REACTJS

Ruthran.kumar@gmail.com

+44 7405708980

+91 9715531887

Agenda

- Overview
- Environment Setup
- Component
- Parent to Child data communication
- Form
- Validation
- Routing
- Flux (Unidirectional Design Pattern)

Overview

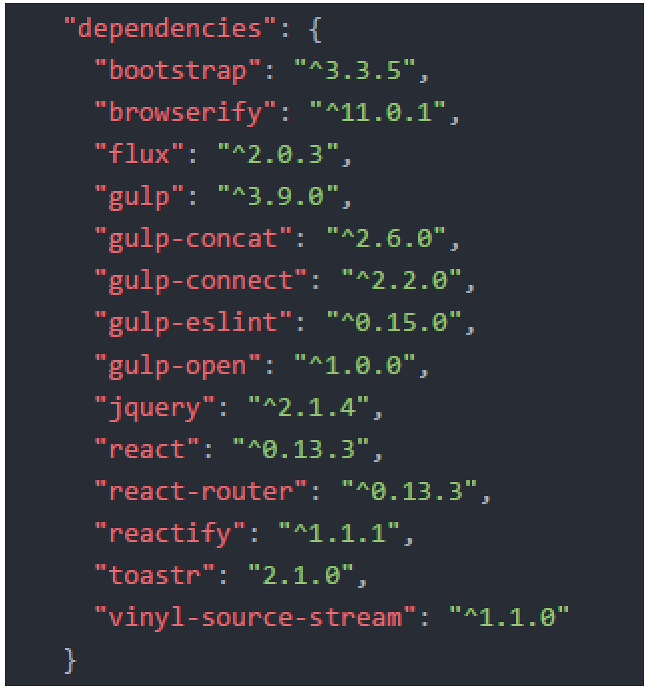
- React is a UI library developed at Facebook to facilitate the creation of interactive, stateful & reusable UI components.
- JSX is JavaScript syntax extension. It isn't necessary to use JSX in React development, but it is recommended.
- React is all about components. You need to think of everything as a component. This will help you maintain the code when working on larger scale projects.
- React implements one-way data flow which makes it easy to reason about your app. Flux is a pattern that helps keeping your data unidirectional.
- Uses virtual DOM JavaScript virtual DOM is faster than the regular DOM.

Environment Setup

By Starter Kit https://github.com/coryhouse/react-flux-starter-kit

npm install

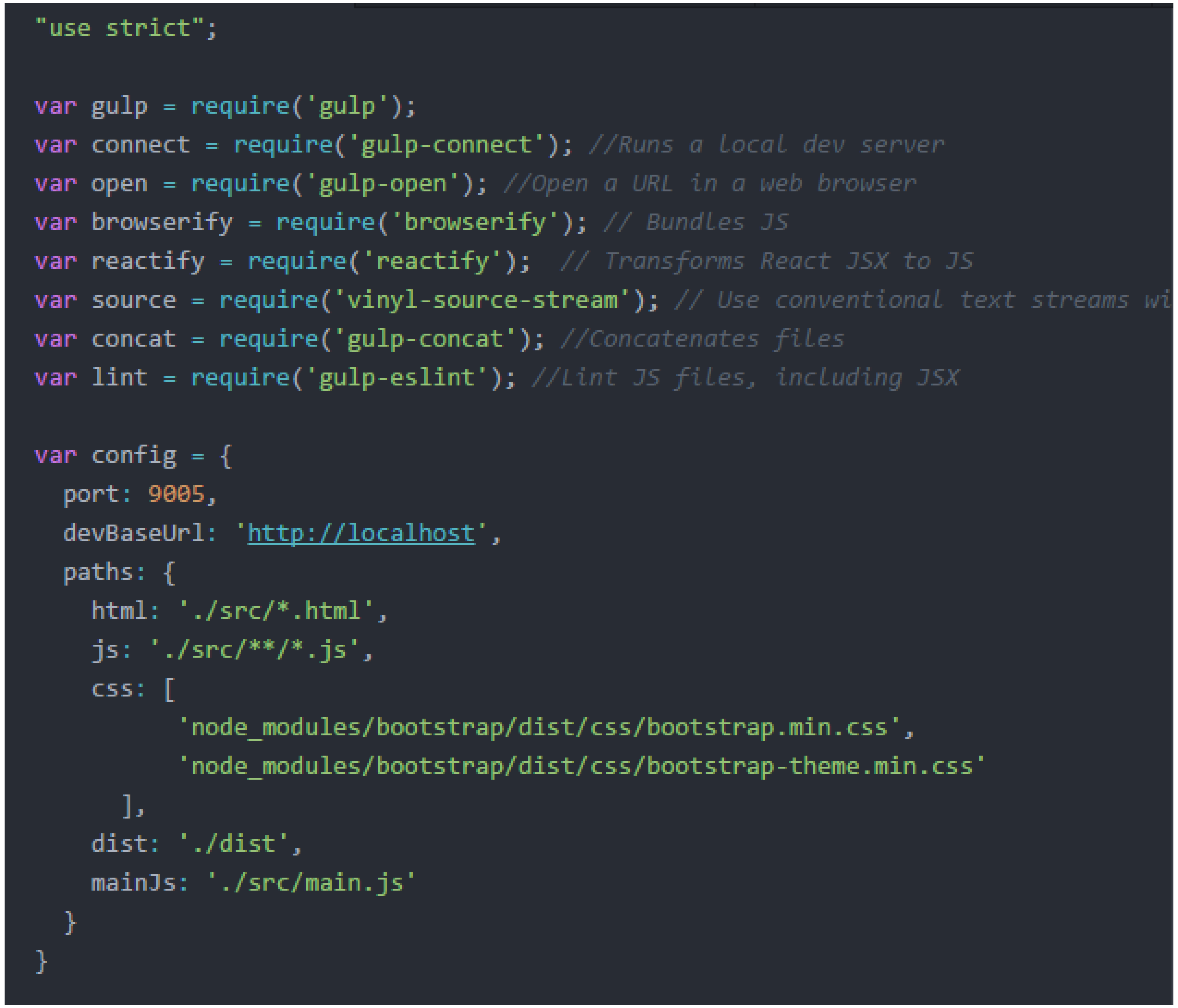
Package JSON



```
  "dependencies": {
    "bootstrap": "^3.3.5",
    "browserify": "^11.0.1",
    "flux": "^2.0.3",
    "gulp": "^3.9.0",
    "gulp-concat": "^2.6.0",
    "gulp-connect": "^2.2.0",
    "gulp-eslint": "^0.15.0",
    "gulp-open": "^1.0.0",
    "jquery": "^2.1.4",
    "react": "^0.13.3",
    "react-router": "^0.13.3",
    "reactify": "^1.1.1",
    "toastr": "2.1.0",
    "vinyl-source-stream": "^1.1.0"
  }
```

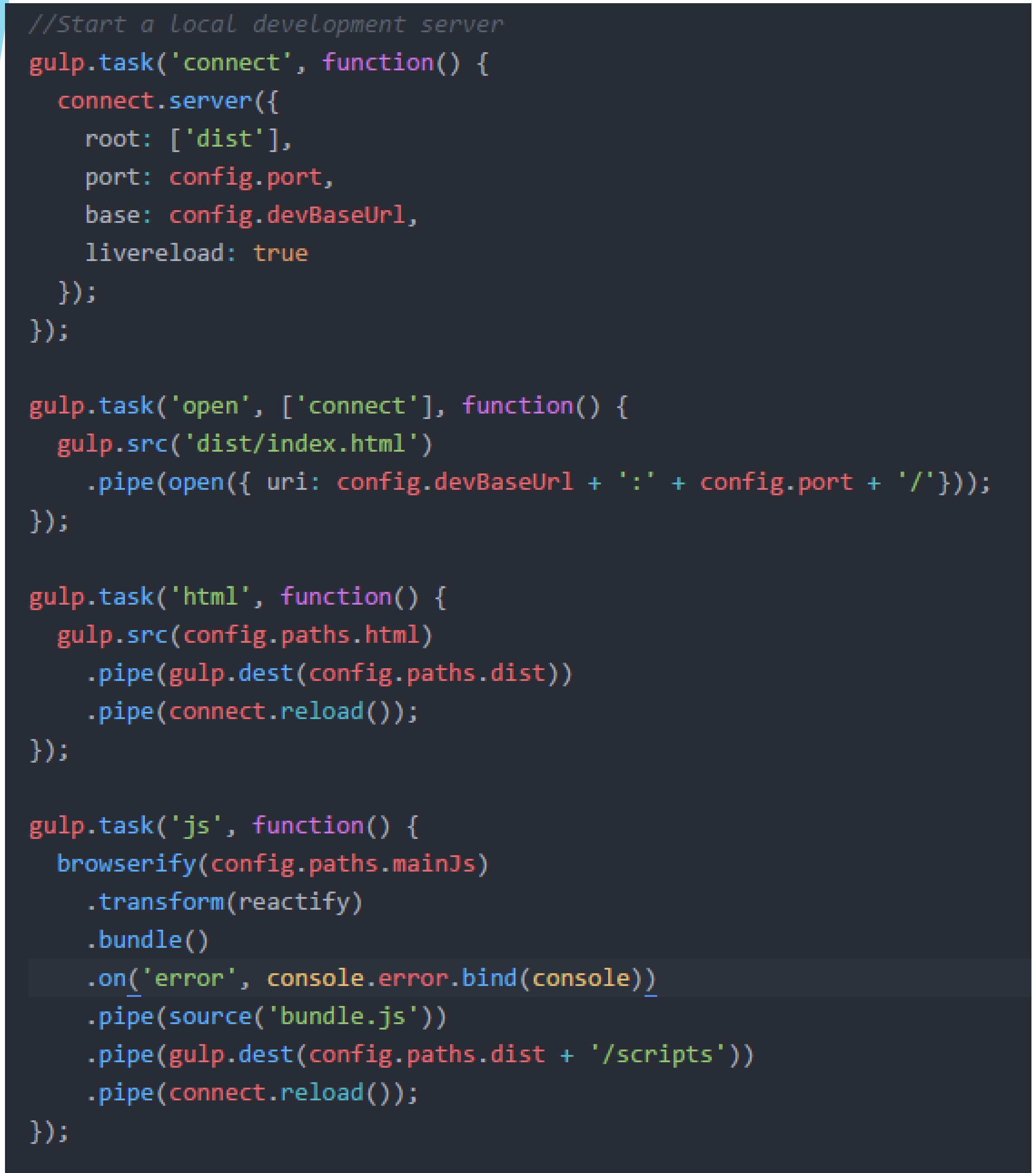
Environment Setup

Gulp.js

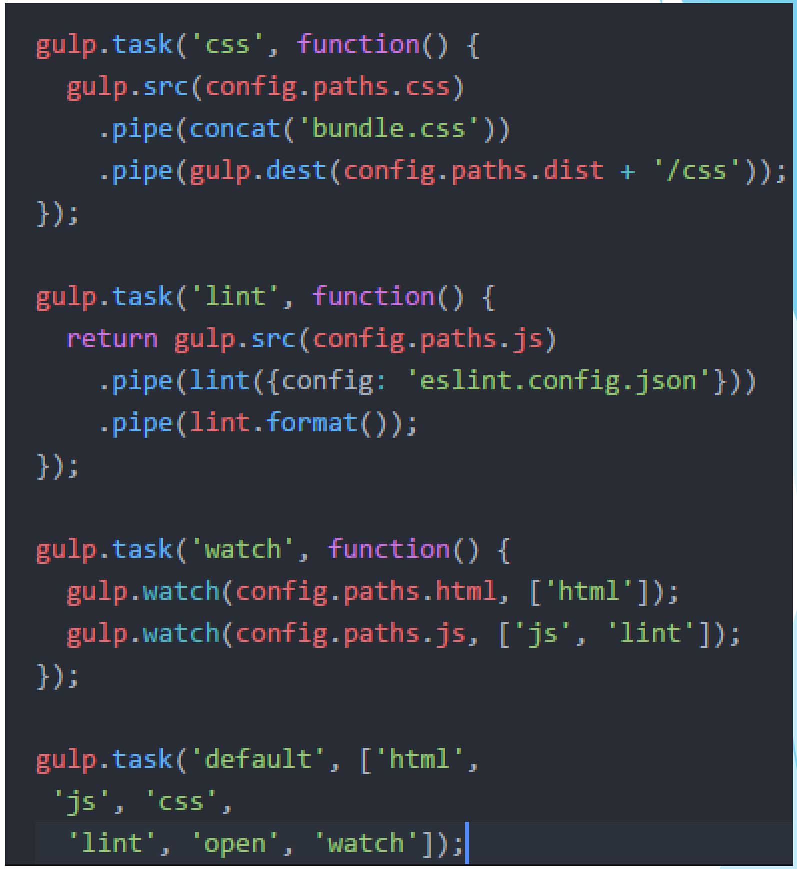


```
"use strict";

var gulp = require('gulp');
var connect = require('gulp-connect'); //Runs a local dev server
var open = require('gulp-open'); //Open a URL in a web browser
var browserify = require('browserify'); // Bundles JS
var reactify = require('reactify');  // Transforms React JSX to JS
var source = require('vinyl-source-stream'); // Use conventional text streams wi
var concat = require('gulp-concat'); //Concatenates files
var lint = require('gulp-eslint'); //Lint JS files, including JSX

var config = {
  port: 9005,
  devBaseUrl: 'http://localhost',
  paths: {
    html: './src/*.html',
    js: './src/**/*.js',
    css: [
          'node_modules/bootstrap/dist/css/bootstrap.min.css',
          'node_modules/bootstrap/dist/css/bootstrap-theme.min.css'
      ],
    dist: './dist',
    mainJs: './src/main.js'
  }
}
```

Gulp.js Cont.

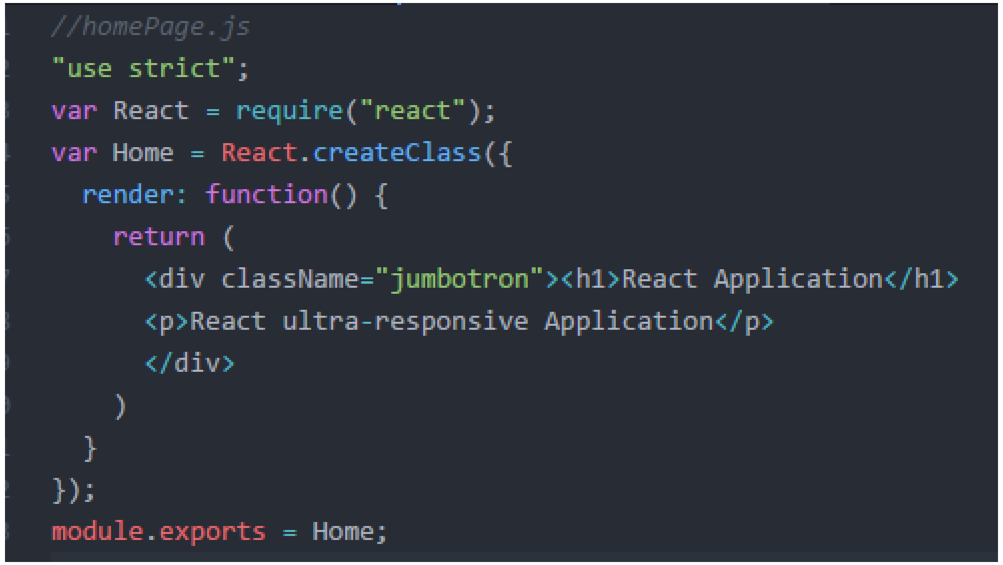


```
//Start a local development server
gulp.task('connect', function() {
  connect.server({
    root: ['dist'],
    port: config.port,
    base: config.devBaseUrl,
    livereload: true
  });
});

gulp.task('open', ['connect'], function() {
  gulp.src('dist/index.html')
    .pipe(open({ uri: config.devBaseUrl + ':' + config.port + '/'}));
});

gulp.task('html', function() {
  gulp.src(config.paths.html)
    .pipe(gulp.dest(config.paths.dist))
    .pipe(connect.reload());
});

gulp.task('js', function() {
  browserify(config.paths.mainJs)
    .transform(reactify)
    .bundle()
    .on('error', console.error.bind(console))
    .pipe(source('bundle.js'))
    .pipe(gulp.dest(config.paths.dist + '/scripts'))
    .pipe(connect.reload());
});
```

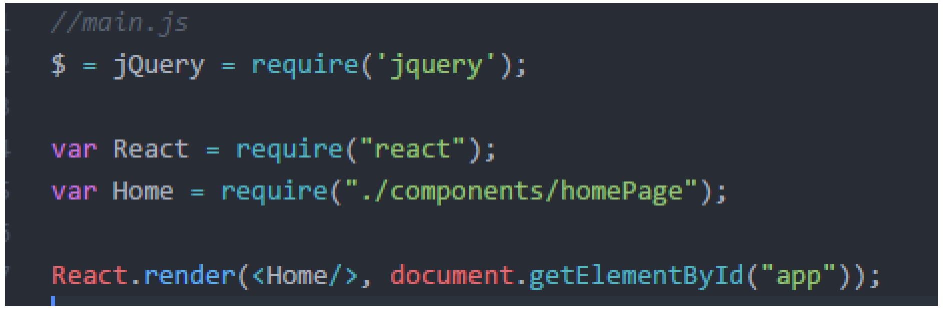


```
gulp.task('css', function() {
  gulp.src(config.paths.css)
    .pipe(concat('bundle.css'))
    .pipe(gulp.dest(config.paths.dist + '/css'));
});

gulp.task('lint', function() {
  return gulp.src(config.paths.js)
    .pipe(lint({config: 'eslint.config.json'}))
    .pipe(lint.format());
});

gulp.task('watch', function() {
  gulp.watch(config.paths.html, ['html']);
  gulp.watch(config.paths.js, ['js', 'lint']);
});

gulp.task('default', ['html',
 'js', 'css',
  'lint', 'open', 'watch']);
```

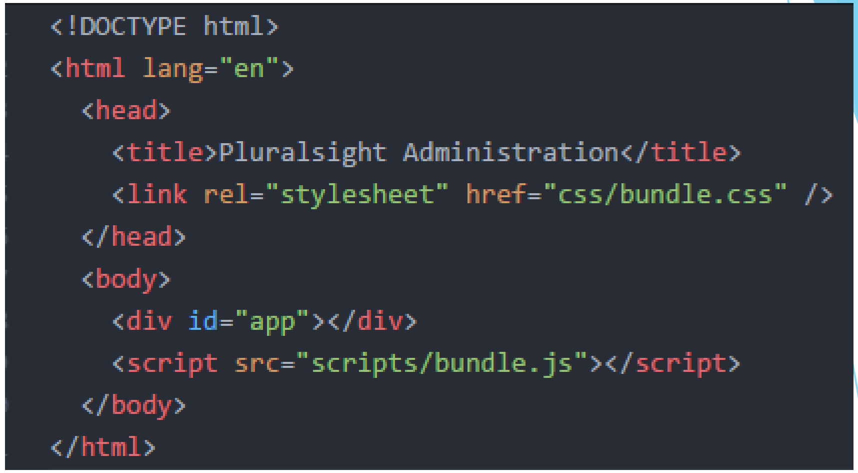
Component



```
//homePage.js
"use strict";
var React = require("react");
var Home = React.createClass({
  render: function() {
    return (
      <div className="jumbotron"><h1>React Application</h1>
      <p>React ultra-responsive Application</p>
      </div>
    )
  }
});
module.exports = Home;
```

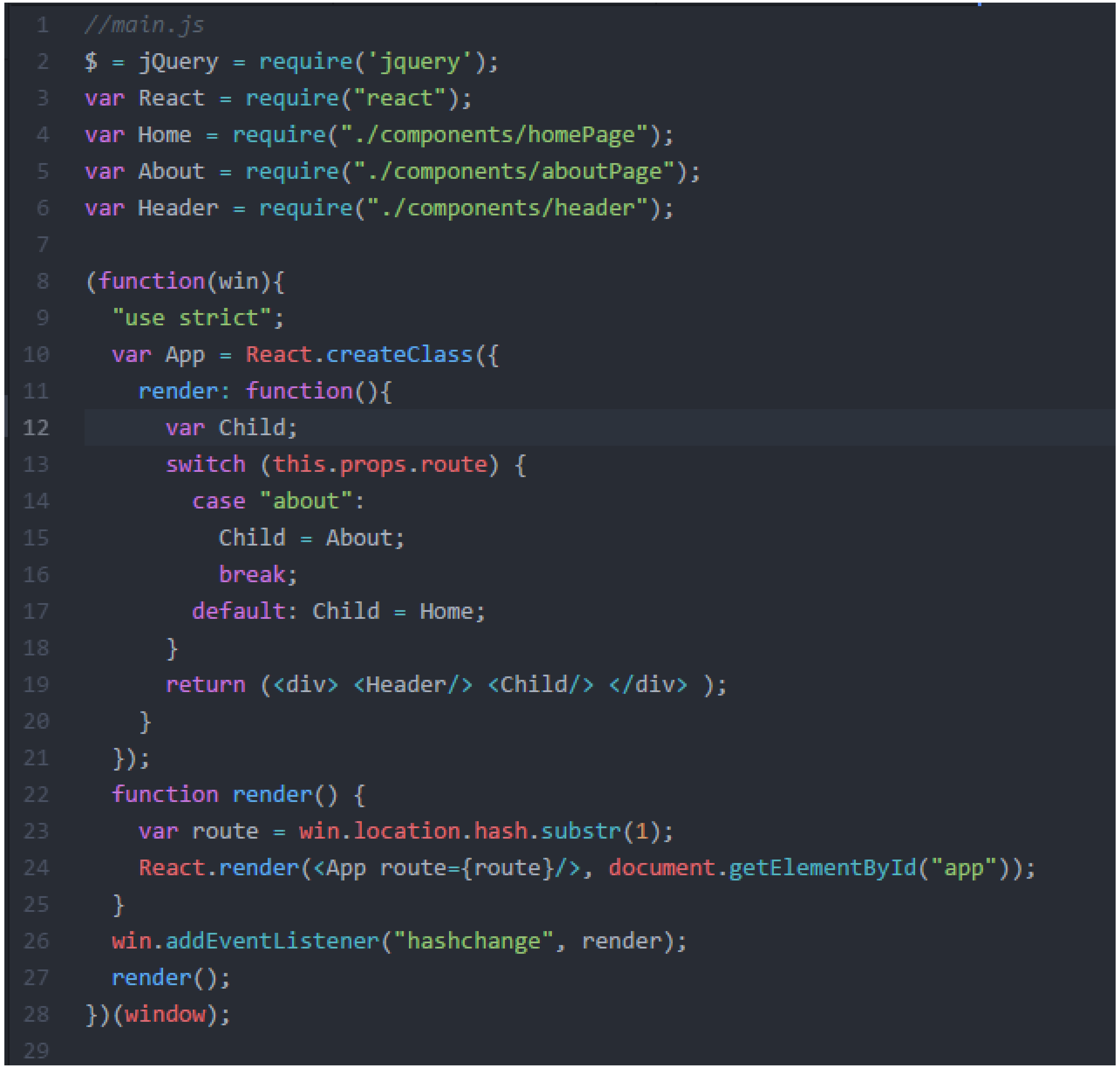


```
//main.js
$ = jQuery = require('jquery');

var React = require("react");
var Home = require("./components/homePage");

React.render(<Home/>, document.getElementById("app"));
```

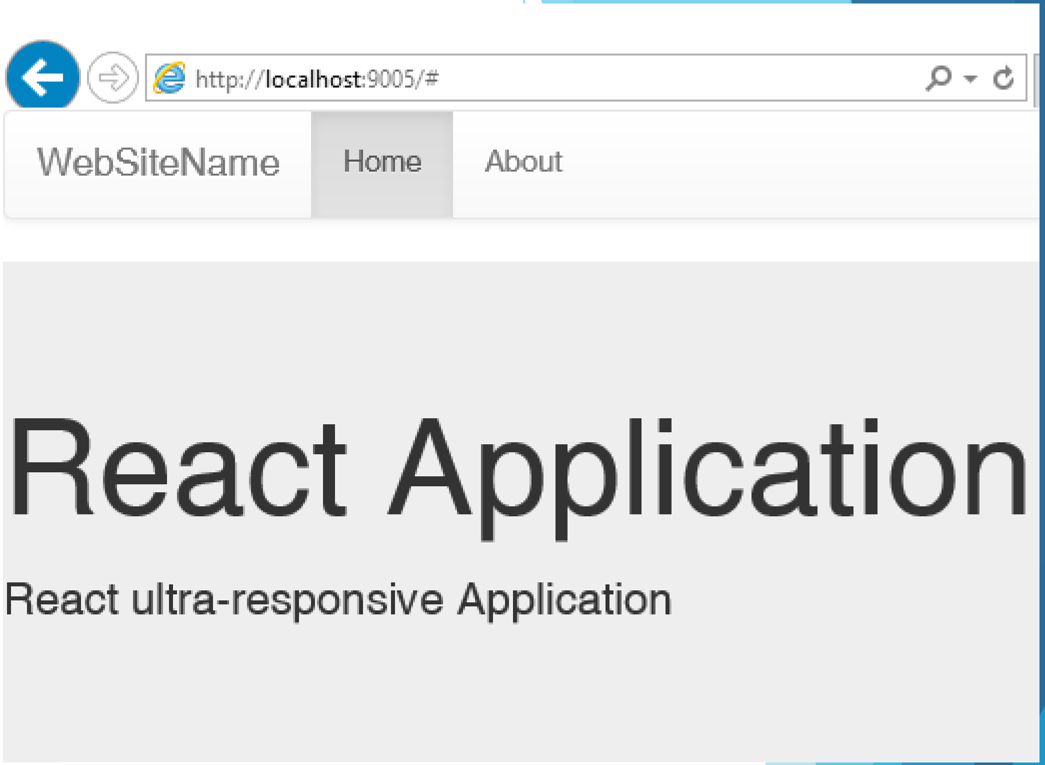


```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Pluralsight Administration</title>
    <link rel="stylesheet" href="css/bundle.css" />
  </head>
  <body>
    <div id="app"></div>
    <script src="scripts/bundle.js"></script>
  </body>
</html>
```

Basic Routing

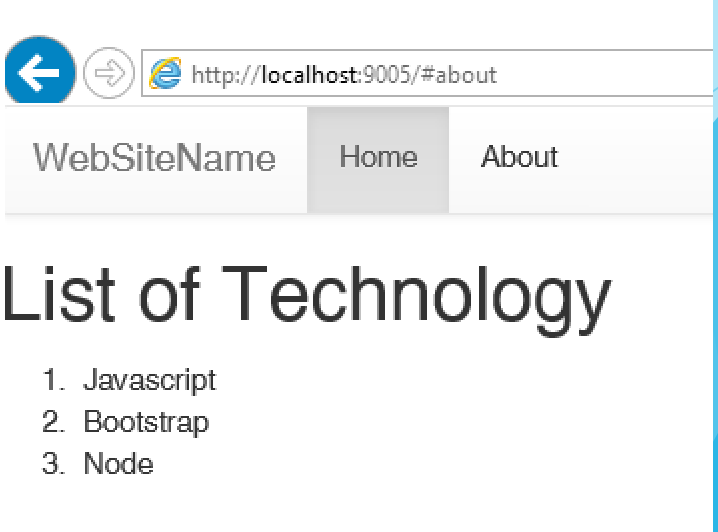


```
//main.js
$ = jQuery = require('jquery');
var React = require("react");
var Home = require("./components/homePage");
var About = require("./components/aboutPage");
var Header = require("./components/header");

(function(win){
  "use strict";
  var App = React.createClass({
    render: function(){
      var Child;
      switch (this.props.route) {
        case "about":
          Child = About;
          break;
        default: Child = Home;
      }
      return (<div> <Header/> <Child/> </div> );
    }
  });
  function render() {
    var route = win.location.hash.substr(1);
    React.render(<App route={route}/>, document.getElementById("app"));
  }
  win.addEventListener("hashchange", render);
  render();
})(window);

```



React Application

React ultra-responsive Application



List of Technology

1. Javascript
2. Bootstrap
3. Node

Props State Key

Props →Immutable, To pass data to child

getDefaultProps

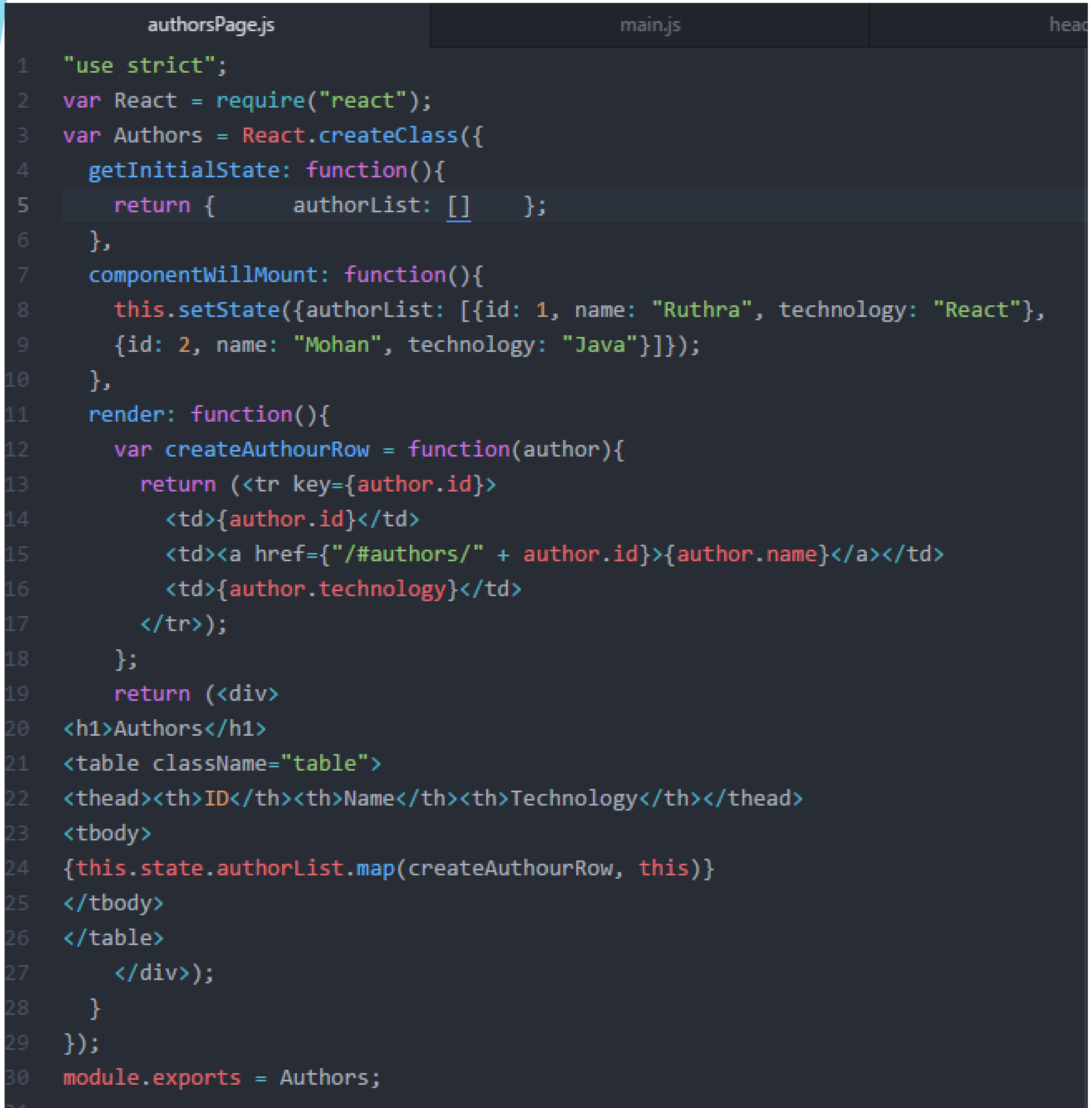
State →Mutable, To Pass data from controller to view

getInitialState

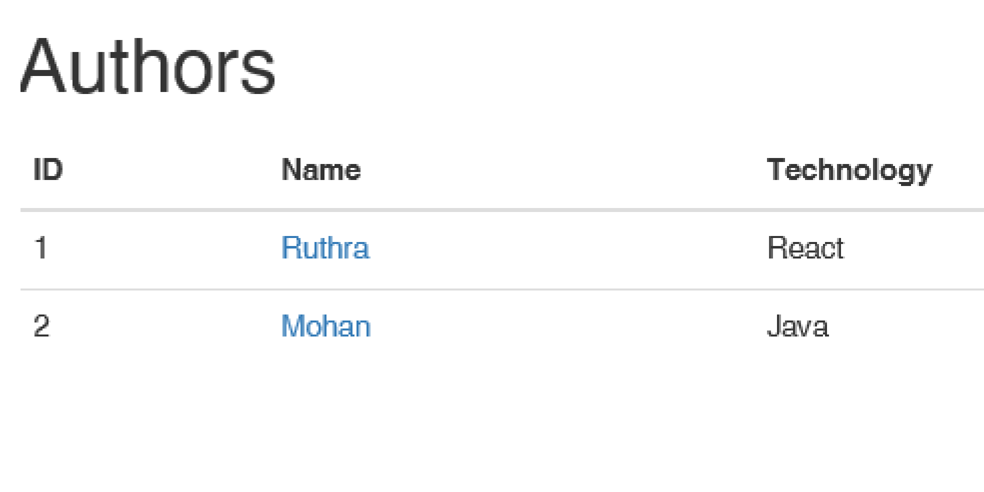
Key → <div key={unique key}></div>

Lifecycle Methods

- componentWillMount
- componentDidMount
- componentWillReceiveProps
- shouldComponentUpdate
- componentWillUpdate
- componentDidUpdate
- componentWillUnmount



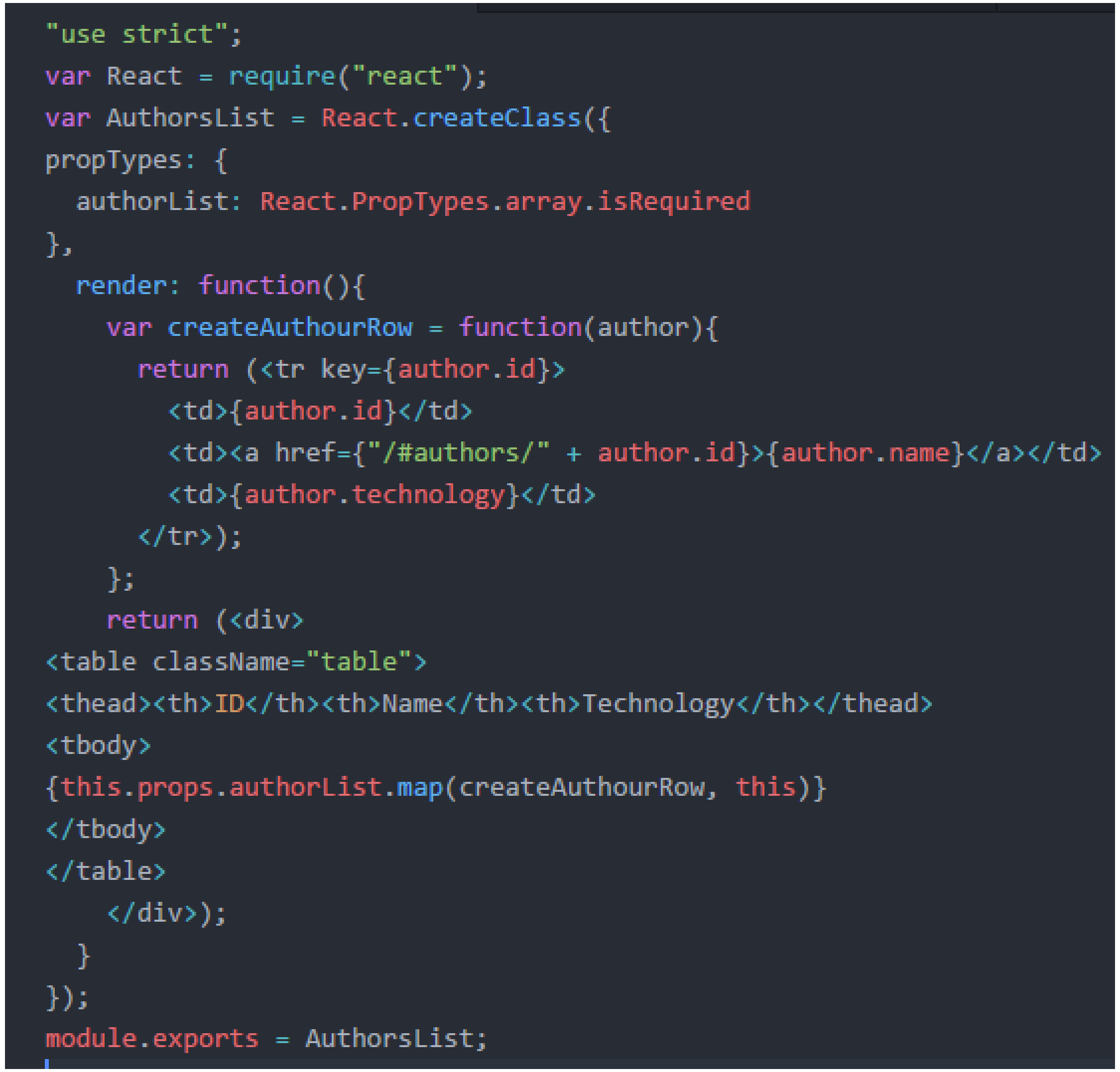
```
"use strict";
var React = require("react");
var Authors = React.createClass({
  getInitialState: function(){
    return {      authorList: []    };
  },
  componentWillMount: function(){
    this.setState({authorList: [{id: 1, name: "Ruthra", technology: "React"},
    {id: 2, name: "Mohan", technology: "Java"}]});
  },
  render: function(){
    var createAuthourRow = function(author){
      return (<tr key={author.id}>
        <td>{author.id}</td>
        <td><a href={"/#authors/" + author.id}>{author.name}</a></td>
        <td>{author.technology}</td>
      </tr>);
    };
    return (<div>
<h1>Authors</h1>
<table className="table">
<thead><th>ID</th><th>Name</th><th>Technology</th></thead>
<tbody>
{this.state.authorList.map(createAuthourRow, this)}
</tbody>
</table>
    </div>);
  }
});
module.exports = Authors;
```



Authors

ID	Name	Technology
1	Ruthra	React
2	Mohan	Java

Props Validation

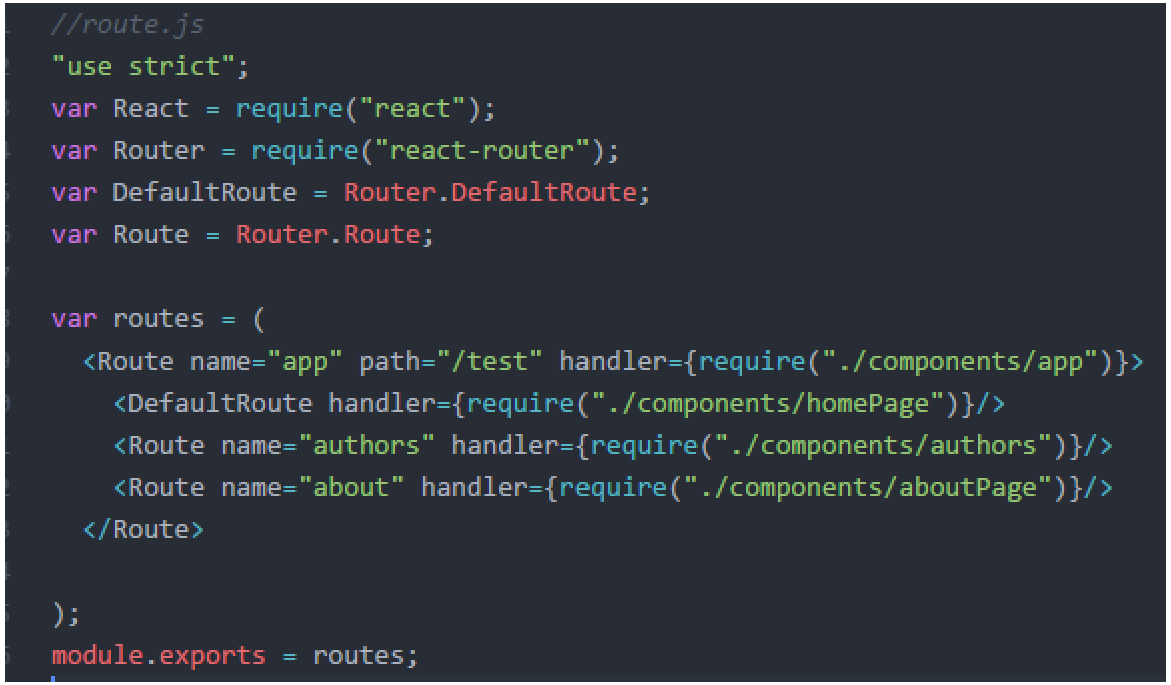


```
"use strict";
var React = require("react");
var AuthorsList = React.createClass({
propTypes: {
  authorList: React.PropTypes.array.isRequired
},
  render: function(){
    var createAuthourRow = function(author){
      return (<tr key={author.id}>
        <td>{author.id}</td>
        <td><a href={"/#authors/" + author.id}>{author.name}</a></td>
        <td>{author.technology}</td>
      </tr>);
    };
    return (<div>
<table className="table">
<thead><th>ID</th><th>Name</th><th>Technology</th></thead>
<tbody>
{this.props.authorList.map(createAuthourRow, this)}
</tbody>
</table>
    </div>);
  }
});
module.exports = AuthorsList;
```

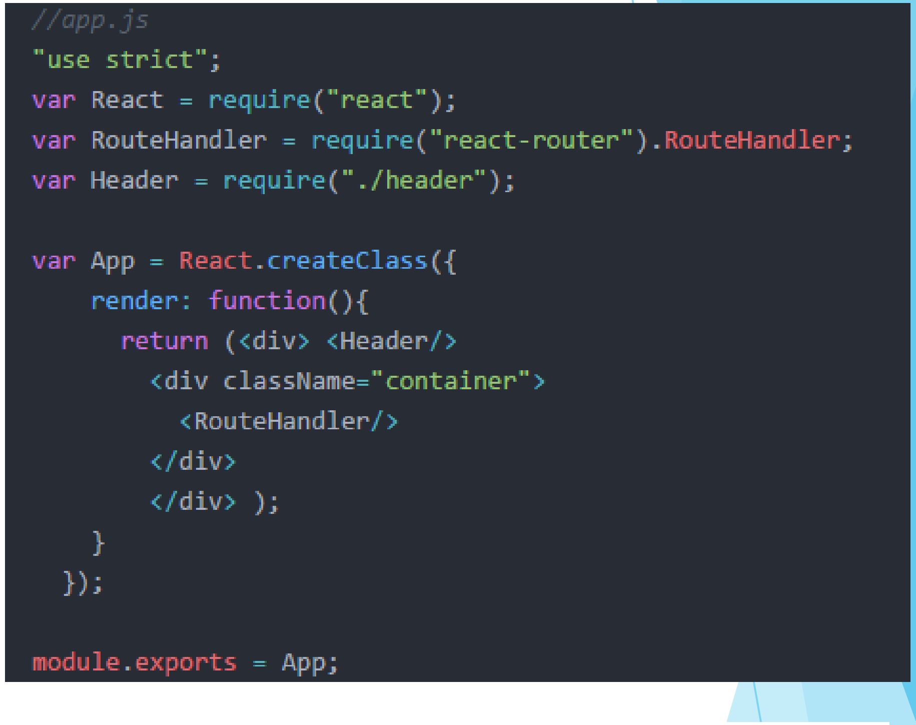
Route

- Route
- DefaultRoute
- NotFoundRoute
- Redirect
- Params (Passing data to targeted state)
- Link
- Transition

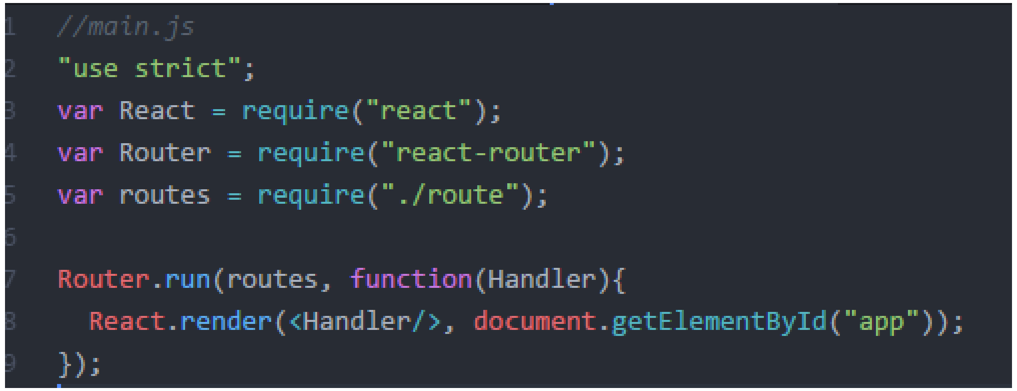
Route



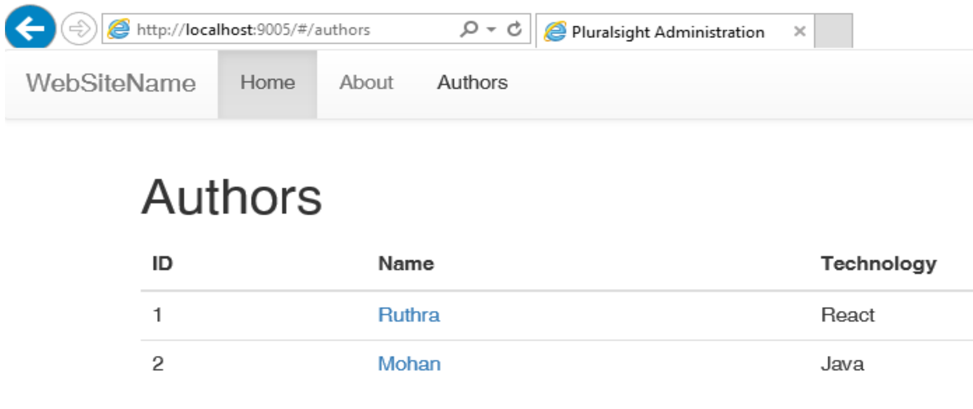
```
//route.js
"use strict";
var React = require("react");
var Router = require("react-router");
var DefaultRoute = Router.DefaultRoute;
var Route = Router.Route;

var routes = (
  <Route name="app" path="/test" handler={require("./components/app")}>
    <DefaultRoute handler={require("./components/homePage")}/>
    <Route name="authors" handler={require("./components/authors")}/>
    <Route name="about" handler={require("./components/aboutPage")}/>
  </Route>

);
module.exports = routes;
```



```
//app.js
"use strict";
var React = require("react");
var RouteHandler = require("react-router").RouteHandler;
var Header = require("./header");

var App = React.createClass({
    render: function(){
      return (<div> <Header/>
        <div className="container">
          <RouteHandler/>
        </div>
        </div> );
    }
  });

module.exports = App;
```



```
//main.js
"use strict";
var React = require("react");
var Router = require("react-router");
var routes = require("./route");

Router.run(routes, function(Handler){
  React.render(<Handler/>, document.getElementById("app"));
});
```



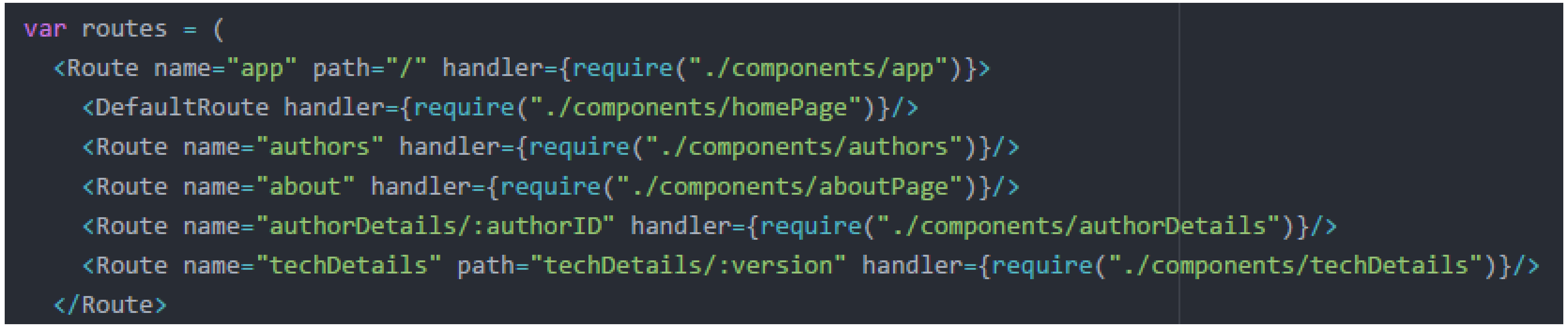
http://localhost:9005/#/authors

Pluralsight Administration

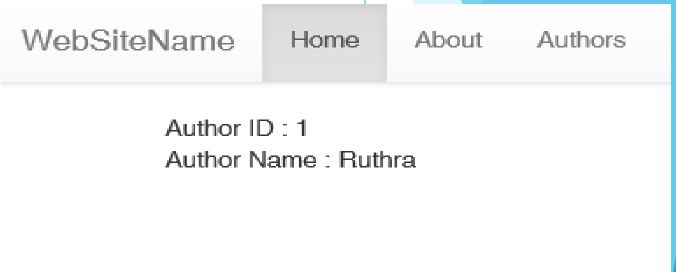
Authors

ID	Name	Technology
1	Ruthra	React
2	Mohan	Java

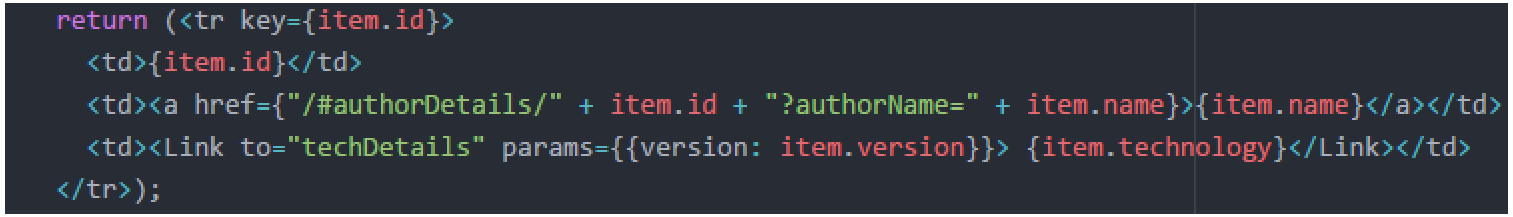
Query / Path Params / Link



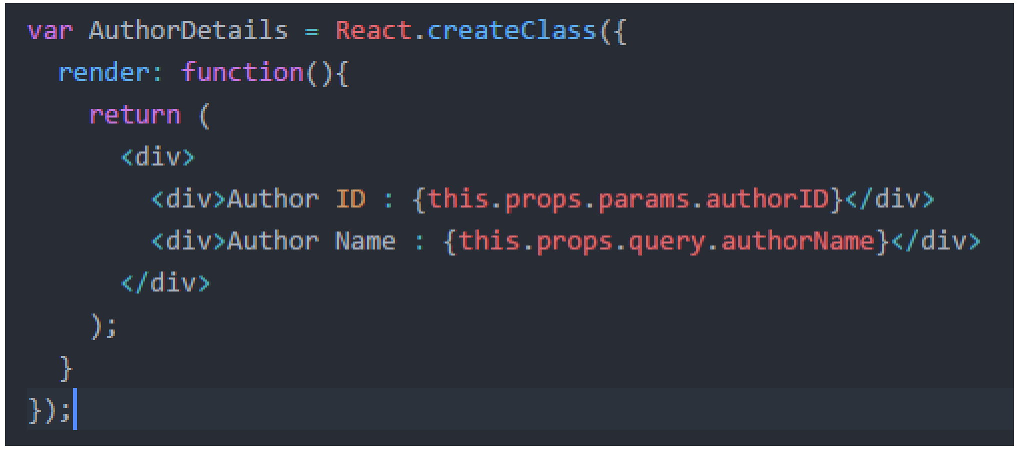
```
var routes = (
  <Route name="app" path="/" handler={require("./components/app")}>
    <DefaultRoute handler={require("./components/homePage")}/>
    <Route name="authors" handler={require("./components/authors")}/>
    <Route name="about" handler={require("./components/aboutPage")}/>
    <Route name="authorDetails/:authorID" handler={require("./components/authorDetails")}/>
    <Route name="techDetails" path="techDetails/:version" handler={require("./components/techDetails")}/>
  </Route>
```



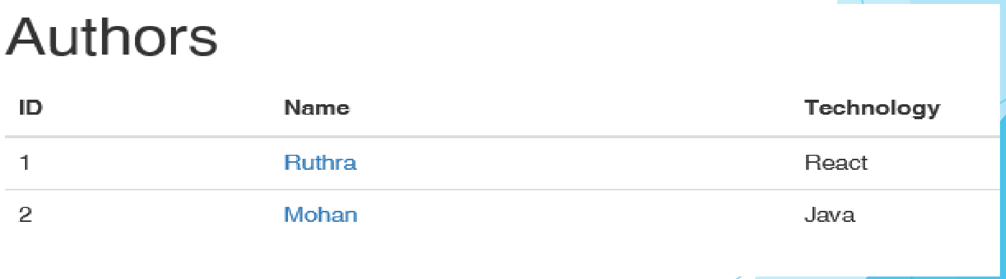
Author ID : 1
Author Name : Ruthra



```
  return (<tr key={item.id}>
    <td>{item.id}</td>
    <td><a href={"/#authorDetails/" + item.id + "?authorName=" + item.name}>{item.name}</a></td>
    <td><Link to="techDetails" params={{version: item.version}}> {item.technology}</Link></td>
  </tr>);
```



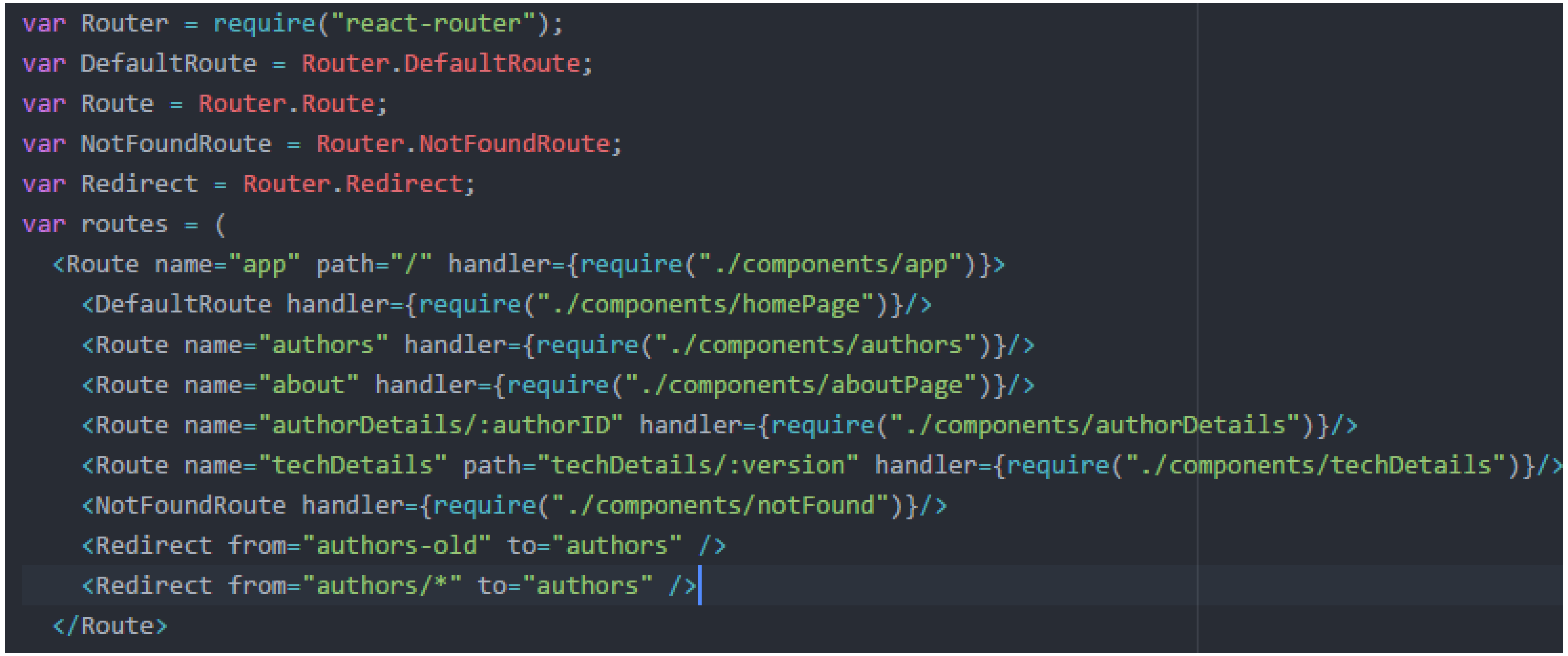
```
var AuthorDetails = React.createClass({
  render: function(){
    return (
      <div>
        <div>Author ID : {this.props.params.authorID}</div>
        <div>Author Name : {this.props.query.authorName}</div>
      </div>
    );
  }
});
```



Authors

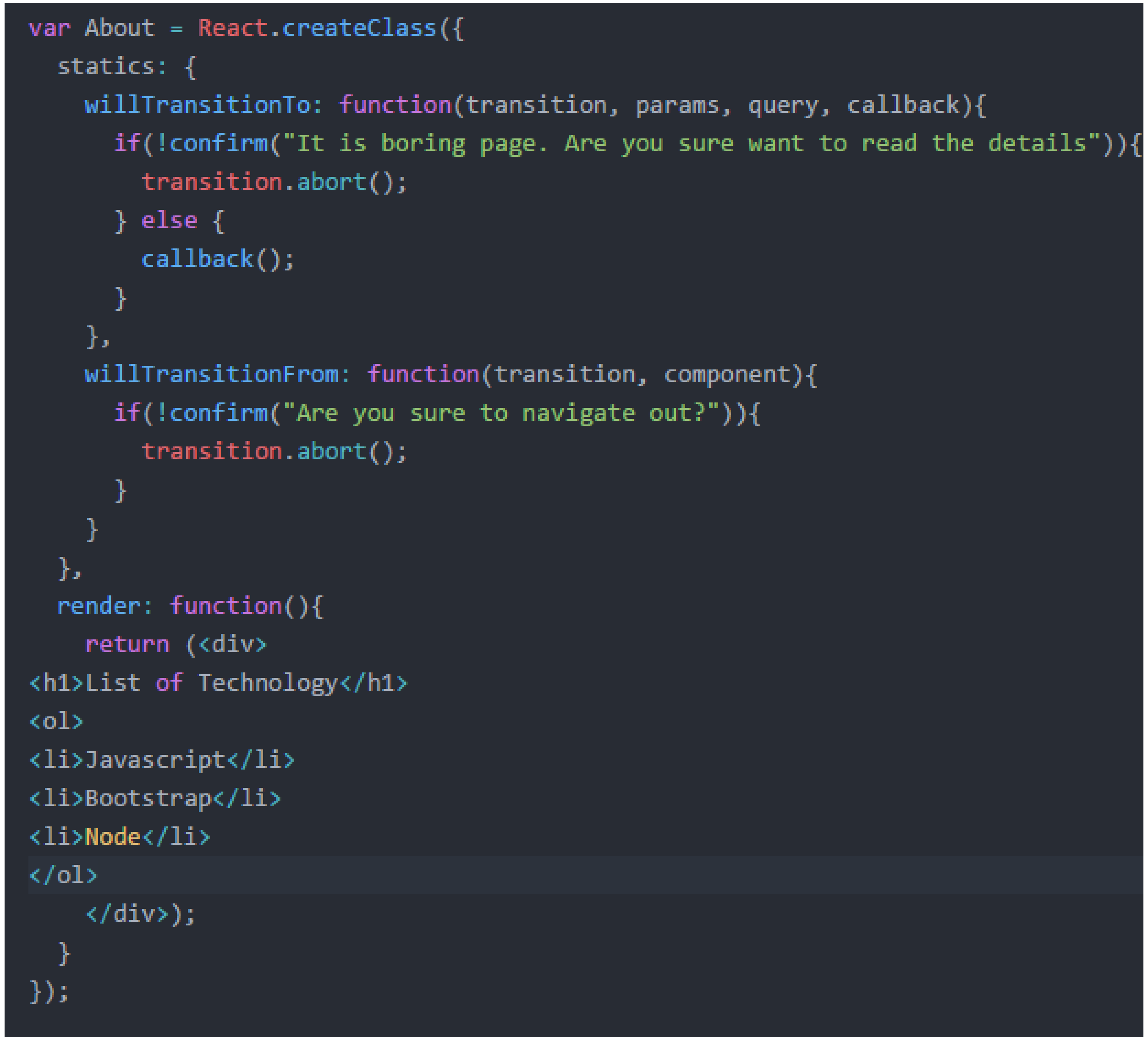
ID	Name	Technology
1	Ruthra	React
2	Mohan	Java

NotFound, Redirect



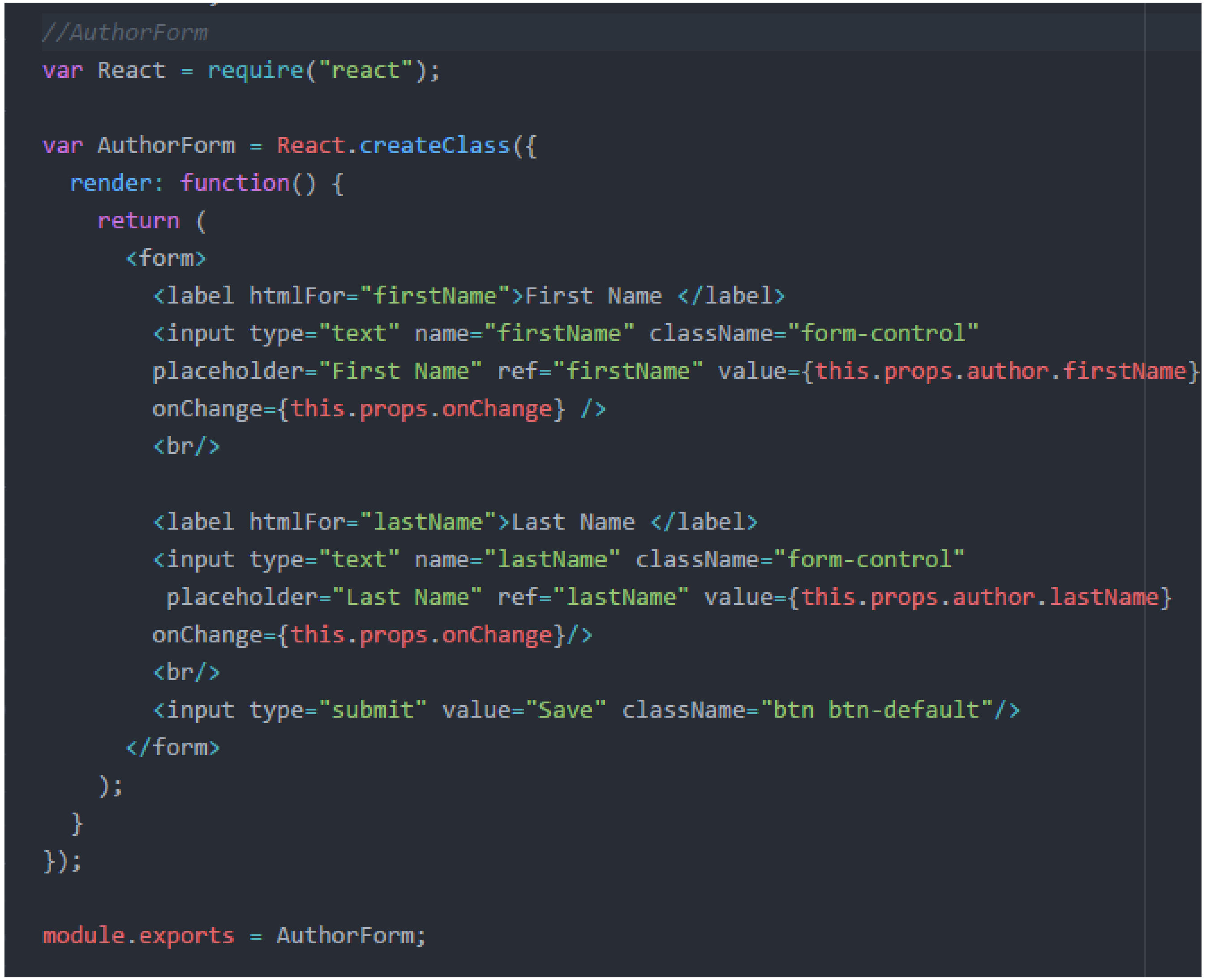
```
var Router = require("react-router");
var DefaultRoute = Router.DefaultRoute;
var Route = Router.Route;
var NotFoundRoute = Router.NotFoundRoute;
var Redirect = Router.Redirect;
var routes = (
  <Route name="app" path="/" handler={require("./components/app")}>
    <DefaultRoute handler={require("./components/homePage")}/>
    <Route name="authors" handler={require("./components/authors")}/>
    <Route name="about" handler={require("./components/aboutPage")}/>
    <Route name="authorDetails/:authorID" handler={require("./components/authorDetails")}/>
    <Route name="techDetails" path="techDetails/:version" handler={require("./components/techDetails")}/>
    <NotFoundRoute handler={require("./components/notFound")}/>
    <Redirect from="authors-old" to="authors" />
    <Redirect from="authors/*" to="authors" />
  </Route>
```

Transition

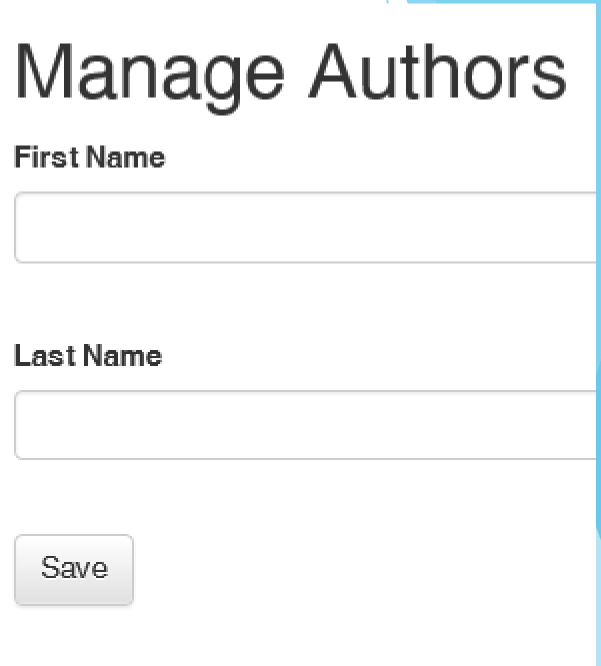


```
var About = React.createClass({
  statics: {
    willTransitionTo: function(transition, params, query, callback){
      if(!confirm("It is boring page. Are you sure want to read the details")){
        transition.abort();
      } else {
        callback();
      }
    },
    willTransitionFrom: function(transition, component){
      if(!confirm("Are you sure to navigate out?")){
        transition.abort();
      }
    }
  },
  render: function(){
    return (<div>
<h1>List of Technology</h1>
<ol>
<li>Javascript</li>
<li>Bootstrap</li>
<li>Node</li>
</ol>
    </div>);
  }
});
```

Form



```
//AuthorForm
var React = require("react");

var AuthorForm = React.createClass({
  render: function() {
    return (
      <form>
        <label htmlFor="firstName">First Name </label>
        <input type="text" name="firstName" className="form-control"
        placeholder="First Name" ref="firstName" value={this.props.author.firstName}
        onChange={this.props.onChange} />
        <br/>

        <label htmlFor="lastName">Last Name </label>
        <input type="text" name="lastName" className="form-control"
         placeholder="Last Name" ref="lastName" value={this.props.author.lastName}
        onChange={this.props.onChange}/>
        <br/>
        <input type="submit" value="Save" className="btn btn-default"/>
      </form>
    );
  }
});

module.exports = AuthorForm;
```

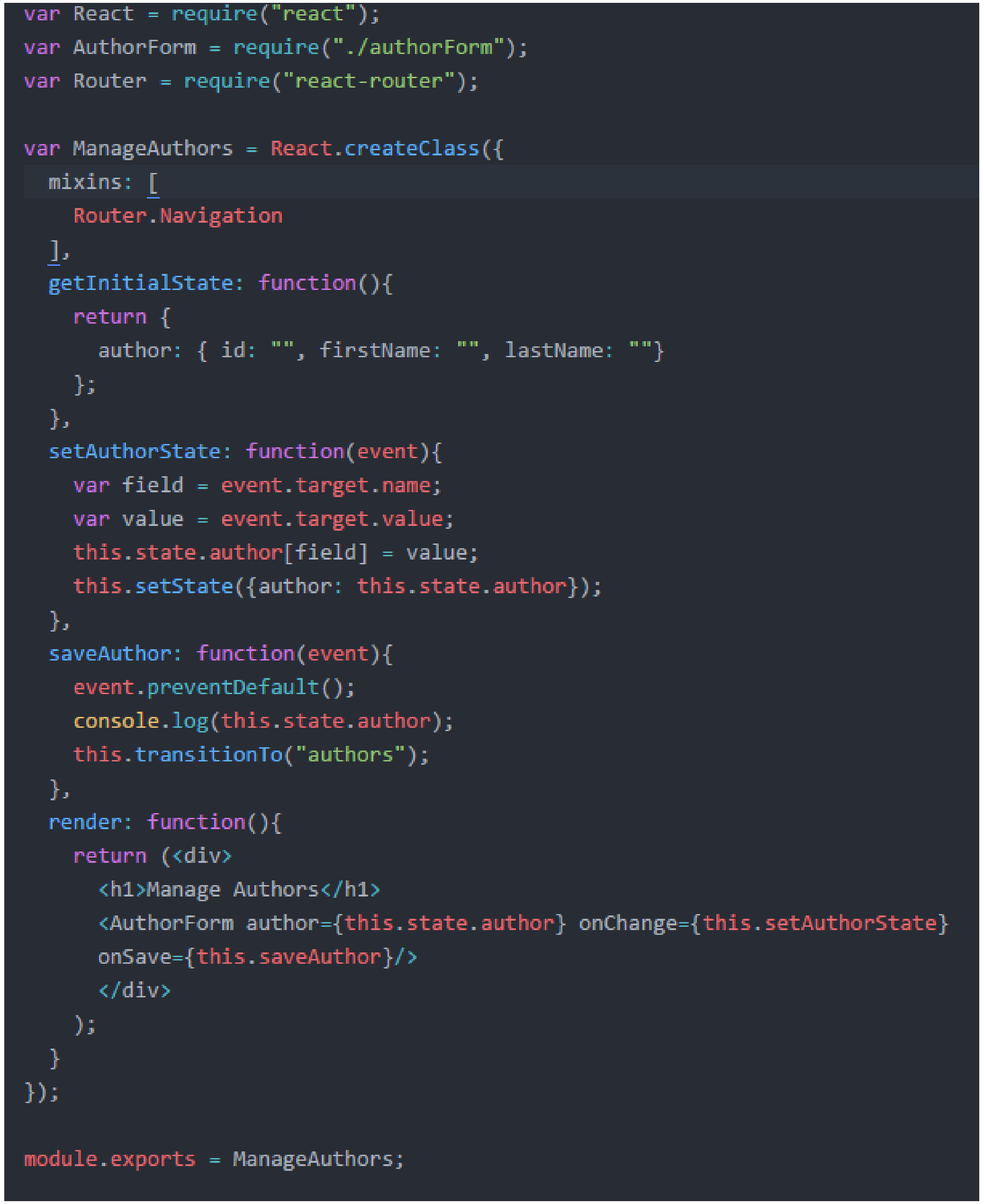


Manage Authors

First Name

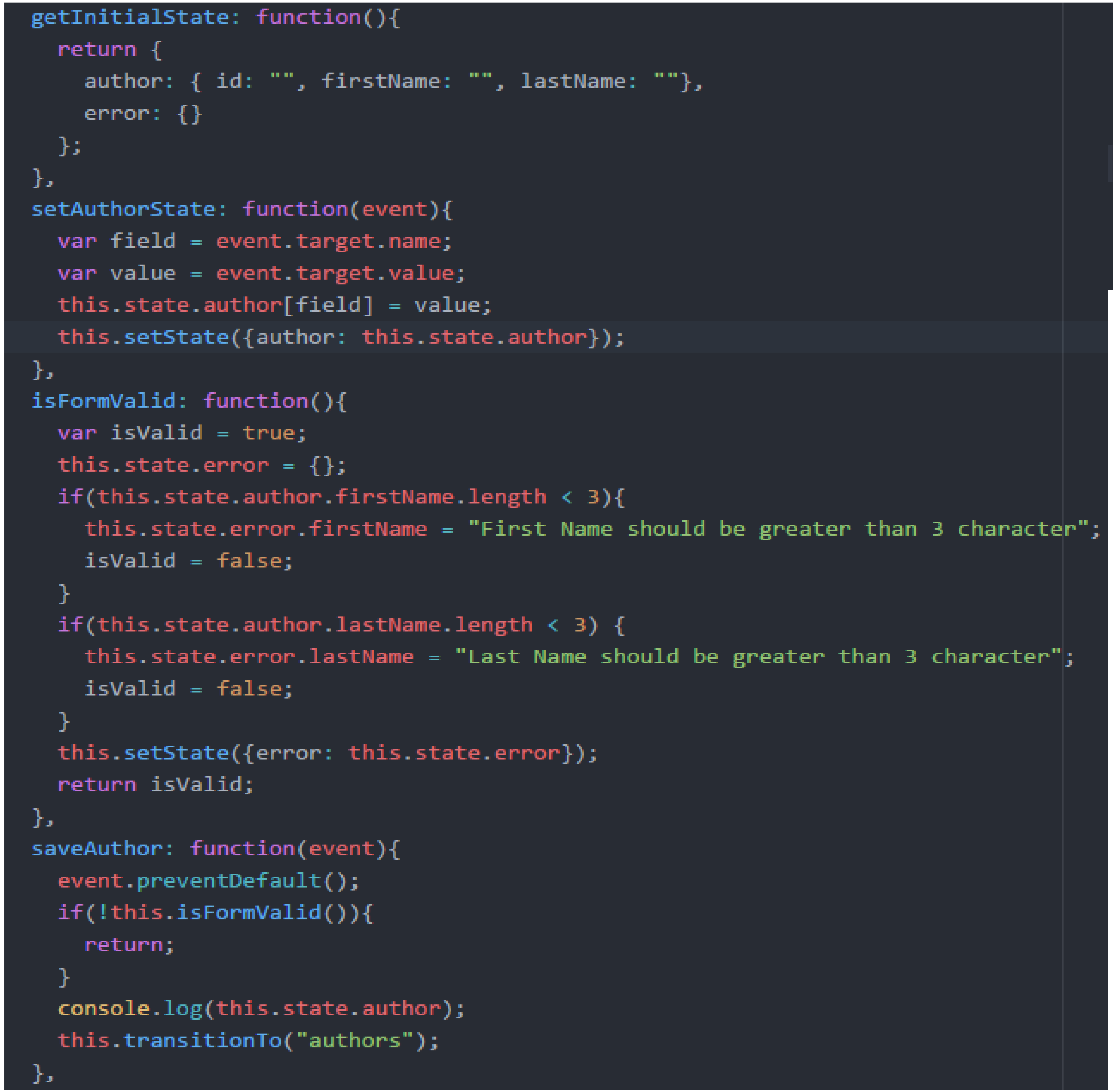
Last Name

Consume Form

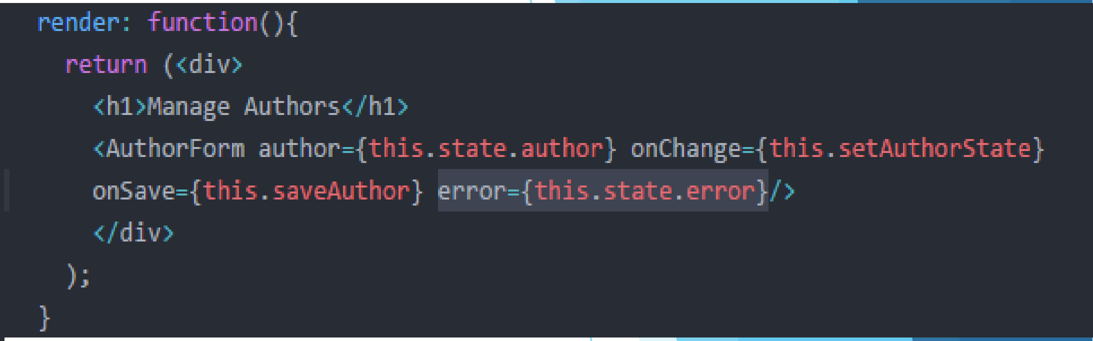


```
var React = require("react");
var AuthorForm = require("./authorForm");
var Router = require("react-router");

var ManageAuthors = React.createClass({
  mixins: [
    Router.Navigation
  ],
  getInitialState: function(){
    return {
      author: { id: "", firstName: "", lastName: ""}
    };
  },
  setAuthorState: function(event){
    var field = event.target.name;
    var value = event.target.value;
    this.state.author[field] = value;
    this.setState({author: this.state.author});
  },
  saveAuthor: function(event){
    event.preventDefault();
    console.log(this.state.author);
    this.transitionTo("authors");
  },
  render: function(){
    return (<div>
      <h1>Manage Authors</h1>
      <AuthorForm author={this.state.author} onChange={this.setAuthorState}
      onSave={this.saveAuthor}/>
      </div>
    );
  }
});

module.exports = ManageAuthors;
```

Form Validation

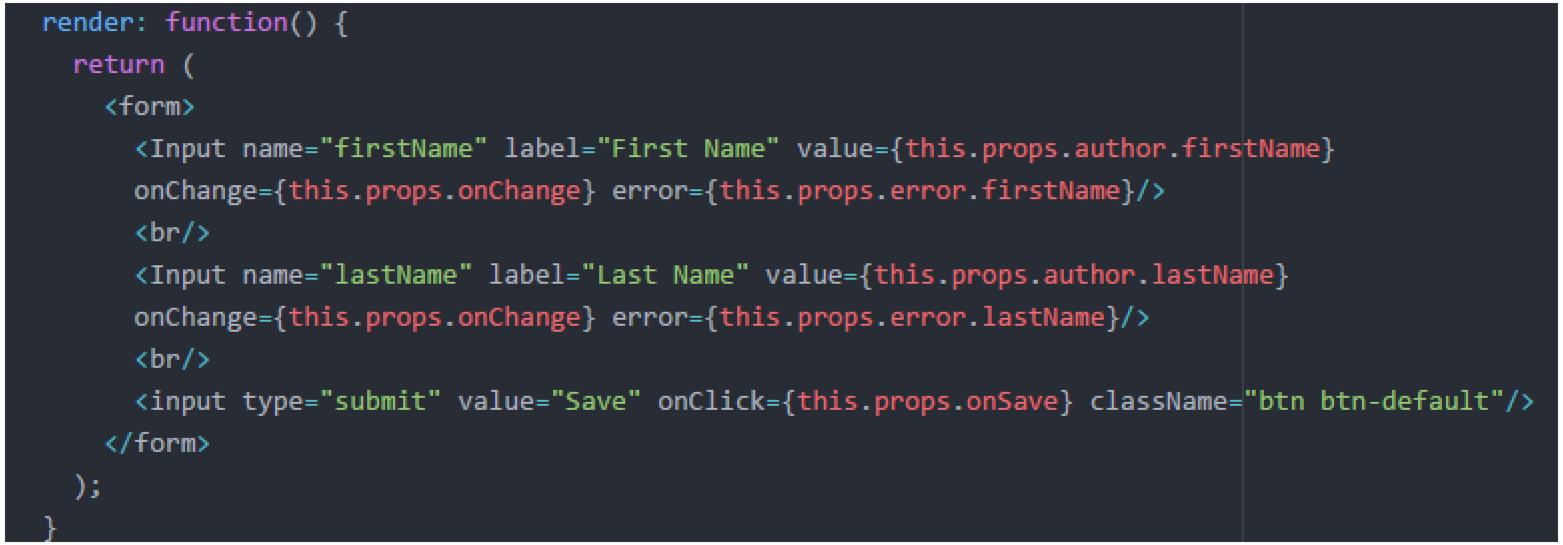


```
getInitialState: function(){
  return {
    author: { id: "", firstName: "", lastName: ""},
    error: {}
  };
},
setAuthorState: function(event){
  var field = event.target.name;
  var value = event.target.value;
  this.state.author[field] = value;
  this.setState({author: this.state.author});
},
isFormValid: function(){
  var isValid = true;
  this.state.error = {};
  if(this.state.author.firstName.length < 3){
    this.state.error.firstName = "First Name should be greater than 3 character";
    isValid = false;
  }
  if(this.state.author.lastName.length < 3) {
    this.state.error.lastName = "Last Name should be greater than 3 character";
    isValid = false;
  }
  this.setState({error: this.state.error});
  return isValid;
},
saveAuthor: function(event){
  event.preventDefault();
  if(!this.isFormValid()){
    return;
  }
  console.log(this.state.author);
  this.transitionTo("authors");
},
```

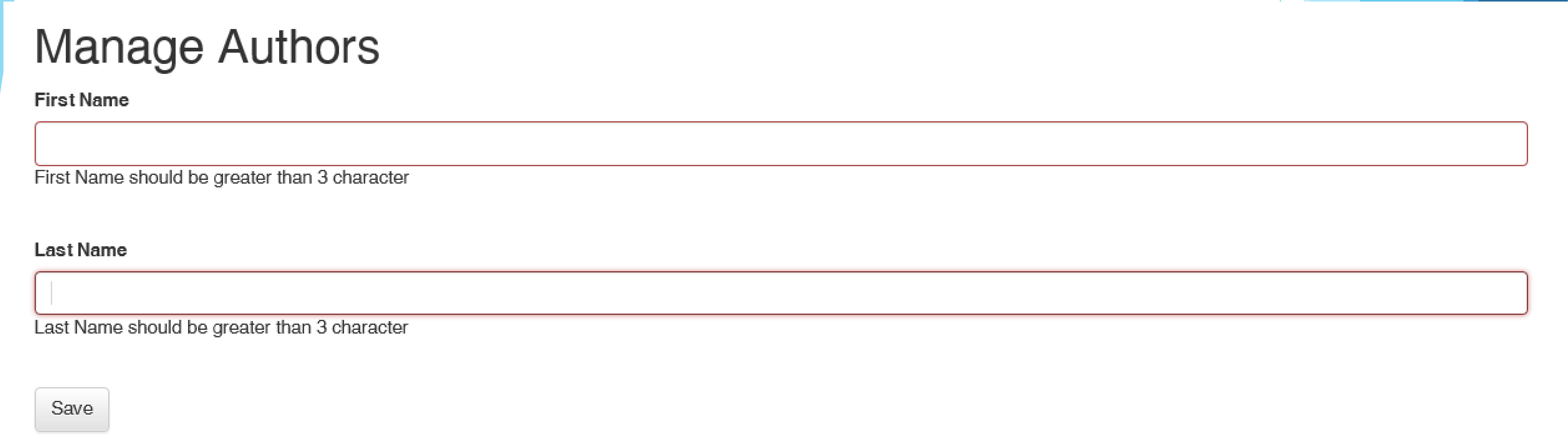


```
render: function(){
  return (<div>
    <h1>Manage Authors</h1>
    <AuthorForm author={this.state.author} onChange={this.setAuthorState}
    onSave={this.saveAuthor} error={this.state.error}/>
    </div>
  );
}
```

Form Validation



```
render: function() {
  return (
    <form>
      <Input name="firstName" label="First Name" value={this.props.author.firstName}
      onChange={this.props.onChange} error={this.props.error.firstName}/>
      <br/>
      <Input name="lastName" label="Last Name" value={this.props.author.lastName}
      onChange={this.props.onChange} error={this.props.error.lastName}/>
      <br/>
      <input type="submit" value="Save" onClick={this.props.onSave} className="btn btn-default"/>
    </form>
  );
}
```



Manage Authors

First Name

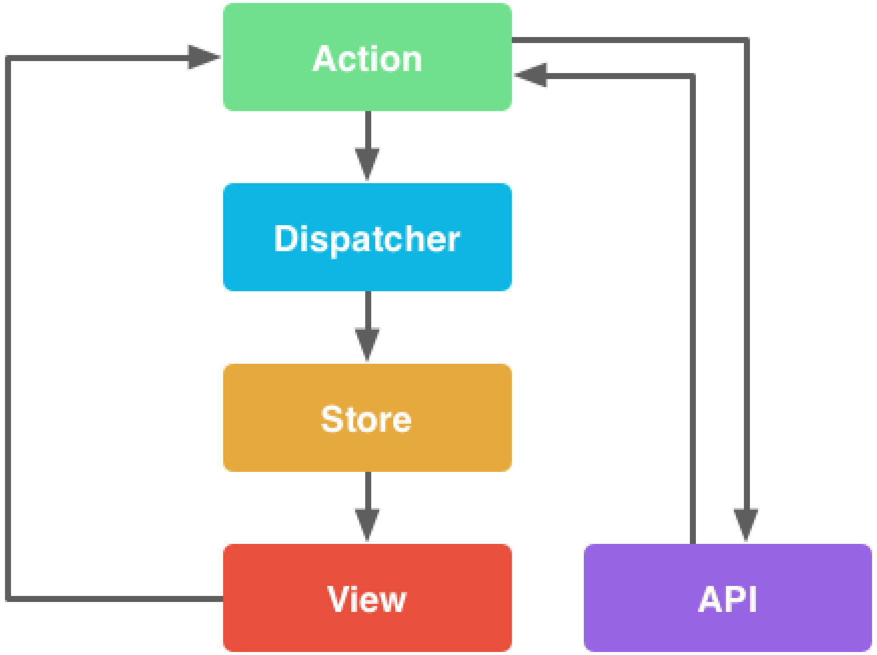
First Name should be greater than 3 character

Last Name

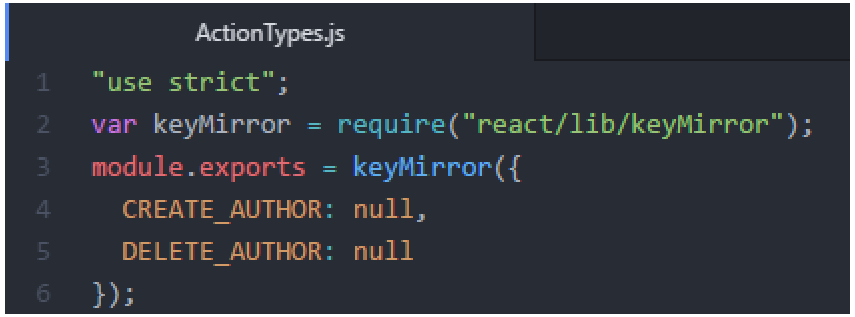
Last Name should be greater than 3 character

Flux

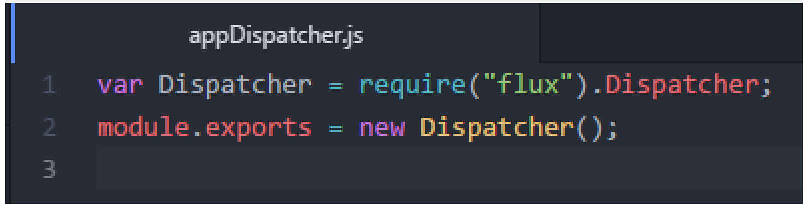
Design pattern for Unidirectional data flow



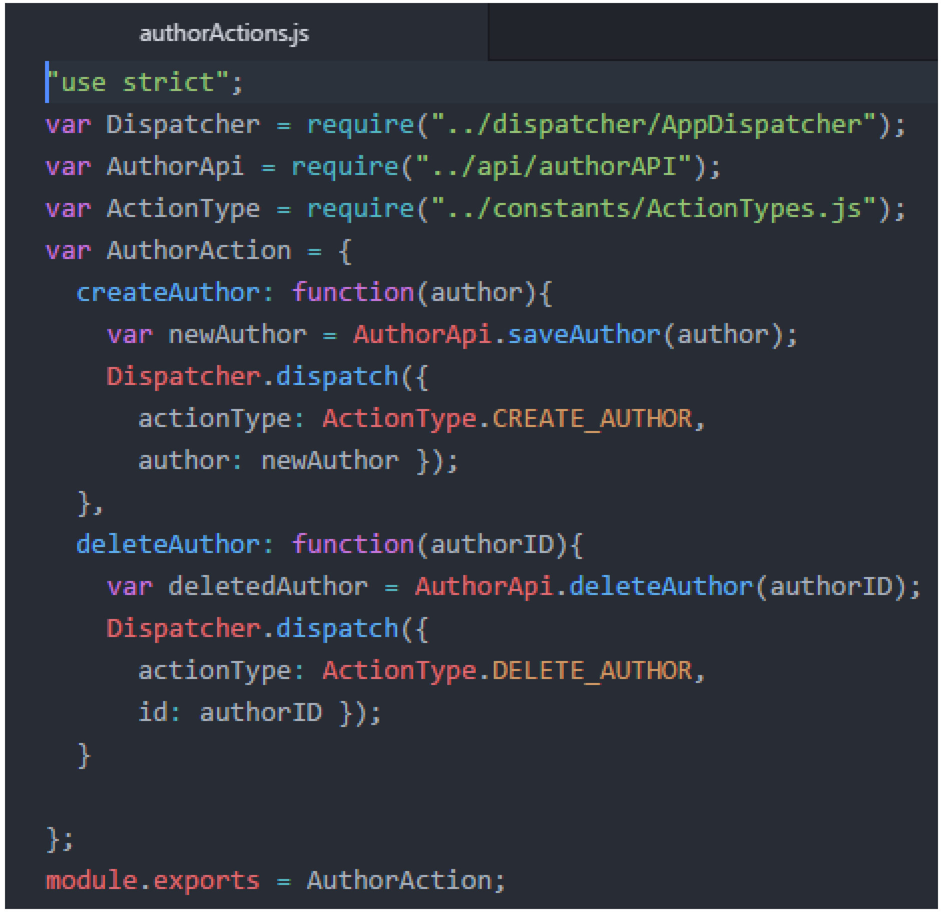
Flux(Constants,Dispatcher, Action)



```
"use strict";
var keyMirror = require("react/lib/keyMirror");
module.exports = keyMirror({
  CREATE_AUTHOR: null,
  DELETE_AUTHOR: null
});
```

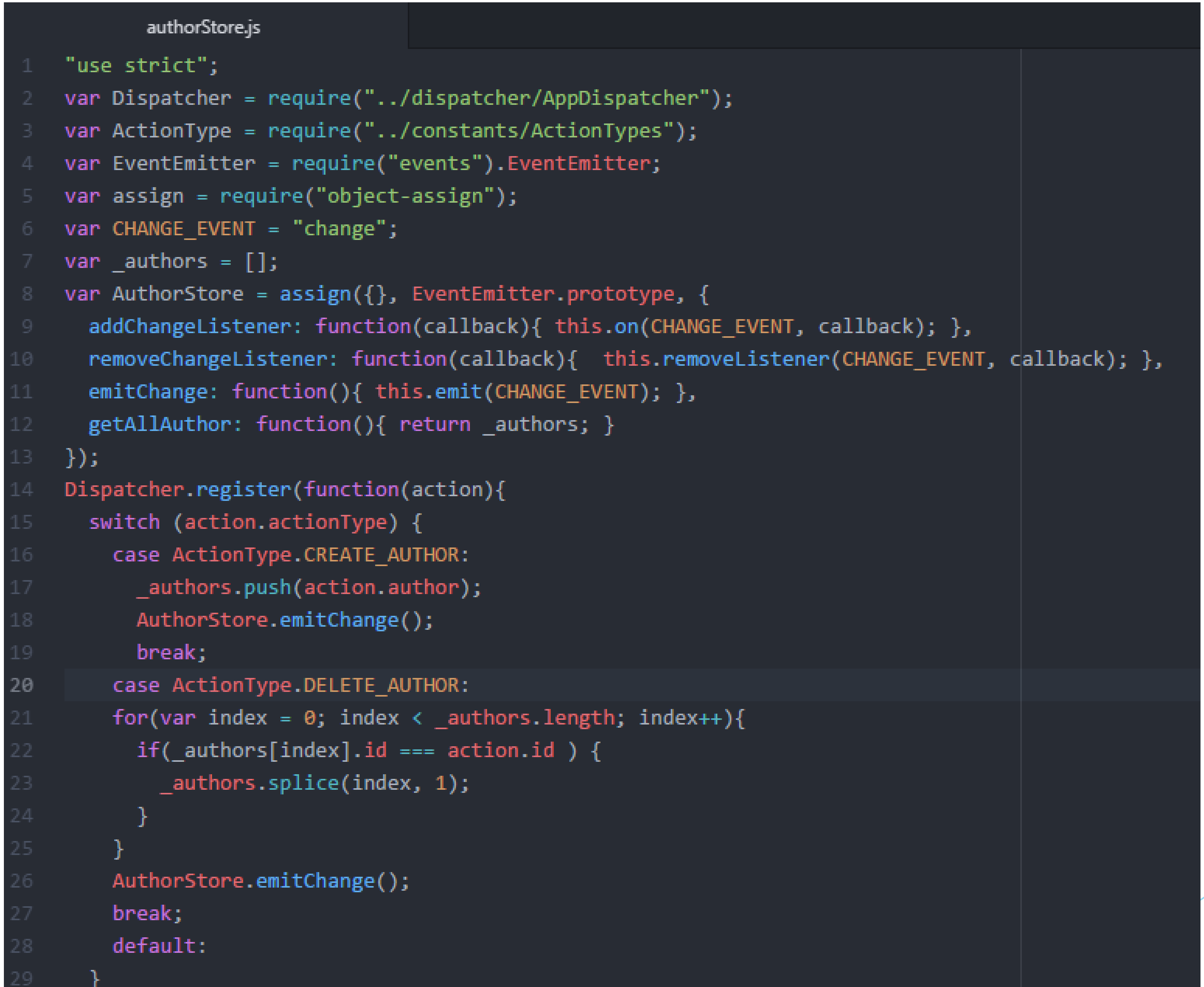


```
var Dispatcher = require("flux").Dispatcher;
module.exports = new Dispatcher();
```



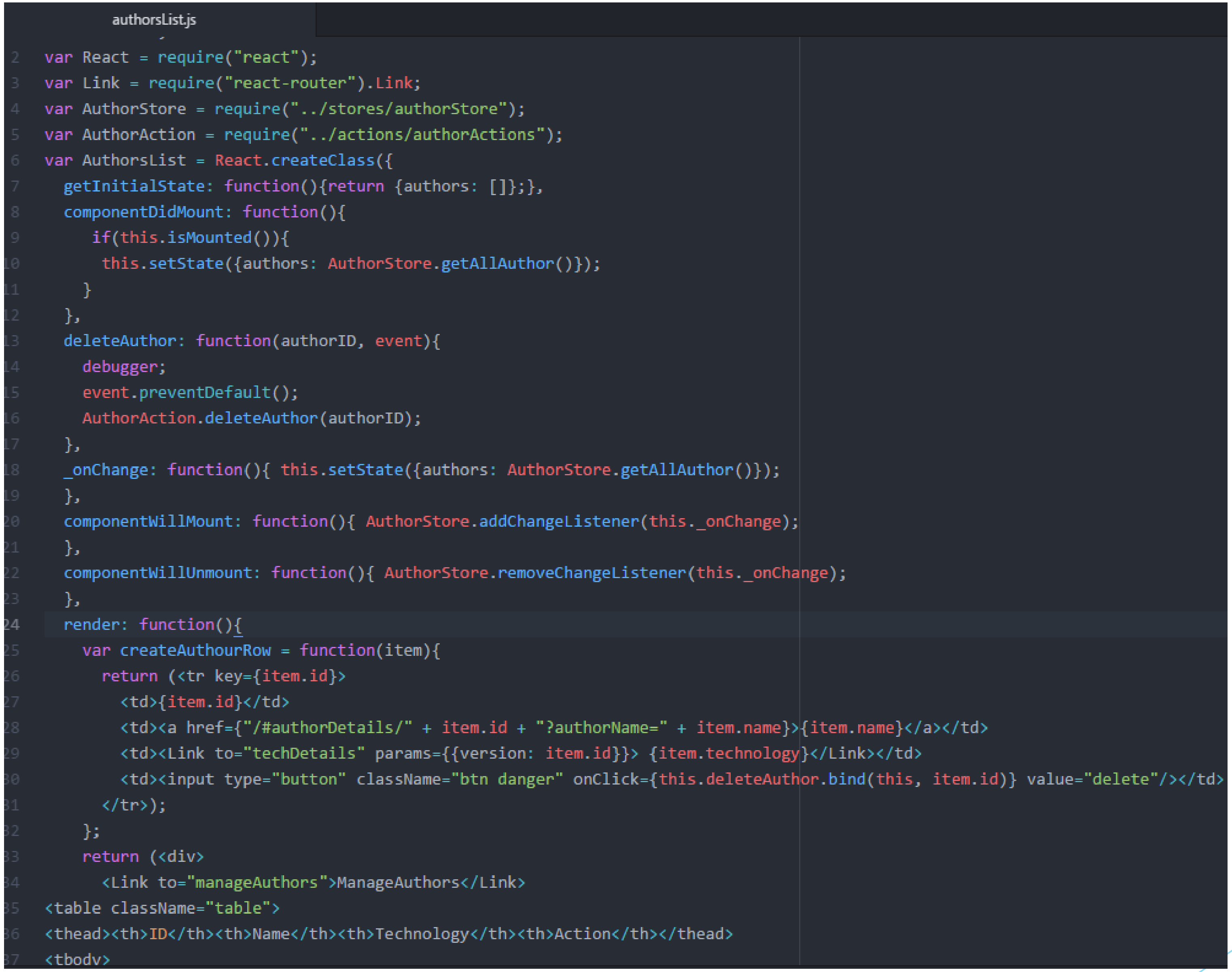
```
"use strict";
var Dispatcher = require("../dispatcher/AppDispatcher");
var AuthorApi = require("../api/authorAPI");
var ActionType = require("../constants/ActionTypes.js");
var AuthorAction = {
  createAuthor: function(author){
    var newAuthor = AuthorApi.saveAuthor(author);
    Dispatcher.dispatch({
      actionType: ActionType.CREATE_AUTHOR,
      author: newAuthor });
  },
  deleteAuthor: function(authorID){
    var deletedAuthor = AuthorApi.deleteAuthor(authorID);
    Dispatcher.dispatch({
      actionType: ActionType.DELETE_AUTHOR,
      id: authorID });
  }

};
module.exports = AuthorAction;
```

Store



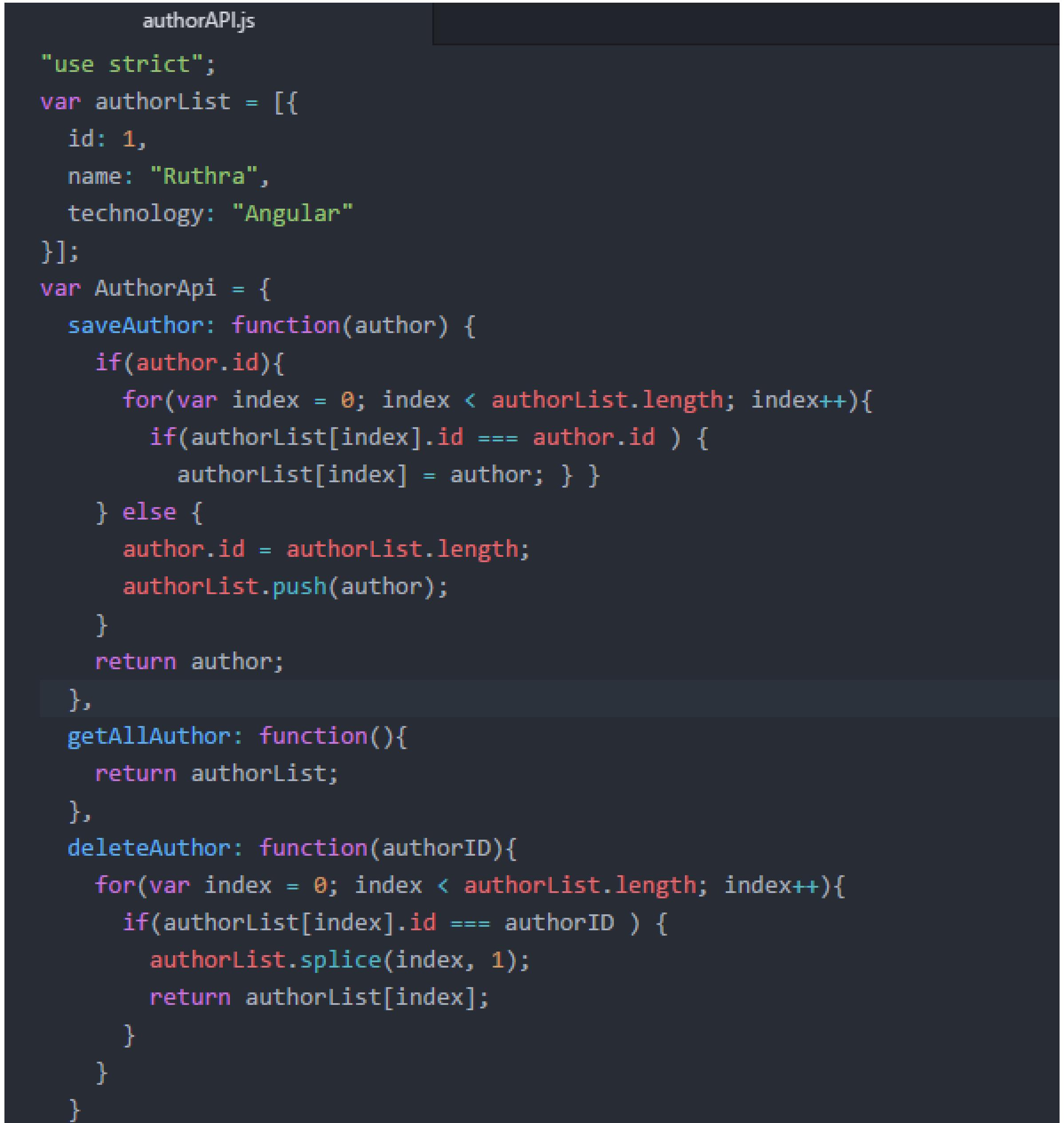
```javascript
"use strict";
var Dispatcher = require("../dispatcher/AppDispatcher");
var ActionType = require("../constants/ActionTypes");
var EventEmitter = require("events").EventEmitter;
var assign = require("object-assign");
var CHANGE_EVENT = "change";
var _authors = [];
var AuthorStore = assign({}, EventEmitter.prototype, {
  addChangeListener: function(callback){ this.on(CHANGE_EVENT, callback); },
  removeChangeListener: function(callback){  this.removeListener(CHANGE_EVENT, callback); },
  emitChange: function(){ this.emit(CHANGE_EVENT); },
  getAllAuthor: function(){ return _authors; }
});
Dispatcher.register(function(action){
  switch (action.actionType) {
    case ActionType.CREATE_AUTHOR:
      _authors.push(action.author);
      AuthorStore.emitChange();
      break;
    case ActionType.DELETE_AUTHOR:
    for(var index = 0; index < _authors.length; index++){
      if(_authors[index].id === action.id ) {
        _authors.splice(index, 1);
      }
    }
    AuthorStore.emitChange();
    break;
    default:
  }
```

Flux (Component)



```
var React = require("react");
var Link = require("react-router").Link;
var AuthorStore = require("../stores/authorStore");
var AuthorAction = require("../actions/authorActions");
var AuthorsList = React.createClass({
  getInitialState: function(){return {authors: []};},
  componentDidMount: function(){
     if(this.isMounted()){
      this.setState({authors: AuthorStore.getAllAuthor()});
    }
  },
  deleteAuthor: function(authorID, event){
    debugger;
    event.preventDefault();
    AuthorAction.deleteAuthor(authorID);
  },
  _onChange: function(){ this.setState({authors: AuthorStore.getAllAuthor()});
  },
  componentWillMount: function(){ AuthorStore.addChangeListener(this._onChange);
  },
  componentWillUnmount: function(){ AuthorStore.removeChangeListener(this._onChange);
  },
  render: function(){
    var createAuthourRow = function(item){
      return (<tr key={item.id}>
        <td>{item.id}</td>
        <td><a href={"/#authorDetails/" + item.id + "?authorName=" + item.name}>{item.name}</a></td>
        <td><Link to="techDetails" params={{version: item.id}}> {item.technology}</Link></td>
        <td><input type="button" className="btn danger" onClick={this.deleteAuthor.bind(this, item.id)} value="delete"/></td>
      </tr>);
    };
    return (<div>
      <Link to="manageAuthors">ManageAuthors</Link>
<table className="table">
<thead><th>ID</th><th>Name</th><th>Technology</th><th>Action</th></thead>
<tbody>
```

Mock Api



```
"use strict";
var authorList = [{
  id: 1,
  name: "Ruthra",
  technology: "Angular"
}];
var AuthorApi = {
  saveAuthor: function(author) {
    if(author.id){
      for(var index = 0; index < authorList.length; index++){
        if(authorList[index].id === author.id ) {
          authorList[index] = author; } }
    } else {
      author.id = authorList.length;
      authorList.push(author);
    }
    return author;
  },
  getAllAuthor: function(){
    return authorList;
  },
  deleteAuthor: function(authorID){
    for(var index = 0; index < authorList.length; index++){
      if(authorList[index].id === authorID ) {
        authorList.splice(index, 1);
        return authorList[index];
      }
    }
  }
```

Thank you!!!

Ruthran.kumar@gmail.com

+44 7405708980

+91 9715531887