sitepoint premium

Handout

Getting Started with Backbone.js

Lesson 5 - Step 2

Ilya Bodrov

Creating router

We create a router by using basic extending functionality:

```
Organizer.Router = Backbone.Router.extend({

});
```

Inside we have to provide the `routes` object. The key is the path and the value is the function's name to fire when a user visits this path:

```
Organizer.Router = Backbone.Router.extend({
  routes: {
    'test': 'greet'
  },
  greet: function() {
    console.log('greet');
  }
});
```

Instantiate router in the `initialize` function inside the *app.js* file:

```
Organizer.router = new Organizer.Router();
```

Navigate to the new route: *http://127.0.0.1/organizer/index.html#test*. However, nothing will happen. We also have to boot the special Backbone's history API:

```
Backbone.history.start();
```

This API is going to record where we are navigating and allow us to use Back and Forward buttons in the browser.

Navigate method

We can also programatically update the URL and fire the corresponding action:

```
Organizer.router.navigate('test');
```

The URL was updated however we see no message in the console. The route's action will not performed until we pass the `trigger: true` option:

```
Organizer.router.navigate('test', {trigger: true});
```

If you do not wish to make an entry in the browser's history, pass `replace: true`:

```
Organizer.router.navigate('test', {trigger: true, replace: true});
```

If you wish to use History API instead of hash symbols, provide `pushState` option:

```
Backbone.history.start({pushState: true})
```

Route method

If you wish to generate route on the fly, call the `route` method:

```
Organizer.router.route('test', 'test', function() {console.log('test page')});
```

The first argument is the actual URL portion, the second is the name and the third is the optional callback.

If the callback is not provided, the handler with that name is being called:

```
Organizer.router.route('test_handler', 'greet');
Organizer.router.navigate('test_handler', {trigger: true});
```

The `greet` function will be called.

You may also be wondering whether jQuery's page load event is fired each time we use navigate. Let's check that out:

```
$(document).ready(function() {
  Organizer.initialize();
  console.log('page is loaded');
});

Organizer.router.navigate('test', {trigger: true});
```

So it is not firing.