1. vue框架有什么优点
2. 只关注视图层的轻巧型框架
3. 实现数据的双向绑定，减少DOM操作
4. 组件化及响应式设计
5. 数据与结构进行分离，易于浏览器的加载速度
6. 实现组件化的目的是什么（有什么好处）

（1）基于可重用性的目的，将一个大的软件系统按照分离关注点的形式，拆分成多个独立的组件，主要目的就是减少耦合。

1. Html、css、js实现什么功能

（1）html是创建页面元素

（2）css是为页面元素提供样式

（3）js是为页面样式提供响应时间及方法

1. Node.js充当什么作用，对于node.js有什么样的理解

Node.js是运行在服务器的javascript

1. 你接触过哪些数据结构，二叉树有什么优点

数据结构：计算机存储以及组织数据的方式

常见数据结构：

1. 线性结构：数组、栈、队列、线性表
2. 树形结构：树，堆
3. 图形结构

二叉树：查询速度快

1. 未来三年的职业生涯规划

哈希表：以键值存储数据的结构

优点：不论哈希表数据量有多少，他的插入、删除、查询速度快，基本上能达到o（1）

运算速度非常快

不需要有序的遍历数据，也不需要提前知道数据量的多少

前端路由，海量数据有作用

<http://webpack.wuhaolin.cn/> webpack的晋级读物 ,以后用webpack的时候,可以看下

1. Cookie

Cookie为持久保存客户端数据提供了方便，分担了服务器存储的压力，但是存在一些弊端

1. cookie数量及长度的限制，每个域名domain最多能存放20条cookie,每个cookie大小不超过4kb，否则会被拦截掉
2. 安全性问题：如果cookie被人拦截，那么他将获得所有的session信息。即使加密也于事无补，因为拦截者不知道cookie的意义，他只要同样转发就能实现效果。
3. 有些状态不可能保存在客户端。

2,数组的解构赋值

|  |
| --- |
| 1、数组的解构赋值            基本用法          ES6允许按照一定模式，从数组和对象中提取值，对变量进行赋值，这被称为解构（Destructuring）。        以前，为变量赋值，只能直接指定值。      var a = 1; var b = 2; var c = 3;        ES6允许写成下面这样。    var [a, b, c] = [1, 2, 3];    let [foo, [[bar], baz]] = [1, [[2], 3]];  foo // 1 bar // 2 baz // 3    let [x, , y] = [1, 2, 3];  x // 1 y // 3    let [x, y, ...z] = ['a'];  x // "a" y // undefined z // []    let [head, ...tail] = [1, 2, 3, 4];  head // 1 tail // [2, 3, 4]               let [ , , third] = ["foo", "bar", "baz"];  third // "baz" |

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | 上面代码表示，可以从数组中提取值，按照对应位置，对变量赋值。        本质上，这种写法属于“模式匹配”，只要等号两边的模式相同，左边的变量就会被赋予对应的值。下面是一些使用嵌套数组进行解构的例子。 |