expert. It is designed to start from the basics, and slowly increase to more sophisticated concepts until you know everything there is to know about building awesome web applications.

These guides are written in Markdown and are available on GitHub. If there is something missing, or you find a typo or mistake, please help us by filing an issue or submitting a pull request. Thanks!

We're excited for all of the great apps you're going to build with Ember.js. To get started, select a topic from the left. They are presented in the order that we think will be most useful to you as you're learning Ember.js, but you can also jump to whatever seems most interesting.

Good luck!

We're done with Guides and Tutorials. Next up: Getting Started - Installing Ember →