Real Routes

- LEVEL 3 -

Optional Routes

Search route with query and optional page parameter

```
var TodoRouter = new (Backbone.Router.extend({
  routes: {
    'search/:query': 'search',
    'search/:query/p:page': 'search'
  },
  search: function(query, page) {
    page = page || 0;
    console.log(query);
    console.log(page);
  }
}));
```

Duplicate Routes

```
TodoRouter.navigate('search/milk', {trigger: true});
```

→ `"milk", 0`

```
TodoRouter.navigate('search/milk/p2', {trigger: true});
```

→ `"milk", 2`

Optional Routes

```
var TodoRouter = new (Backbone.Router.extend({
  routes: {
    'search/:query(/p:page)': 'search',
  },
  search: function(query, page) {
    page = page || 0;
    console.log(query);
    console.log(page);
  }
}));
```

Optional Route Part

```
TodoRouter.navigate('search/milk', {trigger: true});
```

→

```
"milk", 0
```

```
TodoRouter.navigate('search/milk/p2', {trigger: true});
```

→

```
"milk", 2
```

What if we want to match 'search/milk/p2/'?

Optional Trailing Slash

```
var TodoRouter = new (Backbone.Router.extend({
  routes: {
    'search/:query(/p:page)(/)': 'search',
  },
  search: function(query, page) {
    page = page || 0;
    console.log(query);
    console.log(page);
  }
}));
```

Optional trailing slash

```
TodoRouter.navigate('search/milk/p2/', {trigger: true});
```

--→

```
"milk", 2
```

URI with Spaces Gotcha

```
var TodoRouter = new (Backbone.Router.extend({
  routes: {
    'search/:query(/p:page)(/)': 'search',
  },
  search: function(query, page) {
    page = page || 0;
    console.log(query);
    console.log(page);
  }
}));
```

Encoded space

Search for "Hello World"

```
TodoRouter.navigate('search/Hello%20World/p2', {trigger: true})
```

```
"Hello%20World", 2
```

Not decoded

URI with Spaces Gotcha

```
var TodoRouter = new (Backbone.Router.extend({
  routes: {
    'search/:query(/p:page)(/)': 'search',
  },
  search: function(query, page) {
    page = page || 0;
    query = decodeURIComponent(query);
    console.log(query);
    console.log(page);
  }
}));
```

Encoded space

```
TodoRouter.navigate('search/Hello%20World/p2', {trigger: true})
```

```
"Hello World", 2
```

✓ **Decoded**

Regex in Routes

Restrict parameter to numeric input

```
var TodoRouter = new (Backbone.Router.extend({
  routes: {
    'todos/:id': 'show'
  },
  show: function(id) {
    console.log("id = " + id);
  }
}));
```

```
TodoRouter.navigate('todos/2', {trigger: true})
```

---> id = 2

```
TodoRouter.navigate('todos/notanid', {trigger: true})
```

--> id = notanid

✕ **Shouldn’t trigger show**

Regex in Routes

```
var TodoRouter = new (Backbone.Router.extend({
  show: function(id) {
    console.log("id = " + id);
  }
}));
```

Each Regex Capture Group becomes a param

```
TodoRouter.route(/^todos\/(\d+)$/, 'show');
```

```
TodoRouter.navigate('todos/2', {trigger: true})
```

id = 2

```
TodoRouter.navigate('todos/notanid', {trigger: true})
```

✓ Doesn’t trigger show

Regex in Routes Refactor

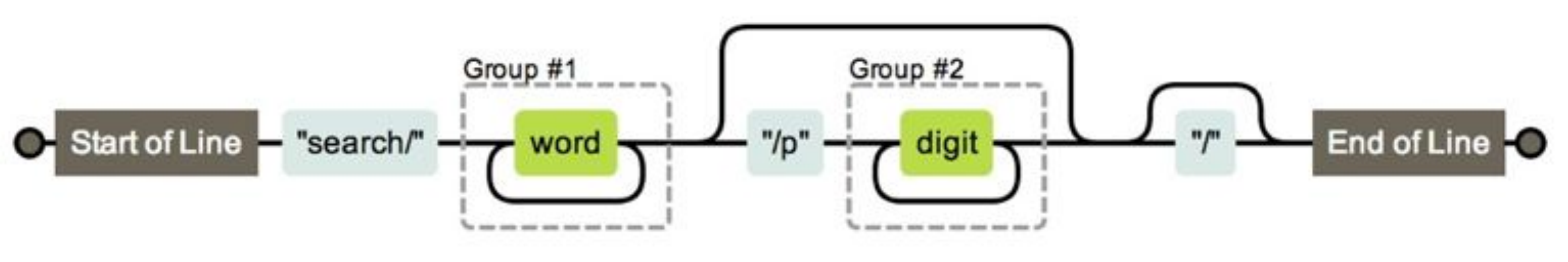
```
var TodoRouter = new (Backbone.Router.extend({
  initialize: function(){
    this.route(/^todos\/(\d+)$/, 'show');
  },
  show: function(id) {
    console.log("id = " + id);
  }
}));
```

Keep routes inside the router

Advanced Regex Routes

```
/^search\/(\w+)(?:\/p(\d+))?\/?$/
```

--> `'search/:query(/p:page)'`



Matches

```
'search/milk'
'search/milk/p2'
'search/milk/p2/'
```

Doesn't Match

```
'searches/milk'
'search/milk/p2/p1'
'search/milk/2'
```

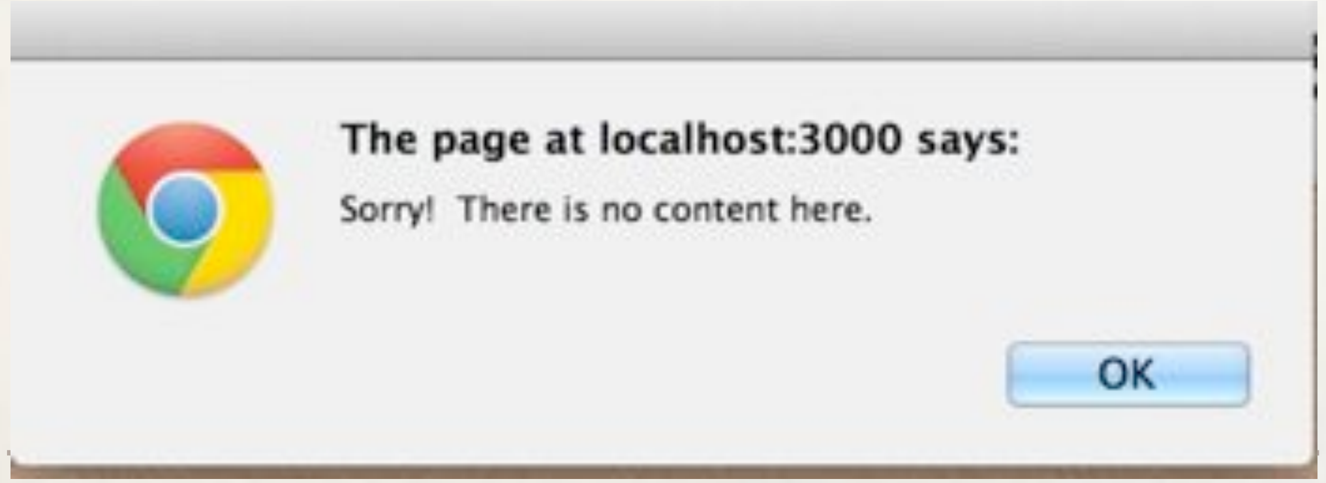
http://www.regexper.com/

Catch-all Routes

Alert user when no route matches

```
var TodoRouter = Backbone.Router.extend({
  routes: {
    '*path': 'notFound'
  },
  notFound: function(path) {
    alert('Sorry!  There is no content here.');
  }
});
```

```
TodoRouter.navigate('nothinghere', {trigger: true})
```



The page at localhost:3000 says:

Sorry! There is no content here.

File Path Route

Accept a file path and get its parts

```
var TodoRouter = new (Backbone.Router.extend({
  routes: {
    'file/*path': 'file'
  },
  file: function(path) {
    var parts = path.split("/");
    console.log(parts);
  }
}));
```

```
TodoRouter.navigate("file/this/is/a/file.txt", {trigger: true});
```

```
["this", "is", "a", "file.txt"]
```

Real Routes

- LEVEL 3 -