[Difference between NPM and NVM](https://stackoverflow.com/questions/32660993/difference-between-npm-and-nvm)

[nvm](https://github.com/creationix/nvm) (Node Version Manager) is a tool that allows you to download and install Node.js. Check if you have it installed via nvm --version.

[npm](https://www.npmjs.com/) (Node Package Manager) is a tool that allows you to install javascript packages. Check if you have it installed via npm --version.

npm comes with Node.js so if you have node installed (node --version) you most likely have npm installed as well.

You don't need nvm unless you you want to keep multiple versions of Node.js installed on your system or if you'd like to upgrade your current version.

# What is React?

React is a **JavaScript** library created by **Facebook**

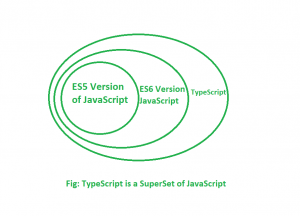
React is a **User Interface** (UI) library

React is a tool for building **UI components**

Before you start, you should have a basic understanding of:

* What is [HTML](https://www.w3schools.com/whatis/whatis_html.asp)
* What is [CSS](https://www.w3schools.com/whatis/whatis_css.asp)
* What is [DOM](https://www.w3schools.com/whatis/whatis_htmldom.asp)
* What is [ES6](https://www.w3schools.com/js/js_es6.asp)
* What is [Node.js](https://www.w3schools.com/nodejs/default.asp)
* What is [npm](https://www.w3schools.com/whatis/whatis_npm.asp)

**Difference between TypeScript and JavaScript**



**Why TypeScript is developed while having JavaScript?**

When JavaScript was developed, the JavaScript development team introduced JavaScript as a client-side programming language. But as people were using JavaScript, developers also realized that JavaScript could be used as a server-side programming language. However, as JavaScript was growing, JavaScript code became complex and heavy. Because of this, JavaScript wasn’t even able to fulfill the requirement of an Object-Oriented Programming language. This prevented JavaScript from succeeding at the enterprise level as a server-side technology. So **TypeScript** was created by the development team to bridge this gap.

**Features of TypeScript:**

* **TypeScript Code is converted into Plain JavaScript Code:** TypeScript code can’t be natively interpreted by browsers. So if the code was written in TypeScript, it gets compiled and converted into JavaScript. This process is known as **Trans-piled**. With the help of JavaScript code, browsers are able to read the code and display it.
* **JavaScript is TypeScript**: Whatever code is written in JavaScript can be converted to TypeScript by changing the extension from **.js** to **.ts**.
* **Use TypeScript anywhere:** TypeScript can be compiled to run on any browser, device, or operating system. TypeScript is not specific to any single environment.
* **TypeScript supports JS libraries:** With TypeScript, developers can use already existing JavaScript code, incorporate popular JavaScript libraries, or call the TS Code from native JavaScript code.

**Difference between TypeScript and JavaScript:**

* TypeScript is known as an Object-oriented programming language whereas JavaScript is a prototype based language.
* TypeScript has a feature known as Static typing but JavaScript does not support this feature.
* TypeScript supports Interfaces but JavaScript does not.

**Advantages of using TypeScript over JavaScript**

* TypeScript always points out the compilation errors at the time of development (pre-compilation). Because of this getting runtime errors is less likely, whereas JavaScript is an interpreted language.
* TypeScript supports static/strong typing. This means that type correctness can be checked at compile time. This feature is not available in JavaScript.
* TypeScript is nothing but JavaScript and some additional features i.e. ES6 features. It may not be supported in your target browser but the TypeScript compiler can compile the **.ts** files into ES3, ES4, and ES5 also.

**Disadvantages of using TypeScript over JavaScript**

* Generally, TypeScript takes time to compile the code.