Introduction

ES6 is a major milestone in JavaScript's evolution.

After the rise of ES6, many libraries and frameworks appeared. React.js is considered to be one of the most popular ones.

That's the reason behind why we chose to focus on learning React in this super skill.

During this super skill we are going to:

- Explore React.js and understand the reasons behind its popularity.
- Learn how to do an environment setup so we can get started on using React.js as soon as possible.
- Discover how to get started with React.js and make our first steps.

What is React?

As we have said earlier, React.js is a JavaScript library for building user interfaces.

Well, you might be wondering right now: **what is a library? how can we define it?**

A library can be defined as a set of functions and classes that are grouped in a package.

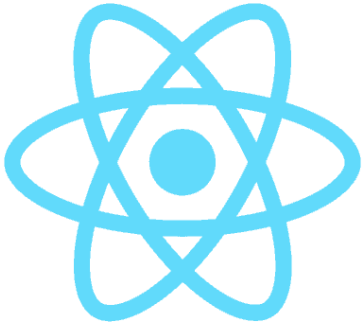
The main idea behind libraries is to create a reusable code that any one in the development community can utilize and contribute to.

What is React?

If you search “what is React.js?” on google you will find hundreds of informative and comprehensive articles talking about React.js. But we’re here to keep things short and sweet.

React.js is an open-source JavaScript library that is used for building user interfaces specifically for Single-Page Applications.

It’s concerned with handling the view layer for web and mobile applications.



What is Single-Page Application ?

Modern apps tend to adhere to what is known as a Single-page App (SPA) model.

In our day-to-day browsing, we never load entirely new and different pages even if we reload the page. Instead, the different views of your app are dynamically loaded and unloaded, written and rewritten into the same page.

The App.js component will be the starting point for our SPA. App.js is the main component.

Why use React?

Oftentimes, in real-world applications, developing a website using HTML, CSS and JavaScript can be very difficult and monotonous. Here's why:

- You need to manually create HTML, CSS and JavaScript files for every website page.
- If you want to move an element that exists in page A to another page B, you will have to copy all of its HTML, CSS and JavaScript to make it work and that is not only tedious but also time-consuming .
- Sharing CSS and JavaScript can make maintenance really difficult as the website grows.

In order to solve the previously mentioned problems, libraries and frameworks have fortunately come to light. They make web development much more easier and more convenient for developers. One of the major solutions is React.js.

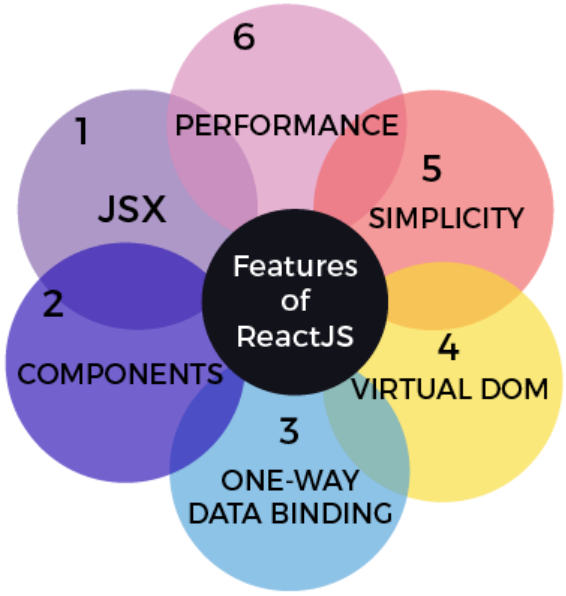
Why should we choose to learn React.js?

The answer for this question is very simple, React is:

- Easy to learn.
- Highly maintainable.
- Super-fast in rendering views.
- Component-based, with the ability to split the code as much as we want
- Simple in the debugging process.

Features of React:

React has several excellent features that are very useful while creating the user interface.



Features of React:

- **JSX**: React uses the JSX syntax in writing components, which makes them more independent.
- **Component**: React adopts the component-based application approach which allows us to use the same component repeatedly.
- **One way data flow**: React only allows us to pass data from parent to child, which is very helpful in tracing data when debugging.
- **Virtual DOM**: React uses the Virtual DOM which makes rendering UI super fast.
- **Simplicity**: React is very simple to learn and work with, especially for newcomers.

Apps built with react

The digital world is always evolving and it is changing as we speak. In a world like that, it is definitely hard to adapt to trends. However, that is exactly what the Big Tech companies in the industry are doing.

Major apps like Facebook, Instagram, Netflix and others are constantly improving their experience and adapting to new frameworks and trends.

As of recently, there is word of mouth going around ReactJS and its impressive features.

The proof for its popularity is best described by the apps that are using ReactJS. Today, we will be showing you the list of the most impressive apps based on ReactJS.

Most of these are : Facebook, Instagram, ADA, Netflix, Dropbox, Paypal, etc...

Example: Netflix movies.

Let's look at an example to highlight the difference between normal HTML and React.

Let's try to rebuild netflix's movie list using HTML, then rebuild it using React. We will compare both versions later on.

Example: Netflix movies using HTML

Using HTML, we end up with this incomprehensible mess of a code:

```
<div class="nm-content-horizontal-row">
 <ul class="nm-content-horizontal-row-item-container">
   <li class="nm-content-horizontal-row-item" role="menuitem">
     <a
       class="nm-collections-title nm-collections-link"
       href="https://www.netflix.com/tn-en/title/70079583"
       data-uia="collections-title"
     >
       <img
         class="nm-collections-title-img"
         src="..."
         data-title-id="70079583"
         alt="The Dark Knight"
       />
       <span class="nm-collections-title-img placeholder"></span>
```

```
    <span class="nm-collections-title-name">The Dark Knight</span>
  </a>
</li>
<li class="nm-content-horizontal-row-item" role="menuitem">
  <a
    class="nm-collections-title nm-collections-link"
    href="https://www.netflix.com/tn-en/title/80013871"
    data-uia="collections-title"
  >
    <img
      class="nm-collections-title-img"
      src="..."
      data-title-id="80013871"
      alt="American Sniper"
    />
    <span class="nm-collections-title-img placeholder"></span>
    <span class="nm-collections-title-name">American Sniper</span>
  </a>
</li>
<li class="nm-content-horizontal-row-item" role="menuitem">
  <a
    class="nm-collections-title nm-collections-link"
    href="https://www.netflix.com/tn-en/title/80064516"
    data-uia="collections-title"
  >
    <img
      class="nm-collections-title-img"
      src="...."
      data-title-id="80064516"
      alt="The Revenant"
    />
    <span class="nm-collections-title-img placeholder"></span>
    <span class="nm-collections-title-name">The Revenant</span>
  </a>
```

```
    </li>
  </ul>
</div>
```

Example: Netflix movies using React.js

On the other hand, after refactoring this code in React using components, we end up having a much cleaner code.

Notice that `<Movie/>` tag is not part of HTML, it's a tag that we created using React. It's called a **React component**.

```
<Container>
 <Row>
   <Movie image={...} title="The Dark Knight"
   href="https://www.netflix.com/tn-en/title/70079583"/> <Movie image={...}
   title="American Sniper"
   href="https://www.netflix.com/tn-en/title/80013871"/> <Movie image={...}
   title="The Revenant" href="https://www.netflix.com/tn-en/title/80064516"/>
 </Row>
</Container>
```

‹ Previous next ›