



به نام خدا



آموزش راهبردی

REACT JS 17

مؤلف:

ایمان احراری

فهرست مطالب

۹.....	مقدمه ناشر
۱۰.....	تقدیم
۱۱.....	مقدمه و پیشگفتار
۱۱.....	مفهوم زبان‌های برنامه‌نویسی
۱۱.....	آشنایی با زبان js
۱۲.....	آشنایی با es6
۱۲.....	مفهوم کتابخانه‌ها
۱۲.....	مفهوم فریم‌ورک‌ها
۱۲.....	مفهوم spa
۱۳.....	React چیست؟
۱۳.....	Next.js
۱۳.....	Gatsby
۱۳.....	بررسی سایر کتابخانه‌ها و فریم‌ورک‌های js
۱۴.....	Npm
۱۴.....	Package.json
۱۴.....	بابل
۱۴.....	پیش‌نیازهای آموزش React
۱۵.....	فصل اول نسل جدید جاوا اسکریپت
۱۵.....	مقدمه
۱۵.....	متغیرها
۱۵.....	عملگرها
۱۷.....	عملگرهای Bitwise
۱۸.....	TEMPLATE LITERALS
۱۸.....	ماژول
۱۹.....	توابع
۱۹.....	کلاس‌ها

فصل دوم نصب و راه اندازی ۲۱

تنظیمات اولیه یک پروژه	۲۱
استفاده از REACT در یک فایل HTML ایستا	۲۱
نصب با استفاده از ترمینال	۲۳
REACT را با JSX امتحان کنید.	۲۳
اضافه کردن JSX به پروژه	۲۴

فصل سوم مفاهیم پایه ۲۶

معرفی JSX	۲۶
چرا JSX؟	۲۶
گنجاندن عبارات در JSX	۲۶
JSX نیز خود یک عبارت است.	۲۷
تعریف خاصیت ها در JSX	۲۸
تعریف تگ های فرزند با JSX	۲۸
جلوگیری از تزریق کد sql (sql injection) با JSX	۲۸
JSX نشان دهنده اشیاء است.	۲۹
رندر عناصر	۳۰
REACT بدون JSX	۳۰
رندر کردن یک عنصر در DOM	۳۱
به روز رسانی عناصر رندر شده	۳۱
کامپوننت ها و ویژگی های آن ها	۳۲
توابع و کلاس های کامپوننت ها	۳۲
استخراج کامپوننت ها	۳۴
پراپس ها فقط خواندنی هستند.	۳۷
استایل دهی به کامپوننت ها	۳۸
استفاده از فایل های CSS	۳۸
استایل های درونی	۴۰
کتابخانه های استایل دهی	۴۲
State ها و چرخه حیات	۴۲
تبدیل تابع به کلاس	۴۴
اضافه کردن Local State به یک کلاس	۴۴
افزودن متدهای چرخه حیات به کلاس	۴۶
استفاده صحیح از STATE	۴۹

۵۱	مدیریت رویدادها
۵۴	انتقال آرگومان ها به EVENT HANDLER ها (گردانندگان رویداد)
۵۵	رندر شرطی
۵۶	متغیرهای عناصر
۵۷	شرط های درون خطی با منطق و عملگر && (AND)
۵۹	If-else درون خطی با عملگرهای شرطی
۵۹	جلوگیری از رندر شدن کامپوننت
۶۱	لیست ها و کلیدها
۶۱	رندر چند کامپوننتی
۶۲	کامپوننت لیست های اصلی
۶۳	کلیدها
۶۴	استخراج کامپوننت ها با کلیدها
۶۵	کلیدها فقط باید در بین خواهر و برادرانشان منحصر به فرد باشند
۶۷	جاسازی MAP() در JSX
۶۸	فرم ها و مدیریت آن ها
۶۸	کامپوننت های کنترل شده
۷۰	تگ textarea
۷۱	تگ select
۷۲	تگ file input
۷۲	مدیریت چندین ورودی
۷۴	ورودی های کنترل شده برای مقادیر خالی
۷۵	جایگزینی برای کامپوننت کنترل شده
۷۵	بالا نگه داشتن state ها
۷۶	افزودن ورودی دوم
۷۸	نوشتن توابع تبدیل
۷۸	بالا نگه داشتن state ها
۸۳	ترکیب و وراثت
۸۴	شامل شدن
۸۵	اجزای تخصصی
۸۷	پس مبحث ارث بری در React چه می شود؟

فصل چهارم مفاهیم پیشرفته ۸۸

۸۸.....	تقسیم‌بندی کدها
۸۹.....	تقسیم کد
۸۹.....	React.lazy
۹۱.....	مرزهای خطا
۹۲.....	تقسیم‌بندی کد مبتنی بر مسیر
۹۲.....	export نام‌گذاری‌شده
۹۳.....	Context
۹۳.....	زمان استفاده از context
۹۵.....	مدیریت و مرزبندی خطاها
۹۷.....	محل قرار دادن مرزهای خطا
۹۷.....	استفاده از رفرنس‌ها
۹۷.....	ارجاع رفرنس به اجزای DOM
۹۸.....	React router
۹۹.....	نصب مسیریاب
۹۹.....	انواع مسیریابی
۹۹.....	کامپوننت‌ها
۱۰۰.....	BrowserRouter
۱۰۰.....	Link
۱۰۱.....	Route
۱۰۲.....	مطابقت با مسیرهای چندگانه
۱۰۲.....	رندرینگ درون خطی
۱۰۲.....	مطابقت پارامتر مسیر دینامیک
۱۰۳.....	فرگمنت
۱۰۵.....	سینتکس کوتاه‌تر
۱۰۶.....	فرگمنت‌های کلیددار
۱۰۶.....	StrictMode یا حالت سخت و دقیق
۱۰۷.....	پروفیلر
۱۰۸.....	کالک تابع onRender
۱۰۹.....	API زمینه
۱۰۹.....	طرز کار API زمینه چگونه است؟
۱۱۱.....	کامپوننت‌های درجه بالا

۱۱۳	رویدادها
۱۱۴	رویدادهای چرخه عمر
۱۱۴	Mounting
۱۱۵	مرحله بهروزرسانی
۱۱۵	غیرفعال سازی کامپوننت
۱۱۶	SyntheticEvent یا رویداد مصنوعی
۱۱۸	معرفی HOOK ها
۱۱۸	Hook ها در یک نگاه
۱۱۹	استفاده از State Hook
۱۲۰	اعلان چند متغیر state
۱۲۰	اما واقعاً هوک چیست؟
۱۲۰	استفاده از Effect Hook
۱۲۳	قوانین Hook
۱۲۳	ساخت اولین Hook
۱۲۵	هوک های بیشتر

فصل پنجم آزمایش کردن ۱۲۶

۱۲۶	تست نرم افزاری
۱۲۶	۱- تست واحد
۱۲۶	۲- تست عملکرد
۱۲۷	۳- تست یکپارچگی
۱۲۷	ابزارهای لازم برای انجام تست
۱۲۷	وابستگی های نرم افزاری برای انجام تست
۱۲۷	jest/jest-cli
۱۲۷	babel-jest/babel-preset-stage-0
۱۲۸	sinon
۱۲۸	react-addons-test-utils/enzymes
۱۲۸	react-test-render
۱۲۸	react-mock-store
۱۲۸	پیکربندی
۱۲۹	نوشتن تست ها
۱۳۰	اجرای تست
۱۳۰	پیش بینی تست ها

۱۳۱	 تست نرم افزار با استفاده از Enzyme
۱۳۴	 Shallow یا Mount؟
۱۳۹	 کار متد find() چیست؟
۱۴۰	 یکپارچه سازی تست
۱۴۰	 نصب
۱۴۲	 نوشتن تست ها

فصل ششم مطالبی که بهتر است بیاموزید. ۱۴۷.....

۱۴۷	 آشنایی اجمالی با REDUX
۱۴۷	 چرا باید از آن استفاده کنیم؟
۱۴۷	 چه زمانی باید از آن استفاده کنیم؟
۱۴۷	 فرآیند کلی Redux
۱۴۸	 آشنایی با مفاهیم SEO در ری اکت
۱۴۹	 در کل آیا React برای سئو خوب است؟
۱۴۹	 آشنایی با Git و مدیریت پروژه و سورس کدها
۱۵۰	 دیپلوی پروژه روی سرور
۱۵۰	 Webpack چیست؟
۱۵۰	 فایل bundle.js چه کاری انجام می دهد؟
۱۵۱	 در نسخه ۱۸ چه خبر است؟

منابع اینترنتی ۱۵۲.....

خط‌مشی انتشارات مؤسسه فرهنگی هنری دیباگران تهران در عرصه کتاب‌هایی با کیفیت عالی است که بتواند
خواسته‌های به روز جامعه فرهنگی و علمی کشور را تا حد امکان پوشش دهد.
هر کتاب دیباگران تهران، یک فرصت جدید شغلی و علمی

حمد و سپاس ایزد منان را که با الطاف بی‌کران خود این توفیق را به ما ارزانی داشت تا بتوانیم در راه ارتقای دانش عمومی و فرهنگی این مرز و بوم در زمینه چاپ و نشر کتب علمی و آموزشی گام‌هایی هرچند کوچک برداشته و در انجام رسالتی که بر عهده داریم، مؤثر واقع شویم.

گسترده‌گی علوم و سرعت توسعه روزافزون آن، شرایطی را به وجود آورده که هر روز شاهد تحولات اساسی چشمگیری در سطح جهان هستیم. این گسترش و توسعه، نیاز به منابع مختلف از جمله کتاب را به عنوان قدیمی‌ترین و راحت‌ترین راه دستیابی به اطلاعات و اطلاع‌رسانی، بیش از پیش برجسته نموده است.

در این راستا، واحد انتشارات مؤسسه فرهنگی هنری دیباگران تهران با همکاری اساتید، مؤلفان، مترجمان، متخصصان، پژوهشگران و محققان در زمینه‌های گوناگون و مورد نیاز جامعه تلاش نموده برای رفع کمبودها و نیازهای موجود، منابعی پُر بار، معتبر و با کیفیت مناسب در اختیار علاقمندان قرار دهد.

کتابی که در دست دارید با همت "جناب آقای ایمان احراری" و تلاش جمعی از همکاران انتشارات میسر گشته که شایسته است از یکایک این گرامیان تشکر و قدردانی کنیم.

با نظرات خود مشوق و راهنمای ما باشید

با ارائه نظرات و پیشنهادات و خواسته‌های خود، به ما کمک کنید تا بهتر و دقیق‌تر در جهت رفع نیازهای علمی و آموزشی کشورمان قدم برداریم. برای رساندن پیام‌هایتان به ما از رسانه‌های دیباگران تهران شامل سایتهای فروشگاهی و صفحه اینستاگرام و شماره‌های تماس که در صفحه شناسنامه کتاب آمده استفاده نمایید.

مدیر انتشارات

مؤسسه فرهنگی هنری دیباگران تهران
dibagaran@mftplus.com

تقدیم و تشکر

تقدیم به همسر مهربانم و دخترکم "باران" که زمان صرف شده برای این کتاب متعلق به آنها بود.

و با تشکر از تمامی معلمین و اساتید بزرگواری که تا آخر عمر در حکم یک شاگرد قدردان زحماتشان هستم و همچنین انتشارات خوب دیباگران که فرصت پدید آمدن این اثر را برای بنده فراهم نمودند.



ایمان احراری

مقدمه و پیشگفتار

مفهوم زبان‌های برنامه‌نویسی

زبان‌های برنامه‌نویسی را شاید بتوان این گونه تعریف کرد، که زبان برنامه‌نویسی یک زبان رسمی است و شامل مجموعه‌ای از رشته‌ها است که توانایی این را دارد تا بتواند انواع مختلفی از خروجی کد ماشین را تولید کند. زبان‌های برنامه‌نویسی یک نوع زبان رایانه‌ای هستند و در برنامه‌نویسی کامپیوتر برای پیاده‌سازی الگوریتم‌ها استفاده می‌شوند. اکثر زبان‌های برنامه‌نویسی؛ شامل دستورالعمل‌هایی برای رایانه‌ها هستند. شاید یکی از بهترین و جذاب‌ترین مهارت‌ها در جهان کنونی، برای داشتن جایگاه شغلی و درآمد خوب، برنامه‌نویسی کامپیوتری^۱ می‌باشد. زبان‌های برنامه‌نویسی را با نگاه‌های مختلف دسته‌بندی می‌کنند، که در ادامه برخی از این دسته‌بندی‌ها را با هم مرور می‌کنیم:

- ۱- با توجه به اینکه به زبان انسان نزدیک باشد و یا به زبان ماشین به دو دسته سطح بالا و سطح پایین تقسیم می‌شود، که زبان‌های سطح بالا به زبان انسان و زبان‌های سطح پایین به زبان ماشین نزدیک هستند.
- ۲- زبان‌های برنامه‌نویسی از نظر نحوه اجرا نیز به دو دسته مفسری^۲ و کامپایلری^۳ تقسیم می‌شوند، که زبان مفسری زبانی است که کدهای آن به صورت خط به خط ترجمه و اجرا می‌شوند و زبان‌های کامپایلری به یکباره کل کدها را ترجمه و اجرا می‌کنند؛ بنابراین سرعت اجرای زبان‌های مفسری نسبت به زبان‌های کامپایلری کم‌تر است.
- ۳- زبان‌های برنامه‌نویسی از نظر اینکه کدها در کدام محل اجرا می‌شوند، نیز دو دسته هستند: زبان‌های سمت سرور و یا زبان‌های سمت کلاینت که با توجه به اینکه کدها در سمت سرور اجرا می‌شوند یا در سیستم کاربر پردازش و اجرا می‌شوند، از هم جدا می‌گردند.

آشنایی با زبان js

زبان برنامه‌نویسی جاوا اسکریپت^۴ که به اختصار به آن js نیز گفته می‌شود، یک زبان سطح بالا، مفسری و سمت کلاینت می‌باشد، که هم در front-end و هم در Back-end به کمک طراحان وب می‌آید و همانگونه هم که از اسم آن پیداست یک زبان اسکریپتی می‌باشد و نیاز است که آن را در لابه‌لای یک زبان دیگر تزریق کرد، که در حوزه طراحی و توسعه وبسایت معمولاً این زبان با زبان‌های HTML و CSS درگیر می‌شود و باعث پویایی و تعاملی شدن صفحات وب می‌گردد (صفحات HTML را که با کمک زبان js تعاملی گردیده‌اند، اصطلاحاً DHTML (Dynamic HTML) گویند).

-
1. Computer Programming
 2. Interpreter
 3. Compiled
 4. JAVA SCRIPT

◀ آشنایی با es6

زبان محبوب برنامه‌نویسی جاوا اسکریپت در سال ۲۰۱۵ میلادی خود نسخه استاندارد ارائه داد، که به آن ES6 یا EcmaScript6 گفته می‌شود. اکما مخفف European Computer Manufacturers Association یا سازمان تولیدکنندگان اروپایی کامپیوتر است که اکما اسکریپت، نسخه جاوا اسکریپت استانداردسازی شده ECMA می‌باشد. از جمله مزایای اکما اسکریپت ۶ نسبت به نسخه‌های قدیمی‌تر جاوا اسکریپت می‌توان به خوانایی و قانونمندتر شدن و تعامل بیش‌تر با کتابخانه‌ها و فریم‌ورک‌های مختلف نام برد که باعث شده‌است توسعه‌دهندگان بیش‌تری به این زبان ترغیب شوند.

◀ مفهوم کتابخانه‌ها

Library یا کتابخانه، مجموعه‌ای از کدها (کلاس‌ها، متدها و ...) می‌باشد که برای سادگی و راحتی بیش‌تر در توسعه نرم‌افزار استفاده می‌شود؛ به عبارت ساده‌تر می‌توان گفت با استفاده از کتابخانه‌ها می‌توان از روی یک یا چندین عملیات یک قالب تهیه کرد و در طول برنامه و یا در برنامه‌های مختلف از آن به راحتی استفاده نمود.

◀ مفهوم فریم‌ورک‌ها

Frameworkها چارچوب‌های کاری هستند، که می‌توان با استفاده از آن‌ها کدنویسی و توسعه نرم‌افزار را اصولی‌تر پیش برد، اشتباهی که در بعضی از تعریف‌ها وجود دارد، این است که فریم‌ورک را مجموعه‌ای از کتابخانه‌ها می‌دانند؛ اما با وجود اینکه این برداشت نیز درست می‌باشد؛ باید دقت کرد فریم‌ورک چارچوب کاری برای کدنویسی و توسعه نرم‌افزار می‌باشد، که در آن ممکن است برنامه‌نویس از چندین کتابخانه هم کمک بگیرد و به عبارتی می‌توان گفت فریم‌ورک‌ها، کتابخانه‌ها و سایر فرآیندهای توسعه نرم‌افزار را رهبری می‌کنند. مفهومی که یادگیری آن به تمامی هنرجویان (به علت اینکه بنده برنامه‌نویسی و طراحی وبسایت را یک هنر می‌دانم، به علاقمندان یادگیری و افزایش مهارت در این حوزه هنرجو می‌گویم) این رشته توصیه می‌شود این است که پس از یادگیری یک زبان (چه نشانه‌گذاری؛ چه طراحی؛ چه برنامه‌نویسی) حتماً بر روی فریم‌ورک آن زبان نیز تسلط پیدا کنند. به طور مثال برای زبان html و css فریم‌ورک‌های مختلفی؛ مانند Bootstrap، Tailwindcss و ... وجود دارد. برای زبان php فریم‌ورکی؛ مانند Laravel و ... وجود دارد. برای Python فریمی؛ مانند Django وجود دارد و برای جاوا اسکریپت و اکما اسکریپت هم فریم‌ورک‌هایی؛ مانند Angular، Vue، React و ... وجود دارد، که کارها را بسیار اصولی‌تر و بر روی یک ساختار مشخص جلو می‌برد و امروزه زبان برنامه‌نویسی و طراحی‌ای وجود ندارد، که فریم‌ورک مطرخی در آن وجود نداشته باشد.

◀ مفهوم spa

Single Page Application یا همان برنامه‌های تک‌صفحه‌ای به نرم‌افزارهایی گفته می‌شود که تمامی صفحات، رویدادها و بخش‌های مختلف یک برنامه (وبسایت) به ظاهر در یک صفحه اتفاق می‌افتد و از Reload کل صفحه و تغییر URLهای متعدد جلوگیری می‌شود. اگر بخواهیم برای نمونه مثالی بزنیم که برای شما بیش‌تر نمایان شود، صفحات ورود و یا عضویت در Gmail شاید واضح‌ترین مثال برای این کار باشد. در این تکنیک کاربر بدون نیاز به بارگذاری مجدد صفحات از وبسایت لذت می‌برد و این امر باعث بالا رفتن UX یا همان تجربه کاربری بازدیدکنندگان

سایت خواهد شد. برنامه‌های تک‌صفحه‌ای دست تیم فرانت کار را برای ایجاد تغییرات و انجام خلاقیت‌های بیش‌تر باز می‌گذارد، به نحوی که تیم بک‌اندکار تنها تمرکز خود را بر روی ساخت و توسعه API ها قرار خواهند داد. فریم‌ورک‌های جاوا اسکریپت؛ نظیر Angular و React به خوبی این تکنیک را پیاده‌سازی می‌کنند؛ اما باید توجه داشت که استفاده از این تکنیک باید به طرز سالم و کامل بر روی وبسایت پیاده‌سازی گردد تا این نقطه مثبت به نقطه ضعف محصول نهایی تبدیل نگردد.

◀ React چیست؟

React یکی از کتابخانه‌های محبوب و پُرطرفدار جاوا اسکریپت با بیش از صدهزار ستاره در گیت هاب می‌باشد، که به عنوان یک پروژه متن باز توسط فیس‌بوک توسعه داده می‌شود. React بر خلاف انگولار یک چارچوب نیست و برای ساخت رابط‌های کاربری و لایه نمایشی در سمت فرانت وبسایت‌ها استفاده می‌شود و در حال حاضر با استفاده از این تکنولوژی وبسایت‌های زیادی در سراسر دنیا توسعه داده می‌شوند. با استفاده از React می‌توانید استاندارد برای محصولات نرم‌افزاری خود آماده کنید و قدم‌های اولیه حرکت به سمت PWA را در پروژه‌های نرم‌افزاری خود بردارید. React با ایجاد یک DOM مجازی در حافظه تمامی تغییرات لازم را قبل از اینکه در مرورگر اعمال شود، در شبیه‌ساز خود اعمال می‌کند و خوبی آن این است که تنها مواردی را که در صفحه نیاز به تغییر دارند، به‌روزرسانی و ویرایش می‌کند (SPA) و از به‌روزرسانی و Reload صفحات به صورت کلی پرهیز به عمل می‌آورد، که این کار نیز باعث افزایش لود صفحات خواهد شد. یکی دیگر از مهم‌ترین جنبه‌های React این واقعیت است که می‌توانید مؤلفه‌هایی؛ مانند عناصر HTML سفارشی و قابل استفاده مجدد برای ساخت سریع و کارآمد رابط کاربری ایجاد کنید و همچنین نحوه ذخیره و مدیریت داده‌ها را با استفاده از state ها و props ساده می‌کند.

◀ Next.js

Next.js یک فریم‌ورک محبوب و سبک‌وزن برای برنامه‌های استاتیک و رندر شده توسط سرور است، که با React ساخته شده است. این فریم‌ورک؛ شامل راه‌حلی برای درج استایل و مسیریابی خارج از چارچوب است و فرض می‌کند که از Node.js به عنوان محیط سرور خود استفاده می‌کنید.

◀ Gatsby

گتسبی بهترین راه برای ایجاد وبسایت‌های استاتیک با React است، که به شما این امکان را می‌دهد تا بتوانید در پروژه‌های خود از کامپوننت‌های React استفاده کنید؛ اما باید توجه کنید که این ابزار HTML و CSS پیش‌رندر شده را برای تضمین سریع‌ترین زمان بارگذاری، در خروجی قرار می‌دهد.

◀ بررسی سایر کتابخانه‌ها و فریم‌ورک‌های JS

شاید یکی از متداول‌ترین سوالاتی که هنجاریان این حوزه دنبال آن هستند، این است که در بین کتابخانه‌ها و فریم‌ورک‌های موجود JS کدام یک از بقیه بهتر است و مدام در مقالات مختلف به بررسی نقاط ضعف و قوت آن‌ها می‌پردازند؛ اما به طور کلی و یکبار برای همیشه می‌خواهم خیالتان را از این بابت راحت کنم. در حال حاضر به علت

پیشرفت هر سه نوع تکنولوژی مطرح این زبان React، Angular و Vue به صورت صد در صد نمی تواند یکی را به عنوان برنده اعلام کرد و بیش تر باید با یک هدف گذاری مناسب و انتخاب مسیر آینده یکی از این سه تکنولوژی را برای اینکه بر روی آن تسلط پیدا کنید، انتخاب نمایید؛ اما در این کتاب با توجه به اینکه در کشور عزیزمان ایران تیم ها و شرکت های زیادی به دنبال توسعه دهنده React هستند و همچنین استفاده از JSX، حجم کم بسته های نصبی، SEO Friendly بودن، سینتکس ساده و نزدیکی با React Native به بررسی این فریم ورک پرداخته ایم، که امیدواریم در پایان کتاب از آن بهره مند و راضی باشید.

◀ Npm

Npm یا همان پکیج منیجر نودجی اس می باشد، که لازم می دانم قبل از هر توضیح دیگری ابتدا یک تعریف کلی از Node.js به شما ارائه دهم و آن این است که با استفاده از نودجی اس برنامه های جاوا اسکریپتی را می توان در طرف سرور اجرا کرد و در ادامه و با استفاده از NPM شما می توانید ابزارهایی را که برای کارتان نیاز دارید، به راحتی مدیریت کنید و آن ها را در اختیار داشته باشید.

◀ Package.json

بعد از نصب npm در داخل شاخه پروژه فایلی با فرمت json به وجود می آید، که داخل آن اطلاعات و تنظیماتی وجود دارد که با توجه به آن ها npm می تواند پروژه شما را مدیریت کند.

◀ بابل

Babeljs یک کامپایلر جاوا اسکریپت است، که جاوا اسکریپت نسخه های جدید را به جاوا اسکریپت ES5 قدیمی تبدیل می کند و این امکان را به کدهای ما می دهد تا بتواند در هر مرورگری اجرا شود. Babeljs تمام ویژگی های سینتکسی را که به نسخه های جدید جاوا اسکریپت افزوده شده است، پشتیبانی می کند و حتی توانایی این را دارد تا بتواند دستورات JSX را نیز به جاوا اسکریپت قابل فهم برای تمامی مرورگرها تبدیل کند!

شما هنرجویان عزیز می توانید با رفتن به وبسایت رسمی آن به آدرس <https://babeljs.io> از جدیدترین مستندات فنی و اطلاعیه های آن باخبر شوید.

◀ پیش نیازهای آموزش React

در ادامه مواردی وجود دارد که از آن ها به عنوان پیش نیازهای React یاد می شود و شما برای اینکه به درک بهتر، سریع تر و در عین حال عمیق تری نسبت به React برسید، باید آن را بیاموزید:

- آشنایی اولیه با HTML و CSS

- دانش اولیه جاوا اسکریپت و برنامه نویسی

- درک اولیه از DOM

- آشنایی با ویژگی های جدید ES6