一、新手向：Vue 2.0 的建议学习顺序

注：2.0 已经有中文文档 。如果对自己英文有信心，也可以直接阅读英文文档。

此指南仅供参考，请根据自身实际情况灵活调整。

https://cn.vuejs.org/v2/guide/

起步

1. 扎实的 JavaScript / HTML / CSS 基本功。这是前置条件。

2. 通读官方教程 (guide) 的基础篇。不要用任何构建工具，就只用最简单的 <script>，把教程里的例子模仿一遍，理解用法。不推荐上来就直接用 vue-cli 构建项目，尤其是如果没有 Node/Webpack 基础。

3. 照着官网上的示例，自己想一些类似的例子，模仿着实现来练手，加深理解。

4. 阅读官方教程进阶篇的前半部分，到『自定义指令 (Custom Directive) 』为止。着重理解 Vue 的响应式机制和组件生命周期。『渲染函数（Render Function)』如果理解吃力可以先跳过。

5. 阅读教程里关于路由和状态管理的章节，然后根据需要学习 vue-router 和 vuex。同样的，先不要管构建工具，以跟着文档里的例子理解用法为主。

6. 走完基础文档后，如果你对于基于 Node 的前端工程化不熟悉，就需要补课了。下面这些严格来说并不是 Vue 本身的内容，也不涵盖所有的前端工程化知识，但对于大型的 Vue 工程是前置条件，也是合格的『前端工程师』应当具备的知识。

前端生态/工程化

1. 了解 JavaScript 背后的规范，ECMAScript 的历史和目前的规范制定方式。学习 ES2015/16 的新特性，理解 ES2015 modules，适当关注还未成为标准的提案。

2. 学习命令行的使用。建议用 Mac。

3. 学习 Node.js 基础。建议使用 nvm 这样的工具来管理机器上的 Node 版本，并且将 npm 的 registry 注册表配置为淘宝的镜像源。至少要了解 npm 的常用命令，npm scripts 如何使用，语义化版本号规则，CommonJS 模块规范（了解它和 ES2015 Modules 的异同），Node 包的解析规则，以及 Node 的常用 API。应当做到可以自己写一些基本的命令行程序。注意最新版本的 Node (6+) 已经支持绝大部分 ES2015 的特性，可以借此巩固 ES2015。

4. 了解如何使用 / 配置 Babel 来将 ES2015 编译到 ES5 用于浏览器环境。

5. 学习 Webpack。Webpack 是一个极其强大同时也复杂的工具，作为起步，理解它的『一切皆模块』的思想，并基本了解其常用配置选项和 loader 的概念/使用方法即可，比如如何搭配 Webpack 使用 Babel。学习 Webpack 的一个挑战在于其本身文档的混乱，建议多搜索搜索，应该还是有质量不错的第三方教程的。英文好的建议阅读 Webpack 2.0 的文档，比起 1.0 有极大的改善，但需要注意和 1.0 的不兼容之处。

Vue 进阶

1. 有了 Node 和 Webpack 的基础，可以通过 vue-cli 来搭建基于 Webpack ，并且支持单文件组件的项目了。建议用 webpack-simple 这个模板开始，并阅读官方教程进阶篇剩余的内容以及 vue-loader 的文档，了解一些进阶配置。有兴趣的可以自己亲手从零开始搭一个项目加深理解。

2. 根据 例子 尝试在 Webpack 模板基础上整合 vue-router 和 vuex

3. 深入理解 Virtual DOM 和『渲染函数 (Render Functions)』这一章节（可选择性使用 JSX)，理解模板和渲染函数之间的对应关系，了解其使用方法和适用场景。

4. （可选）根据需求，了解服务端渲染的使用（需要配合 Node 服务器开发的知识）。其实更重要的是理解它所解决的问题并搞清楚你是否需要它。

5. 阅读开源的 Vue 应用、组件、插件源码，自己尝试编写开源的 Vue 组件、插件。

6. 参考 贡献指南 阅读 Vue 的源码，理解内部实现细节。（需要了解 Flow）