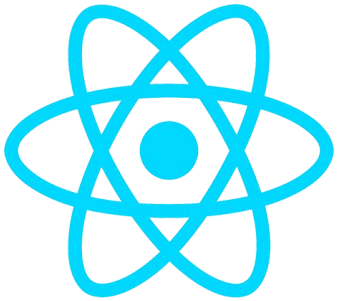

Sponsors

Lessonly

NanoHop

Angie's list

React Indy

6:30 - Pizza
7:00 - Presentations

Chris Achard

Using Modern JS (ES6)
in React and React Native

Want to give a talk? Email: chris@nanohop.com

Sponsors

NanoHop

Angie's list

Intros

Presentation

Chris Achard

Using Modern JS (ES6)
in React and React Native

Using Modern JS (ES6) In React and React Native

CASTER.IO

Home Lessons Courses Instructors Pricing Articles

SIGN IN GO PRO

Recent Releases

Course
Introduction to Android Things

20 lessons, 01:34:03 total

Course
Advanced React Native

17 lessons, 59:44 total

Course
React Native with Redux

21 lessons, 54:04 total

Course
Build a React Native Application

24 lessons, 01:08:15 total

ExoPlayer - Introduction to Audio Playback: Using Custom Resource Files for the Player and

03:27

ES6?

http://www.ecma-international.org/ecma-262/6.0/

Where can you use it?

- Browsers
  - Polyfills, https://caniuse.com/
- React with create-react-app ✔
- React without create-react-app
  - Check webpack config: es2015 or similar
- React Native ✔

What is different?

All the old stuff still works. New stuff is just added:

const, let
Arrow functions
Template literals
Destructuring assignment
Spread operator

Import/Export
Classes
Promises
Async/Await
Generators

var *vs* **const** *vs* **let**

Don’t use **var**

Use **const** if the value won’t change

Use **let** if the value will change

Arrow Functions

Change the way “**this**” is bound

Arrow Functions in React Components

addItem in TodoList.js

See where it's passed in as a prop to the button

With arrow function, don't need .bind in constructor

Defining in class more performant than arrow function directly on prop

Template Literals

Allows for newlines and string interpolation

Destructuring Assignment

Extract from objects and arrays

“Rest” Operator

...rest

Advanced destructuring assignment

Spread Operator

Opposite of ...rest

Creates new objects or arrays

Import / Export

Replaces “require”

Compatible with old libraries

Classes

JS is still not truly object oriented - it’s prototype based

Promises

Avoid problems with callbacks

Async/Await

Even smoother asynchronous code

Generators

A generator is a function that can be paused

Yield control back to the caller