

react

Posted February 22nd, 2024

react is a frontend javascript implementation

1. you can run react direct in a standard html file
https://www.w3schools.com/react/react_getstarted.asp
2. Or create a react.js app
 - npm install -g create-react-app
 - npx create-react-app my-app-name
 - npm start - will start app on port 3000
 - edit package.json to redirect port
under scripts section in package.json
change "start": "PORT=9561 react-scripts start",
 - this is a development version
 - can redirect apache with
ProxyPass / http://localhost:9561/
 - the sample app created expects the app to be at the top level
e.g. react.mindcontract.com
The paths are not set up to allow a subdir app.

react.js cheatsheet

From facebook.

Babel transpiler translates react.js to html/css

Command that puts JSX on DOM
`ReactDOM.render(JSX, document.getElementById('root'))`

When defining a JSX must return a complete element

comments:
`{/* comments here */}`

ReactDOM api renders JSX to HTMLDOM
`ReactDOM.render(componentToRender, targetNode)`

use className instead of class on elements

Components:

1. components start with Capital letter
2. must return element or null
3. stateless

```
const MyComponent = function () {  
  return (  
    <div>foo</div>  
  )  
}
```

4. ES6 format - extend from React.Component
`class MyComponent extends React.Component {`
 `constructor(props) {`
 `super(props);`
 `}`

```
  render() {  
    return (  
      <h1>Hi</h1>  
    );  
  }  
}
```

```
const ChildComponent = () => {  
  return (  
    <div>hello</div>  
  );  
}
```

Composition

Once you define a component, compose with `<NameComponent arg="foo" />`
``${props.foo}` is the value.

```
import PropTypes from 'prop-types';
```

