Building rich one-page applications with Ember.js

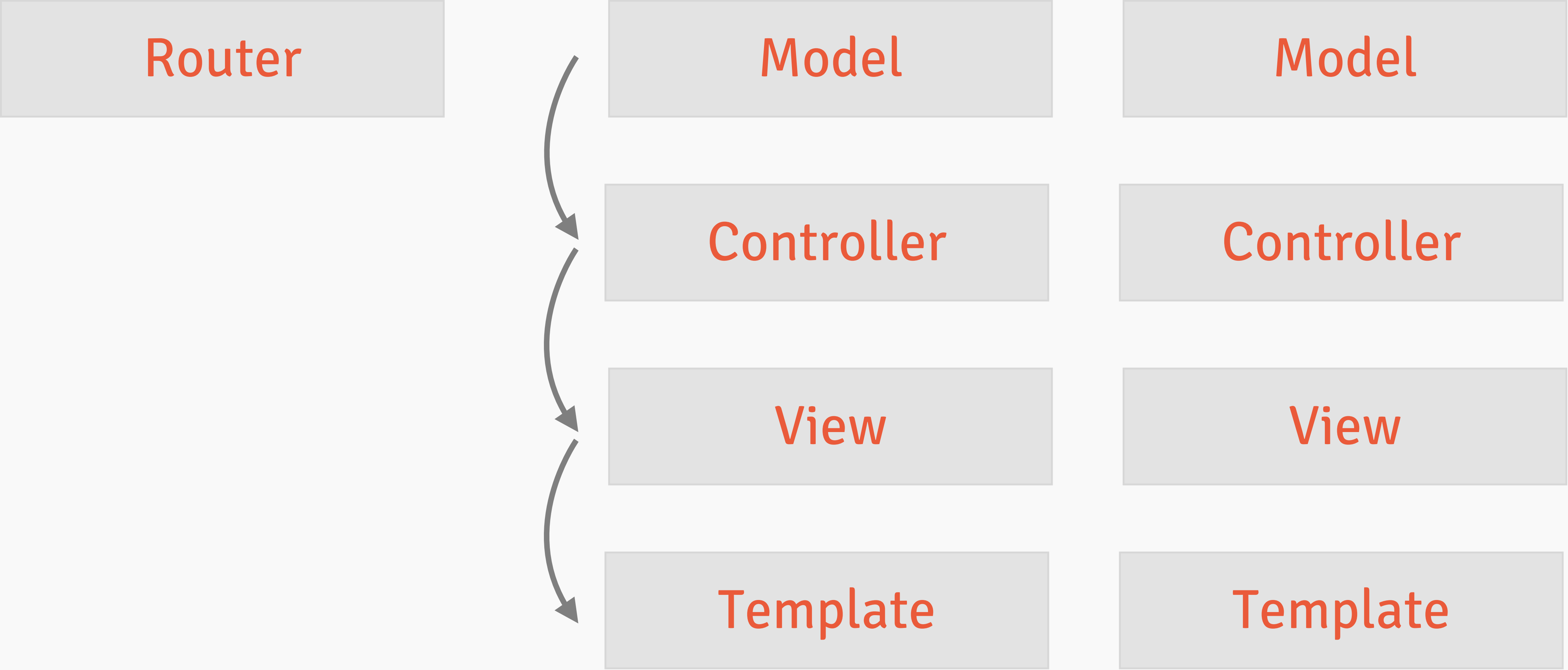
Serge Barysiuk, Quote Roller
serge@quoteroller.com

ember

Frontend DEV Conf `13

quoteroller

Concepts



Router

Model

Controller

View

Template

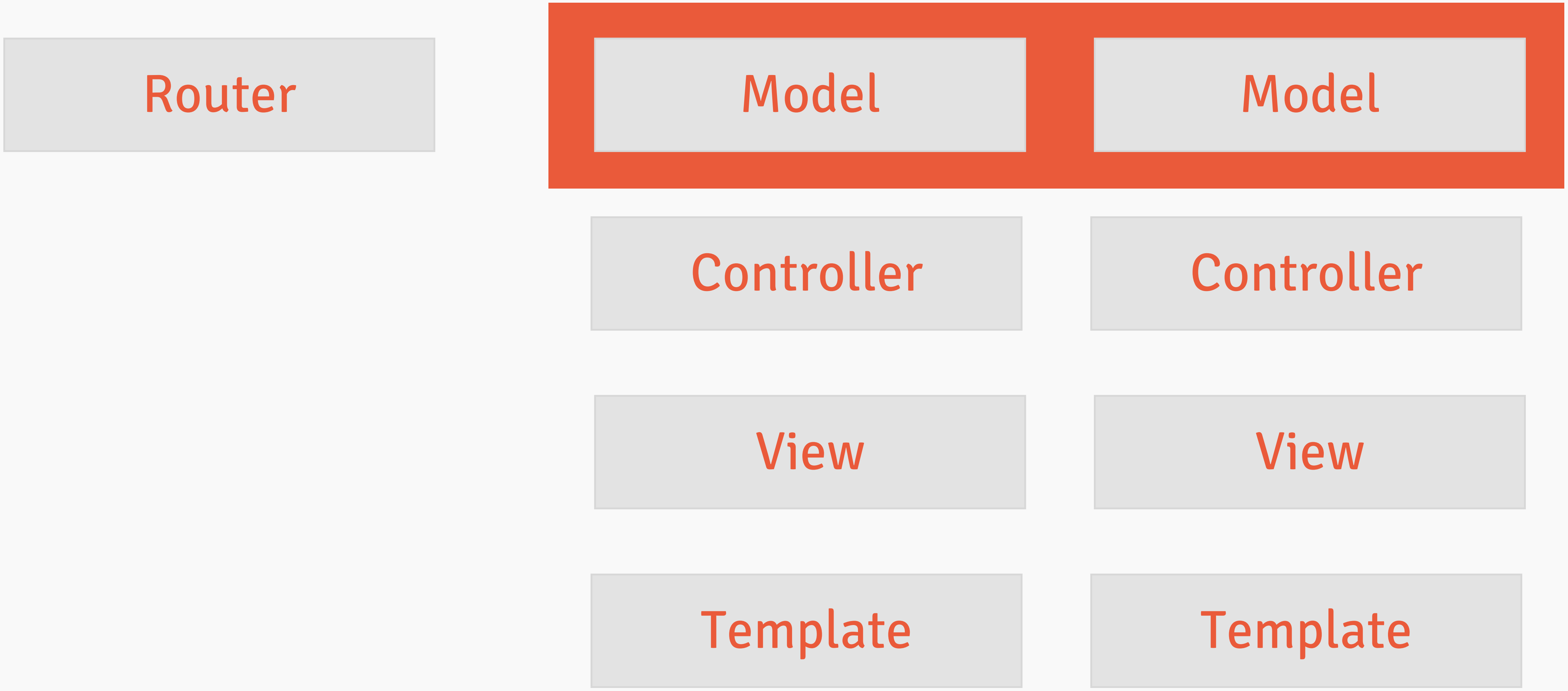
Model

Controller

View

Template

Models



Router

Model Model

Controller Controller

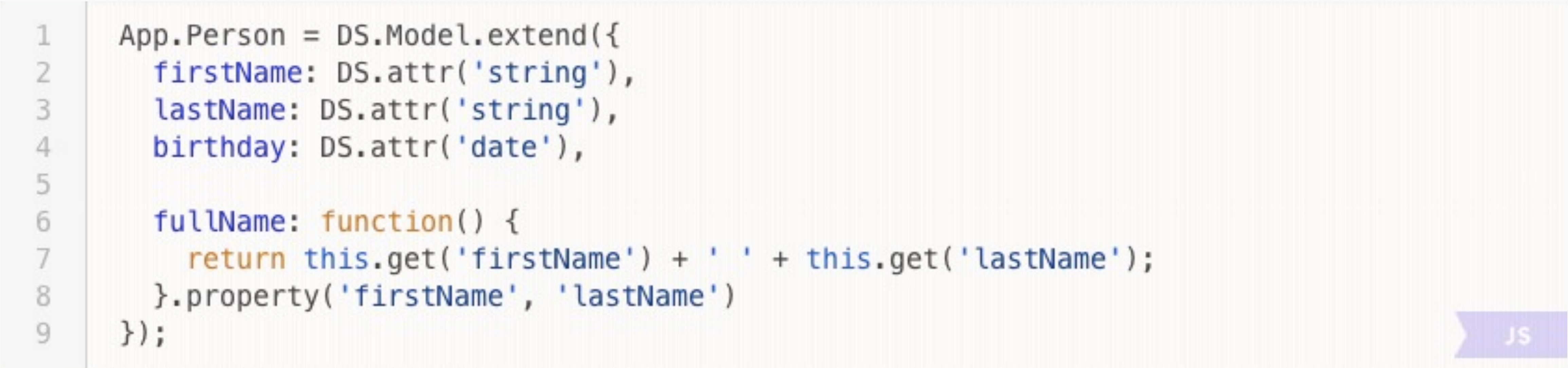
View View

Template Template

Models

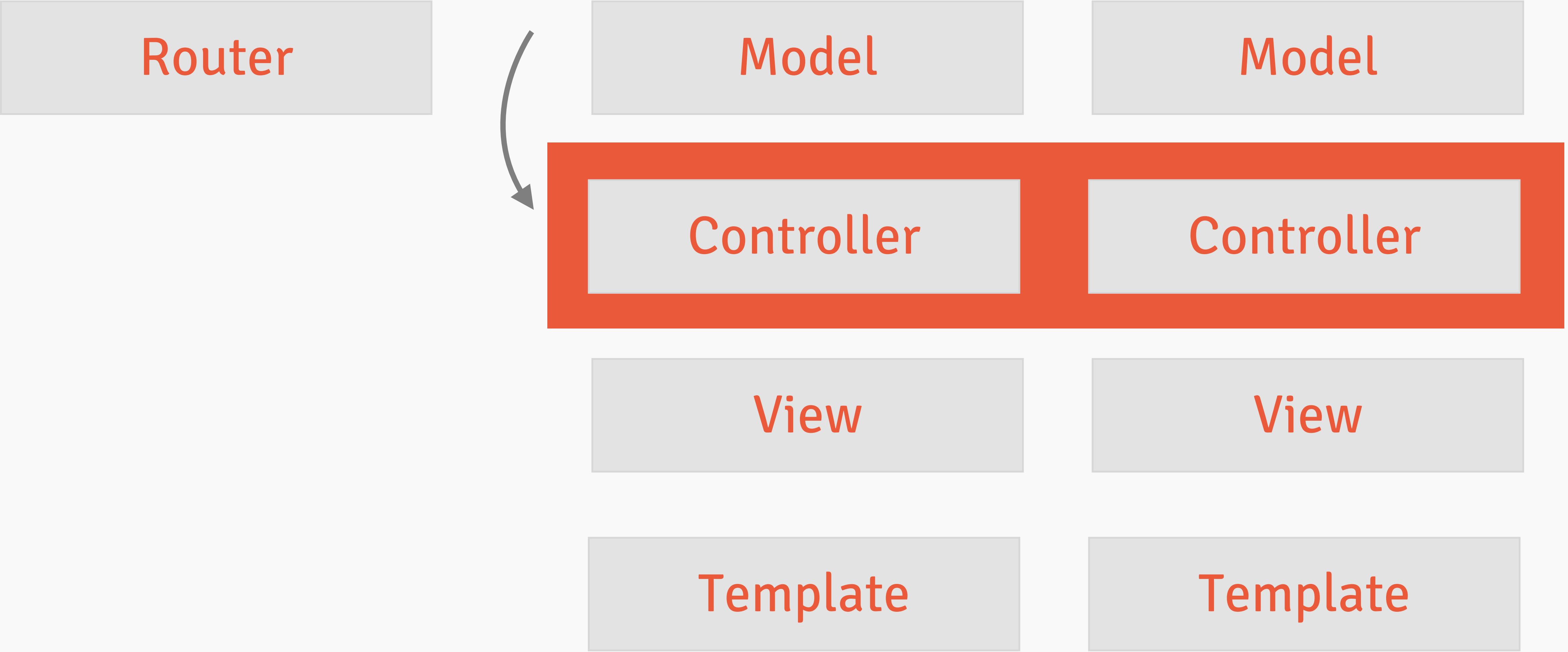
- ember-data
- DS.Store
- DS.RESTAdapter & DS.BasicAdapter

Models



```
App.Person = DS.Model.extend({
  firstName: DS.attr('string'),
  lastName: DS.attr('string'),
  birthday: DS.attr('date'),

  fullName: function() {
    return this.get('firstName') + ' ' + this.get('lastName');
  }.property('firstName', 'lastName')
});
```

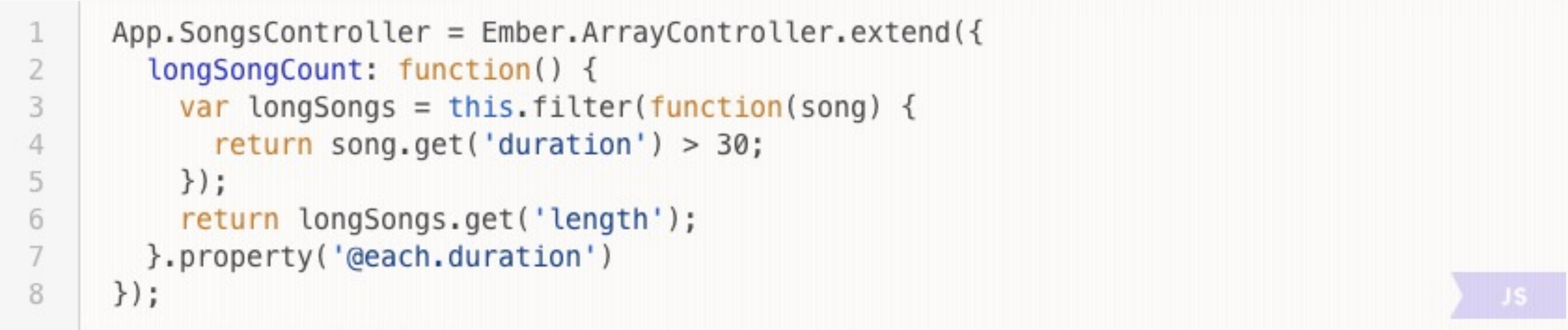
Controllers



Controllers

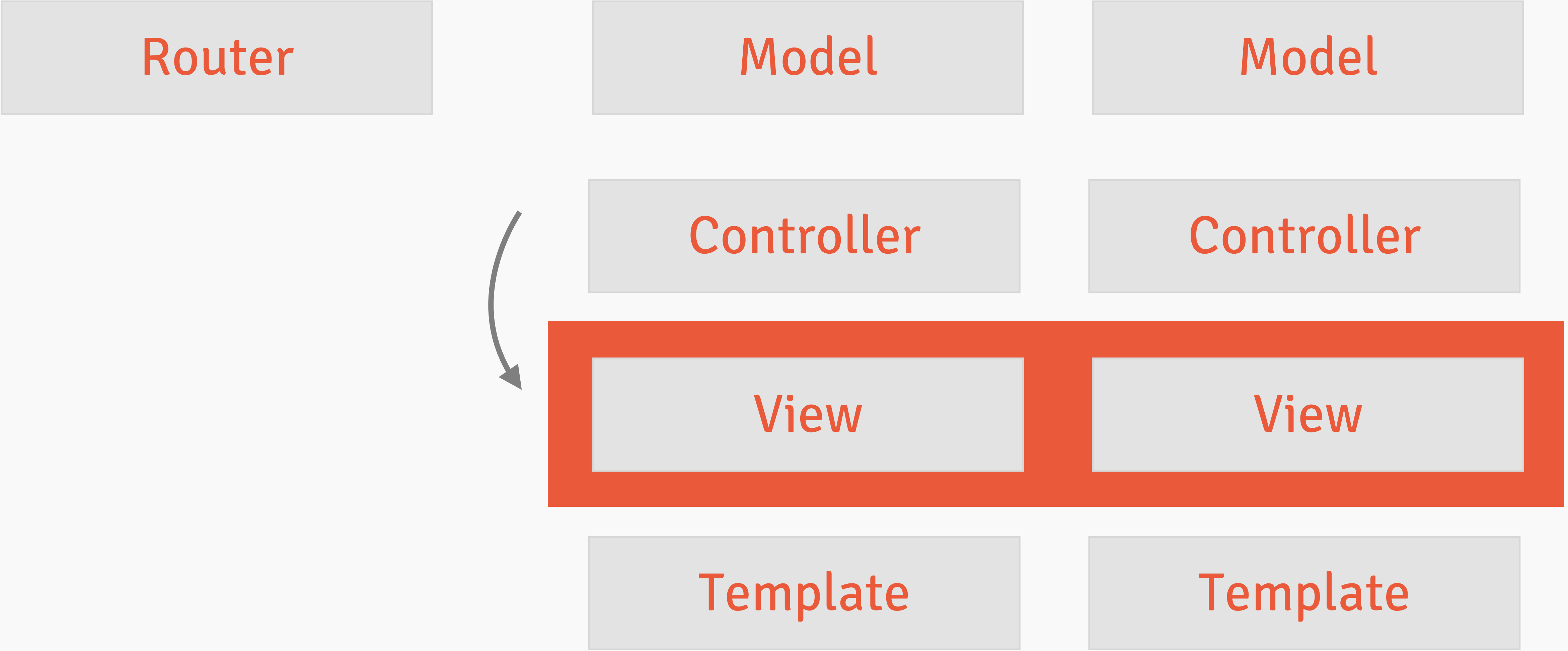
- Present data for view layer to render
- Expose bindable properties
- Work like a proxy for models
- ObjectController, ArrayController

Controllers



```
App.SongsController = Ember.ArrayController.extend({
  longSongCount: function() {
    var longSongs = this.filter(function(song) {
      return song.get('duration') > 30;
    });
    return longSongs.get('length');
  }.property('@each.duration')
});
```

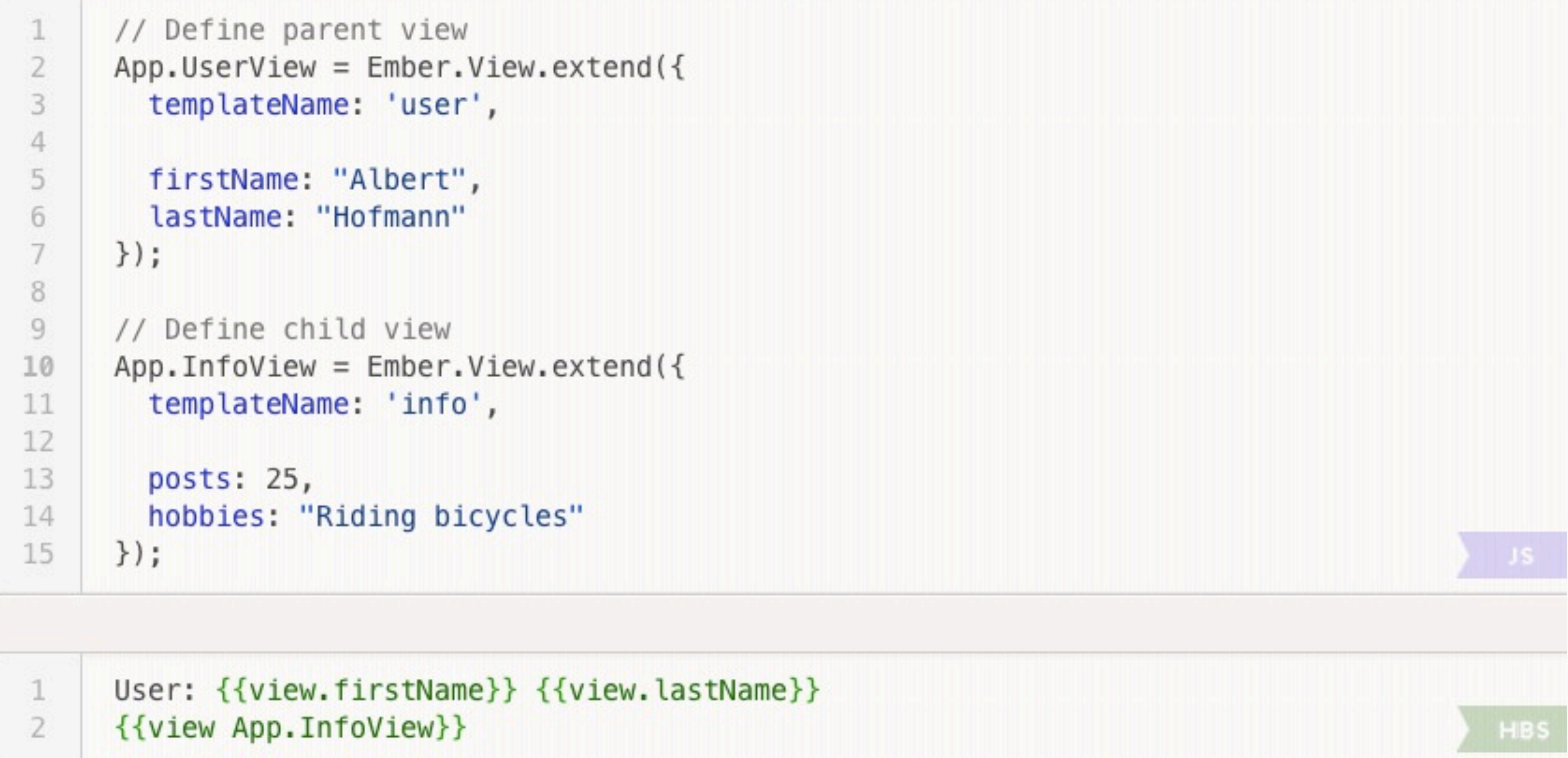
Views


Router
Model
Model
Controller
Controller
View
View
Template
Template

Views

- When you need sophisticated handling of user events
- When you want to create a re-usable component

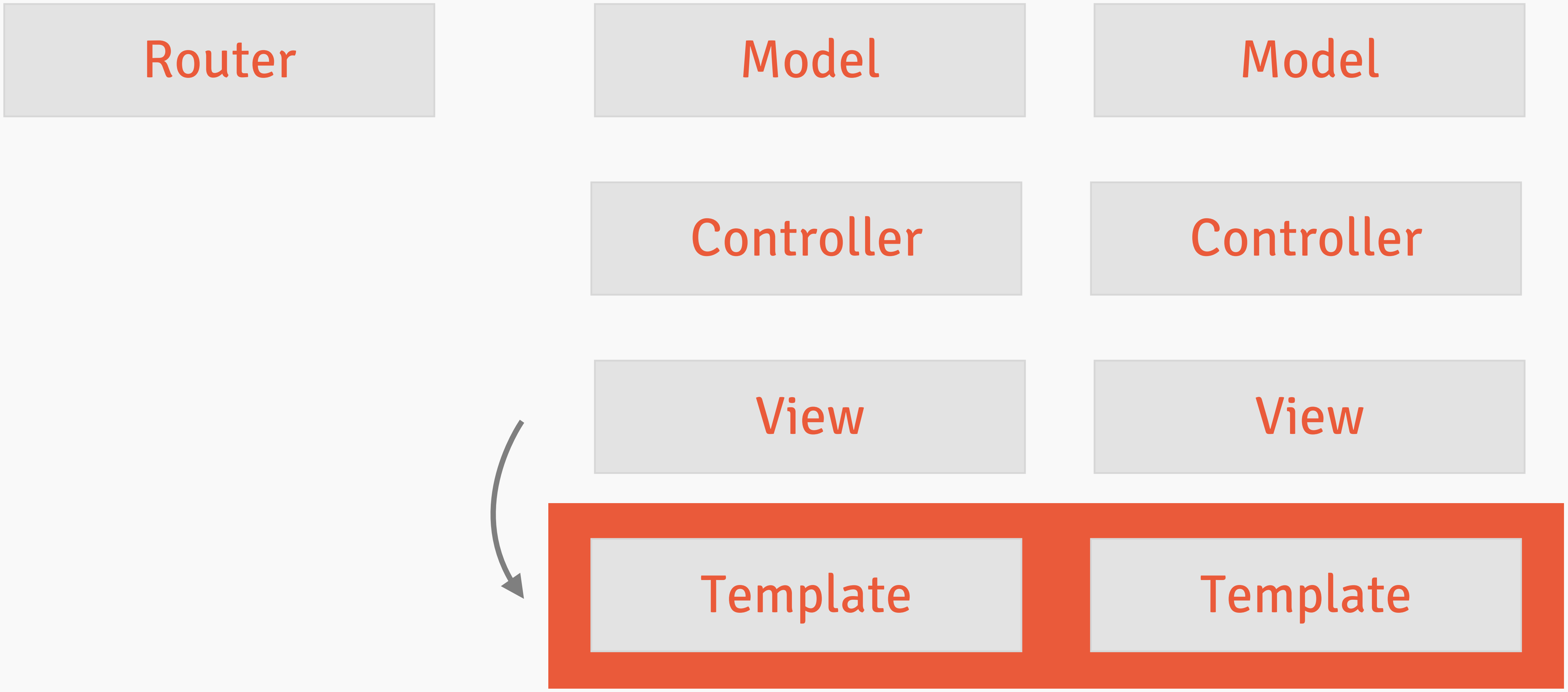
Views



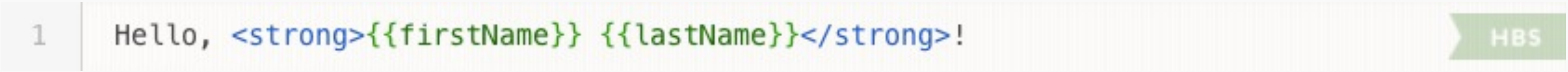
```javascript
// Define parent view
App.UserView = Ember.View.extend({
  templateName: 'user',

  firstName: "Albert",
  lastName: "Hofmann"
});

// Define child view
App.InfoView = Ember.View.extend({
  templateName: 'info',

  posts: 25,
  hobbies: "Riding bicycles"
});
```

```handlebars
User: {{view.firstName}} {{view.lastName}}
{{view App.InfoView}}
```

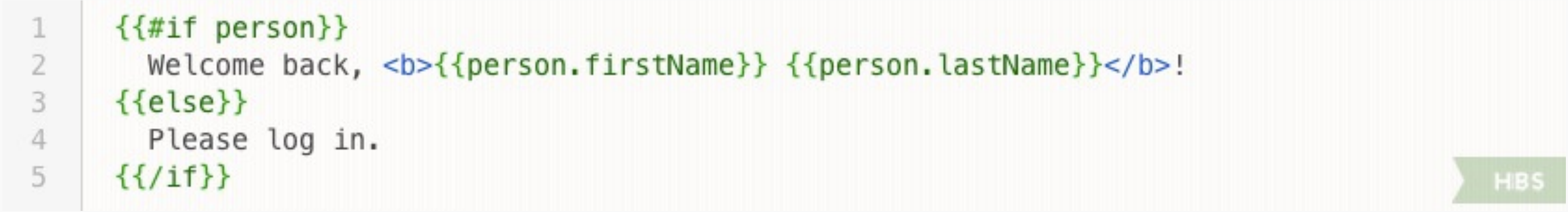
Templates


Router
Model
Model
Controller
Controller
View
View
Template
Template

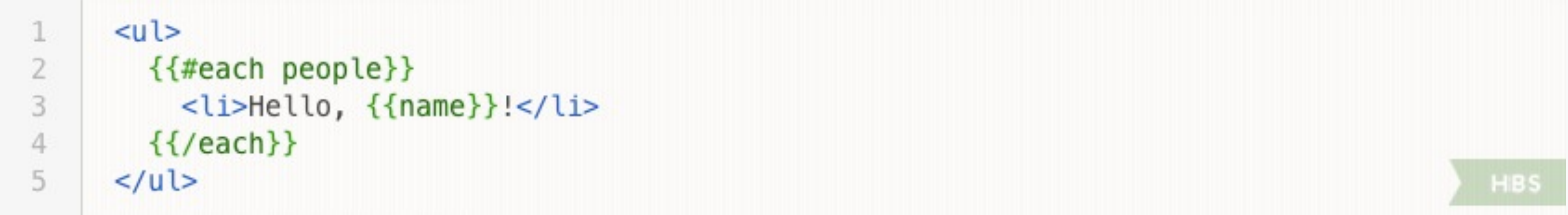
Templates



```
Hello, <strong>{{firstName}} {{lastName}}</strong>!
```

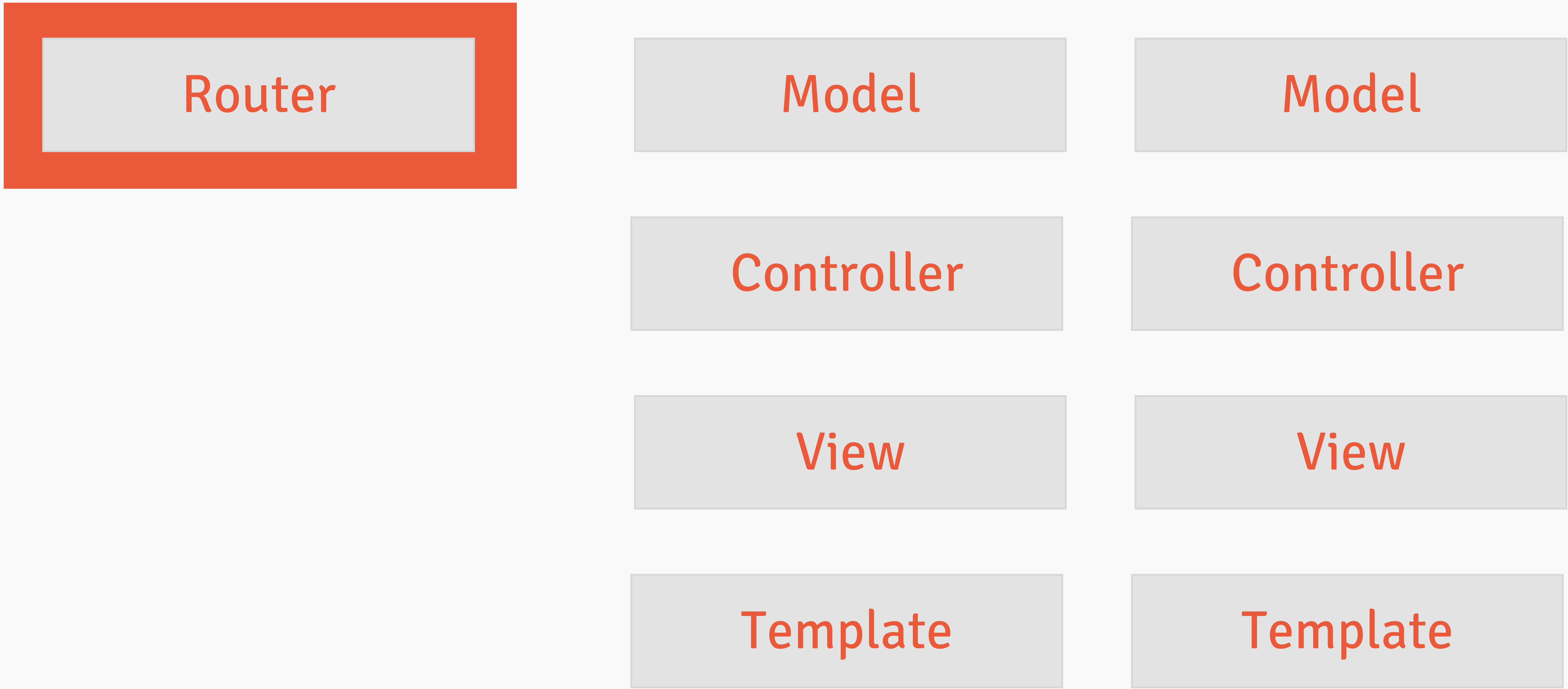


```
{{#if person}}
  Welcome back, <b>{{person.firstName}} {{person.lastName}}</b>!
{{else}}
  Please log in.
{{/if}}
```



```
<ul>
  {{#each people}}
    <li>Hello, {{name}}!</li>
  {{/each}}
</ul>
```

Router



Router

Model

Model

Controller

Controller

View

View

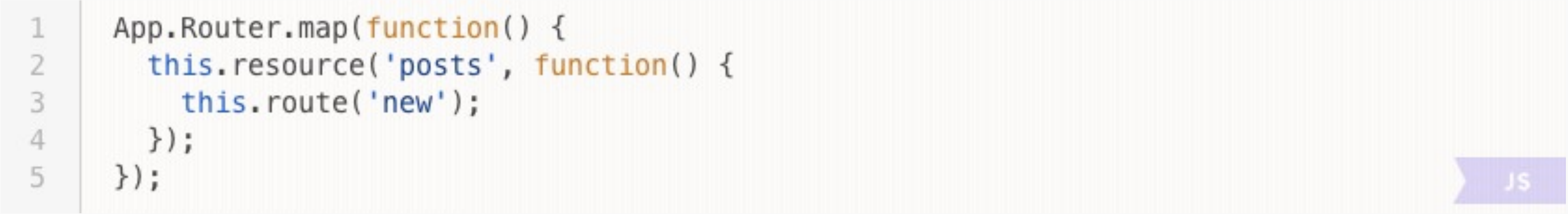
Template

Template

Router

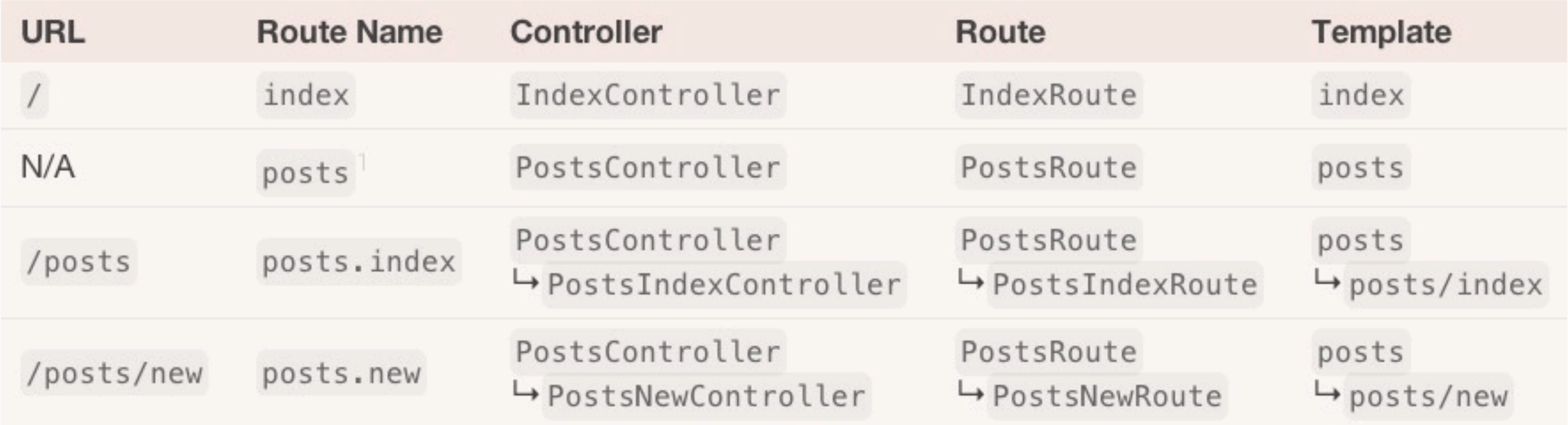
- Manages application state
- Keeps the state up to date with the URL

Router



```
App.Router.map(function() {
  this.resource('posts', function() {
    this.route('new');
  });
});
```

JS



URL	Route Name	Controller	Route	Template
`/`	`index`	`IndexController`	`IndexRoute`	`index`
N/A	`posts`[1]	`PostsController`	`PostsRoute`	`posts`
`/posts`	`posts.index`	`PostsController` ↳ `PostsIndexController`	`PostsRoute` ↳ `PostsIndexRoute`	`posts` ↳ `posts/index`
`/posts/new`	`posts.new`	`PostsController` ↳ `PostsNewController`	`PostsRoute` ↳ `PostsNewRoute`	`posts` ↳ `posts/new`

Let’s code!

goo.gl/KKV8F

Next steps

- emberjs.com & @emberjs
- emberwatch.com & @emberwatch
- stackoverflow.com
- https://github.com/discourse/discourse

Looking for a
JS/Frontend ninja.

serge@quoteroller.com

KiMA VENTURES Google intuit

ORACLE fotolia