

2021年前端大厂面试真题

——核心知识篇

- 悄 悄 变 强 大 -



谨以此书献给所有辛勤努力的同学，
他们的积极分享是我们创作灵感的
源泉。祝早日拿到心仪的OFFER。

—— 永生

“天才是1%的灵感，99%的汗水，
但那1%的灵感是最重要的，甚至比
那99%的汗水都要重要。

—— 李春阳

本书在写作风格上推陈新出，对于用例的讲解，不仅文字描述，更以事例佐证。我们咬文嚼字，字斟句酌，所有的这些付出，只为能让读者对书中的技术点放心，文字描述舒心，感受编程的魅力之处。

—— 陈赟帆



目录

第一章 JavaScript	7
01 函数	8
02 异步编程	13
03 设计模式	18
第二章 Vue	24
01 Vue 的组件时间	25
02 深入 Vue 源码设计	27
03 Vue 生态及实践	29
04 TypeScript 基础篇	31
05 TypeScript 提高篇	32
06 TypeScript 实战	34
07 Vue3.x 的设计理念	35
08 Vue4.x API 设计	37
第三章 React	40
01 步入 React 前厅	41



02 探索 React 正殿	43
03 React 生态应用	44
04 React 原理	45
05 React 状态管理	46
06 React 高级实战和性能优化	47

第四章 Node.js 48

01 Node.js 基础	49
02 Web 服务及 Koa	52
03 企业级 Node.js 框架	53

第五章 工程化 57

01 构建艺术	58
02 持续集成与部署	59
03 Git 操作	61

第六章 计算机网络 63

01 网络协议	64
02 网络请求实战	66
03 网络安全与攻防	67
04 浏览器状态和路由	68
05 工具链和其他	69

更多课程相关项目资料，扫码添加课程助教领取 >>> >>>



Javascript

第一章

01 函数

02 异步编程

03 设计模式

01 函数

考点 1 关于 `typeof`、`instanceof`、`toString` 这三种方法区别的理解。

难度系数：★★☆☆☆

【百度移动互联网事业部 / 阿里云智能事业部】

网易专家解析：

三种都是 JS 中用于获取数据类型的方法，以下是对这三种方法的理解：

1、`typeof` 用于比较基础数据类型和引用类型，返回值有 "number"、"string"、"boolean"、"null"、"function"、"undefined"、"symbol"、"object" 原子类型字符串、数值、布尔值的 `typeof` 运算返回相应的类型值；而合成类型中只有函数的 `typeof` 运算返回 'function'，其他都返回 'object'。存在两个特殊值：`typeof undefined` 为 'undefined'，而 `typeof null` 为 'object'。`typeof` 操作符返回一个字符串，表示未经求值的操作数(unevaluated operand)的类型。

2、`instanceof` 用来判断一个对象是否为某个构造函数的实例。值得注意的是原子类型的值不是对象，所以不能用 `instanceof`。对原型链中的 `_proto_` 逐层向上进行查找，通过 `instanceof` 可以判断一个事例的父类型和祖先类型的实例。`instanceof` 运算符可以用来判断某个构造函数的 `prototype` 属性是否存在另外一个要检测对象的原型链上，返回 `boolean` 值。

3、`toString` 要想区别对象、数组、函数单纯使用 `typeof` 是不行的。`null` 和 `Array` 的结果也是 `object`，有时候我们需要的是 "纯粹" 的 `object` 对象。可以通过使用 `toString.call(obj)` 来检测对象类型。



考点 2 对于 isNaN 和 Number.isNaN 函数区别的理解。

难度系数: ★★☆☆☆

【腾讯云与智慧产业事业部】

网易专家解析:

算术运算中返回一个未定义或无法表示的值。为了判断一个计算结果或者变量的值是否为 NaN，我们不能简单的使用 `NaN == NaN`, 或者 `NaN === NaN`, 因为 `NaN != NaN` 的。所以在 ES6 之前我们用全局的 `isNaN()`方法去做判断是比较安全的。

`Number.isNaN()`方法是 `Number` 对象的扩展方法。主要用于判断数值是否为 `Nan`。

函数 `isNaN` 接收参数后，会尝试将这个参数转换为数值，任何不能被转换为数值的的值都会返回 `true`，因此非数字值传入也会返回 `true`，会影响 `Nan` 的判断。

函数 `Number.isNaN` 会首先判断传入参数是否为数字，如果是数字再继续判断是否为 `Nan`，这种方法对于 `Nan` 的判断更为准确。

`Number.isNaN()`方法和 `isNaN()`的不同点是，`Number.isNaN()`不会将传入的非数值类型进行强制转换为数值类型，而是首先判断传入的参数是否为数值类型，只要是非数值类型就直接返回 `false`。

考点 3 关于 Let、Var 和 const 区别的理解。

难度系数: ★★☆☆☆

【腾讯微信事业部】

网易专家解析:

1、`var`: 解析器在解析 js 的时候，会将脚本都扫描一遍，将变量的声明提前到代码块的顶部，赋值还是在原先的位置。

2、`let`: 声明只在当前代码块内，`var` 的声明会被提升到全局中。

3、`const`: 不可重复声明，同一作用域下 `let` 和 `const` 不能声明同名变量，而 `var` 可以；

4、`var` 声明的变量会挂载在 `window` 上，而 `let` 和 `const` 声明的变量不会。



5、var 声明变量存在变量提升，let 和 const 不存在变量提升。

6、let 和 const 声明形成块作用域。

考点 4 Javascript 中，对于 hasOwnProperty 函数的理解。

难度系数：★★☆☆☆

【字节跳动互动娱乐事业部 / 百度 web 前端研发部】

网易专家解析：

hasOwnProperty 函数，所有继承了 Object 的对象都会继承到 hasOwnProperty 方法。这个方法可以用来检测一个对象是否含有特定的自身属性，和 in 运算符不同，该方法会忽略掉那些从原型链上继承到的属性。

1、hasOwnProperty()函数用于指示一个对象自身(不包括原型链)是否具有指定名称的属性。如果有，返回 true，否则返回 false。

2、语法：object.hasOwnProperty(propertyName)。

3、hasOwnProperty()函数的返回值为 Boolean 类型。如果对象 object 具有名称为 propertyName 的属性，则返回 true，否则返回 false。

4、此方法不会检查对象的原型链中是否存在该属性，该属性只有是对象本身的一个成员才会返回 true。



考点 5 普通函数和箭头函数的区别。

难度系数: ★★★☆☆

【腾讯网络媒体事业部 / 阿里产业电商事业部】

网易专家解析:

箭头函数表达式的语法比函数表达式更简洁，并且没有自己的 this, arguments, super 或 new.target。箭头函数表达式更适用于那些本来需要匿名函数的地方，并且它不能用作构造函数。

普通函数:

- 1、可以通过 bind、call、apply 改变 this 指向；
- 2、可以使用 new。

箭头函数:

- 1、本身没有 this 指向；
- 2、它的 this 在定义的时候继承自外层第一个普通函数的 this；
- 3、被继承的普通函数的 this 指向改变，箭头函数的 this 指向会跟着改变；
- 4、箭头函数外层没有普通函数时，this 指向 window；
- 5、不能通过 bind、call、apply 改变 this 指向；
- 6、使用 new 调用箭头函数会报错，因为箭头函数没有 constructor。

考点 6 Object.is()与比较操作符 “==”、“==” 的区别。

难度系数: ★★☆☆☆

【网易互动娱乐事业部 / 字节跳动互动娱乐事业部】

网易专家解析:

- 1、两等号判等，会在比较时进行类型转换；
- 2、三等号判等(判断严格)，比较时不进行隐式类型转换，(类型不同则会返回 false)；
- 3、Object.is 在三等号判等的基础上特别处理了 NaN、-0 和 +0，保证 -0 和 +0 不再相同，但 Object.is(NaN, NaN) 会返回 true。Object.is 应被认为有其特殊的用途，而不能用它认为它比其



它的相等对比更宽松或严格。

考点 7 介绍 js 的基本数据类型。

难度系数: ★☆☆☆☆

【网易雷火事业部】

网易专家解析:

JavaScript 一共有八种数据类型，其中有七种基本数据类型：Undefined、Null、Boolean、

Number、String、Symbol (es6 新增，表示独一无二的值)；目前 es10 新增一个：BigInt。

一种引用数据类型——Object (Object 本质上是由一组无序的名值对组成的)。里面包含 function、Array、Date 等。JavaScript 不支持任何创建自定义类型的机制，而所有值最终都将 是上述 八种数据类型之一。



02 异步编程

考点 8 关于跨域问题的解决方法。

难度系数: ★★★☆☆

【字节跳动互动娱乐事业部 / 快手电商事业部】

网易专家解析:

解决跨域请求的方法有: jsonp 是利用页面中的脚本标记的加载来实现跨域请求的; cors 是通过响应头中指定的源点与当前源点相匹配来实现跨域请求

跨域问题的解决方法有以下九点:

- 1、通过 jsonp 跨域;
- 2、document.domain + iframe 跨域;
- 3、location.hash + iframe;
- 4、window.name + iframe 跨域;
- 5、postMessage 跨域;
- 6、跨域资源共享 (CORS);
- 7、nginx 代理跨域;
- 8、nodejs 中间件代理跨域;
- 9、WebSocket 协议跨域。

考点 9 关于同步和异步的区别。

难度系数: ★★☆☆☆

【小米互动娱乐事业部 / 美团互联网云事业部】

网易专家解析:

同步指的是当一个进程在执行某个请求的时候, 如果这个请求需要等待一段时间才能返回, 那么这个进程会一直等待下去, 直到消息返回为止再继续向下执行。



异步指的是当一个进程在执行某个请求的时候，如果这个请求需要等待一段时间才能返回，这个时候进程会继续往下执行，不会阻塞等待消息的返回，当消息返回时系统再通知进程进行处理。

考点 10 Ajax 解决浏览器缓存问题的方法。

难度系数: ★★★☆☆

【百度智能生活事业部】

网易专家解析:

Ajax 解决浏览器缓存问题的方法有以下五点：

- 1、在 ajax 发送请求前加上 `anyAjaxObj.setRequestHeader("If-Modified-Since", " 0")`。
- 2、在 ajax 发送请求前加上 `anyAjaxObj.setRequestHeader("Cache-Control", " no-cache")`。
- 3、在 URL 后面加上一个随机数： `" fresh=" + Math.random()`;
- 4、在 URL 后面加上时间戳： `" nowtime=" + new Date().getTime()`;
- 5、如果是使用 jQuery，直接这样就可以了`$.ajaxSetup({cache:false})`。这样页面的所有 ajax 都会执行这条语句就是不需要保存缓存记录。

考点 11 对 Ajax 理解以及创建一个 Ajax 的具体步骤。

难度系数: ★★★☆☆

【阿里社区电商事业部】

网易专家解析:

2005 年 2 月，AJAX 这个词第一次正式提出，AJAX 指异步 JavaScript 及 XML (Asynchronous JavaScript And XML)，它是 Asynchronous JavaScript and XML 的缩写，指的是通过 JavaScript 的 异步通信，从服务器获取 XML 文档从中提取数据，再更新当前网页的对应部分，而不用刷新整个网页。Ajax 的核心是 JavaScript 对象 XMLHttpRequest。该对象在 Internet Explorer 5 中首次引入，它是一种支持异步请求的技术。简而言之，XMLHttpRequest 使您可以



使用 JavaScript 向服务器提出请求并处理响应，而不阻塞用户。

具体来说，创建 AJAX 包括以下 6 个步骤：

- 1、创建 XMLHttpRequest 对象，也就是创建一个异步调用对象；
- 2、创建一个新的 HTTP 请求，并指定该 HTTP 请求的方法、URL 及验证信息；
- 3、设置响应 HTTP 请求状态变化的函数；
- 4、发送 HTTP 请求；
- 5、获取异步调用返回的数据；
- 6、使用 JavaScript 和 DOM 实现局部刷新。

考点 12 对于 promise 的理解。

难度系数: ★★★☆☆

【字节跳动互动娱乐事业部】

网易专家解析：

Promise，翻译过来是承诺，承诺它过一段时间会给你一个结果。从编程讲 Promise 是异步编程的一种解决方案。下面是 Promise 在 MDN 的相关说明：

Promise 对象是一个代理对象（代理一个值），被代理的值在 Promise 对象创建时可能是未知的。它允许你为异步操作的成功和失败分别绑定相应的处理方法（handlers）。这让异步方法可以像同步方法那样返回值，但并不是立即返回最终执行结果，而是一个能代表未来出现的结果的 promise 对象。

Promise 是 js 中的一个对象，用于生成可能在将来产生结果的值。值可以是已解析的值，也可以是说明为什么未解析该值的原因。Promise 可以有以下三种状态：

- 1、pending：初始状态，既不是成功也不是失败
- 2、fulfilled：意味着操作完全成功



3、rejected：意味着操作失败

一个等待状态的 promise 对象能够成功后返回一个值，也能失败后带回一个错误，当这两种情况发生的时候，处理函数会排队执行通过 then 方法会被调用。

resolve 和 reject 分别是两个函数，当在回调中调用时，会改变 promise 实例的状态，resolve 改变状态为成功，reject 为失败。

考点 13 对于 AJAX 的理解。

难度系数: ★★★☆☆

【腾讯互动娱乐事业部 / 百度智能生活事业部】

网易专家解析：

2005 年 2 月，AJAX 这个词第一次正式提出，它是 Asynchronous JavaScript and XML 的缩写，指的是通过 JavaScript 的 异步通信，从服务器获取 XML 文档从中提取数据，再更新当前网页的对应部分，而不用刷新整个网页。

具体来说，创建 AJAX 包括以下 6 个步骤：

- 1、创建 XMLHttpRequest 对象，也就是创建一个异步调用对象；
- 2、创建一个新的 HTTP 请求，并指定该 HTTP 请求的方法、URL 及验证信息；
- 3、设置响应 HTTP 请求状态变化的函数；
- 4、发送 HTTP 请求；
- 5、获取异步调用返回的数据；
- 6、使用 JavaScript 和 DOM 实现局部刷新。

考点 14 JS 区分微任务和宏任务的原因。

难度系数: ★★★★☆

【网易有道事业部 / 腾讯技术工程事业部】

网易专家解析：



JS 区分微任务和宏任务有以下五点：

- 1、js 是单线程的，但是分同步异步；
- 2、微任务和宏任务皆为异步任务，它们都属于一个队列；
- 3、宏任务一般是：script, setTimeout, setInterval, setImmediate；
- 4、微任务：原生 Promise；
- 5、遇到微任务，先执行微任务，执行完后如果没有微任务，就执行下一个宏任务，如果有微任务，就按顺序一个一个执行微任务。

考点 15 对于模块化开发的理解。

难度系数: ★★★☆☆

【快手商业化事业部】

网易专家解析：

一个模块是实现一个特定功能的一组方法。在最开始的时候，JavaScript 只实现一些简单的功能，所以并没有模块的概念，但随着程序越来越复杂，代码的模块化开发变得越来越重要。由于函数具有独立作用域的特点，最原始的写法是使用函数来作为模块，几个函数作为一个模块，但是这种方式容易造成全局变量的污染，并且模块间没有联系。后面提出了对象写法，通过将函数作为一个对象的方法来实现，这样解决了直接使用函数作为模块的一些缺点，但是这种办法会暴露所有的所有的模块成员，外部代码可以修改内部属性的值。现在最常用的是立即执行函数的写法，通过利用闭包来实现模块私有作用域的建立，同时不会对全局作用域造成污染。



03 设计模式

考点 16 对于 cookie 的理解。

难度系数: ★☆☆☆☆

【阿里云智能事业部】

网易专家解析:

cookie 是服务器提供的一种用于维护会话状态信息的数据，通过服务器发送到浏览器，浏览器保存在本地，当下一次有同源的请求时，将保存的 cookie 值添加到请求头部，发送给服务端。这可以用来实现记录用户登录状态等功能。cookie 一般可以存储 4k 大小的数据，并且只能被同源的网页所共享访问。

服务器端可以使用 Set-Cookie 的响应头部来配置 cookie 信息。一条 cookie 包括了 5 个属性值 expires、domain、path、secure、HttpOnly。其中 expires 指定了 cookie 失效的时间，domain 是域名、path 是路径，domain 和 path 一起限制了 cookie 能够被哪些 url 访问。secure 规定了 cookie 只能在确保安全的情况下传输，HttpOnly 规定了这个 cookie 只能被服务器访问，不能使用 js 脚本访问。

在发生 xhr 的跨域请求的时候，即使是同源下的 cookie，也不会被自动添加到请求头部，除非显示地规定。

考点 17 对于 JSON 的理解。

难度系数: ★★☆☆☆

【腾讯网络媒体事业部】

网易专家解析:

JSON 是一种数据交换格式，基于文本，优于轻量，用于交换数据。它可以表示数字、布尔值、字符串、null、数组（值的有序序列），以及由这些值（或数组、对象）所组成的对象（字符串与值的映射）。JSON 使用 JavaScript 语法，但是 JSON 格式仅仅是一个文本。文本可以被任何



编程语言读取及作为数据格式传递。

考点 18 判断一个对象属于某个类的方法。

难度系数: ★☆☆☆☆

【百度 web 前端事业部 / 阿里社区电商事业部】

网易专家解析:

判断一个对象属于某个类的方法有以下三种:

- 1、使用 instanceof 运算符来判断构造函数的 prototype 属性是否出现在对象的原型链中的任何位置。
- 2、可以通过对象的 constructor 属性来判断，对象的 constructor 属性指向该对象的构造函数，但是这种方式不是很安全，因为 constructor 属性可以被改写。
- 3、如果需要判断的是某个内置的引用类型的话，可以使用 Object.prototype.toString() 方法来打印对象的[[Class]] 属性来进行判断。

考点 19 对于事件委托的理解。

难度系数: ★★☆☆☆

【网易雷火事业部】

网易专家解析:

事件委托是给父元素绑定事件，用来监听子元素的冒泡事件，并找道是那个子元素的事件。事件委托本质上是利用了浏览器事件冒泡的机制。因为事件在冒泡过程中会上传到父节点，并且父节点可以通过事件对象获取到目标节点，因此可以把子节点的监听函数定义在父节点上，由父节点的监听函数统一处理多个子元素的事件，这种方式称为事件代理。使用事件代理我们可以不需要为每一个子元素都绑定一个监听事件，这样减少了内存上的消耗。并且使用事件代理我们还可以实现事件的动态绑定，比如说新增了一个子节点，我们并不需要单独地为它添加一个监听事件，它所发生的事件会交给父元素中的监听函数来处理。



网易专家解析:

事件是用户操作网页时发生的交互动作或者网页本身的一些操作，现代浏览器有三种事件模型。

第一种事件模型是最早的 DOM0 级模型，这种模型不会传播，所以没有事件流的概念，但是现在有的浏览器支持以冒泡的方式实现，它可以在网页中直接定义监听函数，也可以通过 js 属性来指定监听函数。这种方式是所有浏览器都兼容的。

第二种事件模型是 IE 事件模型，在该事件模型中，一次事件共有两个过程，事件处理阶段，和事件冒泡阶段。事件处理阶段会首先执行目标元素绑定的监听事件。然后是事件冒泡阶段，冒泡指的是事件从目标元素冒泡到 document，依次检查经过的节点是否绑定了事件监听函数，如果有则执行。这种模型通过 attachEvent 来添加监听函数，可以添加多个监听函数，会按顺序依次执行。

第三种是 DOM2 级事件模型，在该事件模型中，一次事件共有三个过程，第一个过程是事件捕获阶段。捕获指的是事件从 document 一直向下传播到目标元素，依次检查经过的节点是否绑定了事件监听函数，如果有则执行。后面两个阶段和 IE 事件模型的两个阶段相同。这种事件模型，事件绑定的函数是 addEventListener，其中第三个参数可以指定事件是否在捕获阶段执行。



考点 21 对于 DOM 和 BOM 的理解。

难度系数:★★★☆☆

【网易互动娱乐事业部 / 腾讯技术工程事业部】

网易专家解析:

DOM 指的是文档对象模型，它指的是把文档当做一个对象来对待，这个对象主要定义了处理网页内容的方法和接口。

BOM 指的是浏览器对象模型，它指的是把浏览器当做一个对象来对待，这个对象主要定义了与浏览器进行交互的法和接口。BOM 的核心是 window，而 window 对象具有双重角色，它既是通过 js 访问浏览器窗口的一个接口，又是一个 Global (全局) 对象。这意味着在网页中定义的任何对象，变量和函数，都作为全局对象的一个属性或者方法存在。window 对象含有 location 对象、navigator 对象、screen 对象等子对象，并且 DOM 的最根本的对象 document 对象也是 BOM 的 window 对象的子对象。

考点 22 对于 undefined 与 undeclared 区别的理解。

难度系数: ★★★☆☆

【美团互联网云事业部 / 京东电子文娱事业部】

网易专家解析:

已在作用域中声明但还没有赋值的变量，是 undefined。相反，还没有在作用域中声明过的变量，是 undeclared。对于 undeclared 变量的引用，浏览器会报引用错误，如 ReferenceError: b is not defined。但是我们可以使用 typeof 的安全防范机制来避免报错，因为对于 undeclared (或者 not defined) 变量，typeof 会返回 "undefined"。



考点 23 对于 Cookie 隔离的理解。

难度系数: ★★★☆☆

【京东电子文娱事业部】

网易专家解析:

通过使用多个非主要域名来请求静态文件，如果静态文件都放在主域名下，那静态文件请求的时候带有的 cookie 的数据提交给 server 是非常浪费的，还不如隔离开。因为 cookie 有域的限制，因此不能跨域提交请求，故使用非主要域名的时候，请求头中就不会带有 cookie 数据，这样可以降低请求头的大小，降低请求时间，从而达到降低整体请求延时的目的。同时这种方式不会将 cookie 传入 server，也减少了 server 对 cookie 的处理分析环节，提高了 server 的 http 请求的解析速度。

考点 24 对于 JS 垃圾回收机制的理解。

难度系数: ★★☆☆☆

【百度 web 前端事业部 / 阿里社区电商事业部】

网易专家解析:

JavaScript 具有自动垃圾回收机制，垃圾回收器会定期对那些我们不再使用的变量、对象所占用的内存进行释放、回收。具体到浏览器中的实现，则通常有两个策略，分别为标记清除和引用计数：

1、标记清除：这个算法把“对象是否不再需要”简化定义为“对象是否可以获得”。这个算法假定设置一个叫做根（root）的对象（在 JavaScript 里，根是全局对象）。定期的，垃圾回收器将从根开始，找所有从根开始引用的对象，然后找这些对象引用的对象。从根开始，垃圾回收器将找到所有可以获得的对象和所有不能获得的对象。

2、引用计数：这是最简单的垃圾收集算法。此算法把“对象是否不再需要”简化定义为“对象有没有其他对象引用到它”。如果没有引用指向该对象（零引用），对象将被垃圾回收机制回收。该算法有个限制，无法处理循环引用。两个对象被创建，并互相引用，形成了一个循环。它们被



调用之后不会离开函数作用域，所以它们已经没有用了，可以被回收了。然而，引用计数算法考虑到它们互相都有至少一次引用，所以它们不会被回收。

考点 25 对于 XML 与 JSON 区别的理解。

难度系数: ★★☆☆☆

【腾讯微信事业部】

网易专家解析：

XML：扩展标记语言 (Extensible Markup Language, XML)，用于标记电子文件使其具有结构性的标记语言，可以用来标记数据、定义数据类型，是一种允许用户对自己的标记语言进行定义的源语言。

JSON：JSON(JavaScript Object Notation)一种轻量级的数据交换格式，具有良好的可读和便于快速编写的特性。可在不同平台之间进行数据交换。

XML 与 JSON 区别有以下四点：

- 1、数据体积方面，JSON 相对于 XML 来讲，数据的体积小，传递的速度更快些；
- 2、数据交互方面，JSON 与 JavaScript 的交互更加方便，更容易解析处理，更好的数据交互；
- 3、数据描述方面，JSON 对数据的描述性比 XML 较差；
- 4、输速度方面，JSON 的速度要远远快于 XML。



Vue

第二章

- 01 探索Vue的组件时间
- 02 深入Vue源码设计
- 03 Vue生态及实践
- 04 TypeScript基础篇
- 05 TypeScript提高篇
- 06 TypeScript实战
- 07 Vue3.x的设计理念
- 08 Vue4.x API设计

01 探索 Vue 的组件世界

考点 26 在 Vue 实际开发过程中直接给一个数组项赋值，Vue 不能检测到变化的原因。

难度系数: ★★★☆☆

【腾讯互动娱乐事业部 / 百度智能生活事业部】

网易专家解析:

由于 JavaScript 的限制，Vue 不能检测到数组的变动，当你利用索引直接设置一个数组项时，可以使用 Vue 中提供的 `Vue.set()` 方法，当你修改数组的长度时可以使用 Vue 中提供的 `Array.prototype.splice()` 方法。

考点 27 Vue 实际开发过程中 `keep-alive` 是很常用的组件，请谈一谈你对它的理解。

难度系数: ★★☆☆☆

【网易有道事业部 / 腾讯技术工程事业部】

网易专家解析:

`keep-alive` 是 Vue 内置的一个组件，可以使被包含的组件保留状态，避免重新渲染，简单从以下几个功能点描述：

- 1、一般会结合路由和动态组件一起使用，用于缓存组件；
- 2、提供 `include` 和 `exclude` 属性，二者都支持字符串或正则表达式，`include` 表示只有名称匹配的组件会被缓存，`exclude` 表示任何名称匹配的组件都不会被缓存，此外 `exclude` 的优先级比 `include` 高；
- 3、对应两个钩子函数 `activated` 和 `deactivated`，当组件被激活时，触发钩子函数 `activated`，当组件被移除时，触发钩子函数 `deactivated`。



考点 28 我们通常会借助 Vue-router 实现组件页面跳转，以下为路由模式的介绍。

难度系数: ★★☆☆☆

【百度信息技术工程事业部】

网易专家解析:

Vue-router 有 3 种路由模式: hash、history、abstract，以下是对这三种路由模式的理解：

1、hash: 使用 URL hash 值来作路由。支持所有浏览器，包括不支持 HTML5 History API 的浏览器；后面的 hash 值的变化，浏览器既不会向服务器发出请求，浏览器也不会刷新，每次 hash 值的变化会触发 hashchange 事件。

2、history: 依赖 HTML5 History API 和服务器配置。具体可以查看 HTML5 History 模式；利用了 HTML5 中新增的 pushState() 和 replaceState() 方法。这两个方法应用于浏览器的历史记录栈，在当前已有的 back、forward、go 的基础之上，它们提供了对历史记录进行修改的功能。只是当它们执行修改时，虽然改变了当前的 URL，但浏览器不会立即向后端发送请求。

3、Abstract: 支持所有 JavaScript 运行环境。



02 深入 Vue 源码设计

考点 29 关于虚拟 DOM 的实现原理。

难度系数: ★☆☆☆☆

【腾讯微信事业部 / 百度移动生态事业部】

网易专家解析:

虚拟 DOM 可以看作是一个使用 JavaScript 模拟了 DOM 结构的树形结构，这个树结构包含整个 DOM 结构的信息，虚拟 DOM 的实现原理：用 JavaScript 对象模拟真实 DOM 树对真实 DOM 进行抽象；用 Diff 算法 — 比较两棵虚拟 DOM 树的差异；运用 patch 算法 — 将两个虚拟 DOM 对象的差异应用到真正的 DOM 树。

考点 30 Vue 组件中的 data 必须是一个函数的原因。

难度系数: ★★☆☆☆

【字节跳动互动娱乐事业部】

网易专家解析:

如果 data 是一个函数的话，这样每复用一次组件，就会返回一份新的 data，类似于给每个组件实例创建一个私有的数据空间，让各个组件实例维护各自的数据。而单纯的写成对象形式，就使得所有组件实例共用了一份 data，就会造成一个变了全都会变的结果。所以说 Vue 组件的 data 必须是函数，而这都是因为 JavaScript 的特性带来的。



网易专家解析:

Vue 常用修饰符有以下五种:

- 1、.stop: 阻止点击事件冒泡, 等同于 JavaScript 中的 event.stopPropagation();
- 2、.prevent: 防止执行预设的行为;
- 3、.capture: 与事件冒泡的方向相反, 事件捕获由外到内, 捕获事件: 嵌套两三层父子关系, 然后所有都有点击事件, 点击子节点, 就会触发从外至内 父节点-> 子节点的点击事件;
- 4、.self: 只会触发自己范围内的事件, 不包含子元素;
- 5、.once: 只执行一次。



03 Vue 生态及实践

考点 32 关于 VueRouter 之 query 与 params 两种传参区别的理解。

难度系数: ★★★☆☆

【网易雷火事业部】

网易专家解析:

query 与 params 区别有: query 在刷新页面时参数不会消失, 而 params 刷新页面时参数会消失, 可以考虑本地存储解决; query 传得参数会显示在 url 地址栏当中, 而 params 传参不会显示在地址栏。以下是这两种语法的传参样例:

query 语法:

```
this.$router.push({path:"地址",query:{id:"123"}}); 这是传递参数;  
this.$route.query.id; 这是接受参数。
```

params 语法:

```
this.$router.push({name:"地址",params:{id:"123"}}); 这是传递参数;  
this.$route.params.id; 这是接受参数。
```

考点 33 关于\$route 和\$router 区别的理解。

难度系数: ★★★☆☆☆

【百度搜索研发事业部】

网易专家解析:

对\$route 的理解:

route 是路由信息对象, 里面主要包含路由的一些基本信息, 包括 name、meta、path、hash、query、params、fullPath、redirectedFrom 等。

对\$router 的理解:



router 是 VueRouter 的实例，包含了一些路由的跳转方法，钩子函数等。

考点 34 对于 computed 和 watch 使用场景的介绍。

难度系数: ★★★☆☆

【网易互动娱乐事业部 / 腾讯技术工程事业部】

网易专家解析：

computed:

当一个属性受多个属性影响的时候就需要用到 computed。

使用场景：购物车结算的时候。

watch:

当一条数据影响多条数据的时候就需要用到 watch。

使用场景：搜索数据。



04 TypeScript 基础篇

考点 35 关于 TypeScript 的描述与理解。

难度系数: ★☆☆☆☆

【小米互动娱乐事业部 / 美团互联网云事业部】

网易专家解析:

TypeScript 是一种由微软开发和维护的免费开源编程语言。它是一个强类型的 JavaScript 超集，可编译为纯 JavaScript。它是一种用于应用级 JavaScript 开发的语言。对于熟悉 c#、Java 和所有强类型语言的开发人员来说，TypeScript 非常容易学习和使用。

TypeScript 可以在任何浏览器、主机和操作系统上执行。TypeScript 不是直接在浏览器上运行的。它需要一个编译器来编译和生成 JavaScript 文件。TypeScript 是带有一些附加特性的 ES6 JavaScript 版本。

考点 36 对于 TypeScript 中类的理解以及其相关特性。

难度系数: ★★★☆☆

【百度智能生活事业部】

网易专家解析:

TypeScript 是一种面向对象的 JavaScript 语言，支持 OOP 编程特性，比如类、接口等。与 Java 一样，类是用于创建可重用组件的基本实体。它是一组具有公共属性的对象。类是创建对象的模板或蓝图。它是一个逻辑实体。“class”关键字用于在 TypeScript 中声明一个类。类的特征有：继承、封装、多态性、抽象等。



05 TypeScript 提高篇

考点 37 关于 TypeScript 优点的介绍。

难度系数: ★★★☆☆

【阿里社区电商事业部】

网易专家解析:

使用 TypeScript 有以下优点:

- 1、它提供了可选静态类型的优点。在这里，TypeScript 提供了可以添加到变量、函数、属性等的类型；
- 2、TypeScript 能够编译出一个能在所有浏览器上运行的 JavaScript 版本；
- 3、TypeScript 总是在编译时强调错误，而 JavaScript 在运行时指出错误；
- 4、TypeScript 支持强类型或静态类型，而这不是在 JavaScript 中；
- 5、它有助于代码结构；
- 6、它使用基于类的面向对象编程；
- 7、它提供了优秀的工具支持和智能感知，后者在添加代码时提供活动提示；
- 8、它通过定义模块来定义名称空间概念。



考点 38 关于 TypeScript 缺点的介绍。

难度系数: ★★★☆☆

【阿里社区电商事业部 / 百度信息技术工程部】

网易专家解析:

使用 TypeScript 有以下缺点:

- 1、TypeScript 需要很长时间来编译代码；
- 2、TypeScript 不支持抽象类；
- 3、如果我们在浏览器中运行 TypeScript 应用程序，需要一个编译步骤将 TypeScript 转换成 JavaScript；
- 4、Web 开发人员使用了几十年的 JavaScript，而 TypeScript 不是都是新东西；
- 5、要使用任何第三方库，必须使用定义文件。并不是所有第三方库都有可用的定义文件；
- 6、类型定义文件的质量是一个问题，即如何确保定义是正确的。



06 TypeScript 实战

考点 39 在 TypeScript 中创建变量的方式。

难度系数: ★★★☆☆

【腾讯网络媒体事业部】

网易专家解析:

我们可以通过以下四种方式之一声明一个变量：

- 1、在一条语句中声明类型和值。语法: var [identifier]: [type-annotation] = value;
- 2、声明没有值的类型。语法: var [identifier]: [type-annotation];
- 3、在没有类型的情况下声明它的值。语法: var [identifier] = value;
- 4、声明没有值和类型。语法: var(标识符)。

考点 40 对于 TypeScript 映射文件的理解。

难度系数: ★★★☆☆

【字节跳动互动娱乐事业部 / 百度 web 前端研发部】

网易专家解析:

TypeScript Map 文件是一个源映射文件，其中包含有关我们原始文件的信息；.map 文件是源映射文件，可让工具在发出的 JavaScript 代码和创建它的 TypeScript 源文件之间进行映射；许多调试器可以使用这些文件，因此我们可以调试 TypeScript 文件而不是 JavaScript 文件。



07 Vue3.x 的设计理念

考点 41 Vue 中子组件调用父组件的方法。

难度系数: ★★★☆☆

【网易有道事业部 / 京东电子文娱事业部】

网易专家解析:

Vue 中子组件调用父组件的方法有以下三种:

- 1、直接在子组件中通过 `this.$parent.event` 来调用父组件的方法;
- 2、在子组件里用`$emit` 向父组件触发一个事件，父组件监听这个事件;
- 3、父组件把方法传入子组件中，在子组件里直接调用这个方法。

考点 42 关于 Vue 优缺点的介绍。

难度系数: ★★★☆☆

【阿里云智能事业部】

网易专家解析:

Vue 的优点:

- 1、数据驱动视图：对真实 dom 进行抽象出 virtual dom (本质就是一个 js 对象)，并配合 Diff 算法、响应式和观察者、异步队列等手段以最小代价更新 dom，渲染页面；
- 2、组件化：组件用单文件的形式进行代码的组织编写，使得我们可以在一个文件里编写 html、css (scoped 属性配置 css 隔离)、js 并且配合 Vue-loader 之后，支持更强大的预处理器等功能；
- 3、强大且丰富的 API 提供一系列的 api 能满足业务开发中各类需求；
- 4、生命周期钩子函数，选项式的代码组织方式，但仍然有优化空间 (Vue3 composition-api)；
- 5、生态好，社区活跃，较为友好，易复用。



Vue 的缺点：

- 1、由于底层基于 Object.defineProperty 实现响应式，所以这个 api 本身不支持 IE8 及以下浏览器；
- 2、csr 的先天不足，首屏性能问题，比如：容易出现白屏；
- 3、由于百度等搜索引擎爬虫无法爬取 js 中的内容，故 spa 先天就对 seo 优化心有余力不足。



08 Vue4.x API 设计

考点 43 对于 Vue 生命周期的理解。

难度系数: ★★★☆☆

【腾讯技术工程事业部 / 小米互动娱乐事业部】

网易专家解析:

生命周期就是 Vue 从开始创建到销毁的过程，分为四大步（创建，挂载，更新，销毁），每一步又分为两小步，如 beforeCreate, created。beforeCreate 前，也就是 new Vue 的时候会初始化事件和生命周期；beforeCreate 和 created 之间会挂载 Data，绑定事件；接下来会根据 el 挂载页面元素，如果没有设置 el 则生命周期结束，直到手动挂载；el 挂载结束后，根据 template/outerHTML(el) 渲染页面；在 beforeMount 前虚拟 DOM 已经创建完成；之后在 mounted 前，将 vm.\$el 替换掉页面元素 el；mounted 将虚拟 dom 挂载到真实页面（此时页面已经全部渲染完成）；之后发生数据变化时触发 beforeUpdate 和 updated 进行一些操作；最后主动调用销毁函数或者组件自动销毁时 beforeDestroy，手动撤销监听事件，计时器等；destroyed 时仅存在 Dom 节点，其他所有东西已自动销毁。



考点 44 关于 Vue 和 React 的区别与使用场景的介绍。

难度系数:★★★☆☆

【百度 AI 技术平台事业部】

网易专家解析:

React 框架特点如下:

- 1、灵活性和响应性：它提供最大的灵活性和响应能力。虚拟 DOM：由于它基于文档对象模型，因此它允许浏览器友好地以 HTML, XHTML 或 XML 格式排列文档；
- 2、丰富的 JavaScript 库：来自世界各地的贡献者正在努力添加更多功能；
- 3、可扩展性：由于其灵活的结构和可扩展性，React 已被证明对大型应用程序更好；
- 4、不断发展：React 得到了 Facebook 专业开发人员的支持，他们不断寻找改进方法。Web 或移动平台：React 提供 React Native 平台，可通过相同的 React 组件模型为 iOS 和 Android 开发本机呈现的应用程序。无论是 Web 还是本机移动开发，React 都是大多数用户界面设计平台的理想选择。

Vue 框架特点如下：

- 1、易于使用：Vue.js 包含基于 HTML 的标准模板，可以更轻松地使用和修改现有应用程序；
- 2、更顺畅的集成：无论是单页应用程序还是复杂的 Web 界面，Vue.js 都可以更平滑地集成更小的部件，而不会对整个系统产生任何影响；
- 3、更好的性能，更小的尺寸：它占用更少的空间，并且往往比其他框架提供更好的性能；
- 4、精心编写的文档：通过详细的文档提供简单的学习曲线，前端基础铺垫后即可进行学习；
- 5、适应性：整体声音设计和架构使其成为一种流行的 JavaScript 框架。它提供无障碍的迁移，简单有效的结构和可重用的模板。



考点 45 在 Vue 项目中引入第三方库（比如 jQuery）的方法。

难度系数:★★★☆☆

【快手商业化事业部】

网易专家解析:

方法一：绝对路径直接引入。

在 index.html 中用 script 引入

```
<script src="./static/jquery-1.12.4.js"></script>
```

然后在 webpack 中配置 external externals: { 'jquery': 'jQuery' } 在组件中使用时 import
import \$ from 'jquery'.

方法二：在 webpack 中配置 alias。

```
resolve: { extensions: ['.js', '.Vue', '.json'], alias: { '@': resolve('src'), 'jquery':  
    resolve('static/jquery-1.12.4.js') } }
```

然后在组件中 import。

方法三：在 webpack 中配置 plugins。

```
plugins: [ new webpack.ProvidePlugin({ $: 'jquery' }) ]
```

全局使用，但在使用 eslint 情况下会报错，需要在使用了 \$ 的代码前添加 /* eslint-disable */ 来
去掉 ESLint 的检查。



React

第三章

- 01 步入React前厅
- 02 探索React正殿
- 03 React生态应用
- 04 React原理
- 05 React状态管理
- 06 React高级实践和性能优化

01 步入 React 前厅

考点 46 请简单描述 React 中的 Refs。

难度系数: ★★★☆☆

【百度信息技术工程事业部】

网易专家解析:

React 中的 Refs 简单描述如下:

- 1、Refs 是使用 React.createRef() 方法创建的，并通过 Ref 属性添加到 React 元素上；
- 2、为了在整个组件中使用 Refs，只需将 Ref 分配给构造函数中的实例属性；
- 3、可以为元素添加 Ref 属性然后在回调函数中接受该元素在 Dom 树中的句柄，该值会作为回调函数的第一个参数返回；
- 4、Refs 并不是类组件的专属，函数式组件同样能够利用闭包暂存其值。

考点 47 请简单描述 React 事件处理。

难度系数: ★★★☆☆

【网易有道事业部 / 腾讯技术工程事业部】

网易专家解析:

React 中的事件处理逻辑如下:

- 1、React 中事件的绑定是直接写在 JSX 元素上的，不需要通过 AddEventListener 事件委托的方式进行监听；
- 2、为了解决跨浏览器兼容性问题，React 中的事件处理程序将传递 SyntheticEvent 实例；
- 3、React 实际上并未将事件附加到子节点本身，React 将使用单个事件侦听器在顶层侦听所有事件；
- 4、React 在更新 Dom 时无需担心跟踪事件监听器。



考点 48 请简单描述 React 中的 JSX。

难度系数: ★★★☆☆

【腾讯互动娱乐事业部 / 百度智能生活事业部】

网易专家解析:

React 中的 JSX 描述如下:

- 1、JSX 即 JavaScript XML，是一种在 React 组件内部构建标签的类 XML 语法。
- 2、React 在不使用 JSX 的情况下一样可以工作，然而使用 JSX 可以提高组件的可读性，因此推荐使用 JSX。
- 3、提供更加语义化且移动的标签；程序结构更容易被直观化；
- 4、抽象了 React Element 的创建过程；可以随时掌控 HTML 标签以及生成这些标签的代码。



02 探索 React 正殿

考点 49 请简单描述 React 中的状态及其使用方式。

难度系数:★★☆☆☆

【腾讯微信事业部 / 百度移动生态事业部】

网易专家解析:

状态是 React 组件的核心是数据的来源，必须尽可能简单。基本上状态是确定组件呈现和行为的对象；与 Props 不同，它们是可变的，并创建动态和交互式组件；可以通过 `this.State()` 访问它们；当状态值发生变化时，React 会自动调用 `Render()` 方法重新渲染页面；不能直接对 State 的值进行修改，需要调用 `this.setState()` 完成。

考点 50 请简单描述 React 中的 Context。

难度系数:★★☆☆☆

【字节跳动互动娱乐事业部】

网易专家解析:

当实际开发过程中不想在组件树中通过逐层传递 Props 或者 State 的方式来传递数据时，可以使用 Context 来实现 跨层级 的组件数据传递。使用 Props 或者 State 传递数据，数据自顶下流，而使用 Context，可以跨越组件进行数据传递。



03 React 生态应用

考点 51 请简单描述 React 中的 Key 值的作用。

难度系数: ★★★☆☆

【百度搜索研发事业部】

网易专家解析:

React 中 Key 值的作用简单描述如下:

- 1、React 在创建 Key 值相同的组件时，遇到 Key 值相同的，不会重新创建；
- 2、有了 Key 值之后，Key 值与组件就会形成一种对应的关系，React 会根据 Key 值来决定组件的创建销毁或者更新；
- 3、如果 Key 值相同，组件的属性发生了变化，这时组件不会被销毁，只会对相应的属性进行更新；
- 4、如果 Key 值不同，那么组件会先被销毁，之后再重新创建。



04 React 原理

考点 52 请简单描述简述 Flux 思想。

难度系数: ★★★☆☆

【百度智能生活事业部】

网易专家解析:

Flux 的最大特点，就是数据的"单向流动"，流程如下：

- 1、用户访问 View；
- 2、View 发出用户的 Action；
- 3、Dispatcher 收到 Action，要求 Store 进行相应的更新；
- 4、Store 更新后，发出一个"change"事件；
- 5、View 收到"change"事件后，更新页面。



05 React 状态管理

考点 53 请简单描述 Redux 遵循的原则。

难度系数: ★★★☆☆

【阿里社区电商事业部 / 百度信息技术工程部】

网易专家解析:

Redux 遵循的原则简单描述如下：

- 1、单一事实来源：整个应用的状态存储在单个 Store 中的对象/状态树里，单一状态树可以更容易地跟踪随时间的变化，并调试或检查应用程序；
- 2、状态是只读的：改变状态的唯一方法是去触发一个动作，动作是描述变化的普通 JS 对象。类似于 State 是数据的最小表示，该操作是对数据更改的最小表示；
- 3、使用纯函数进行更改：为了指定状态树如何通过操作进行转换，需要使用纯函数。纯函数是那些返回值仅取决于其参数值的函数。

考点 54 请简单描述 React 组件中如何做事件代理。

难度系数: ★★★☆☆

【百度 AI 技术平台事业部】

网易专家解析:

- 1、React 基于 Virtual Dom 实现了一个 SyntheticEvent 层（合成事件层），定义的事件处理器会接收到一个合成事件对象的实例，它符合 W3C 标准，且与原生的浏览器事件拥有同样的接口，支持冒泡机制，所有的事件都自动绑定在最外层上；
- 2、事件委派：React 会把所有的事件绑定到结构的最外层，使用统一的事件监听器，这个事件监听器上维持了一个映射来保存所有组件内部事件监听和处理函数；
- 3、自动绑定：React 组件中，每个方法的上下文都会指向该组件的实例，即自动绑定 this 为当前组件。



06 React 高级实践和性能优化

考点 55 请简单描述对 Time Slice 的理解。

难度系数: ★★★☆☆

【腾讯云与智慧产业事业部】

网易专家解析:

Time Slice 时间分片:

如果任务不能在 50 毫秒内执行完，那么为了不阻塞主线程，这个任务应该让出主线程的控制权，使浏览器可以处理其他任务。让出控制权意味着停止执行当前任务，让浏览器去执行其他任务，随后再回来继续执行没有执行完的任务。所以时间切片的目的是不阻塞主线程，而实现目的的技术手段是将一个长任务拆分成很多个不超过 50ms 的小任务分散在宏任务队列中执行。



Node.js

第四章

01 Node.js基础

02 Web服务及Koa

03 企业级Node.js框架

01 Node.js 基础

考点 56 对于 Node.js 和 Javascript 区别的理解。

难度系数:★★☆☆☆

【百度移动互联网事业部 / 阿里云智能事业部】

网易专家解析:

- 1、 JavaScript 是客户端编程语言，需要浏览器的 JavaScript 解释器进行解释执行；
- 2、 node.js 是一个基于 Chrome JavaScript 运行时建立的平台，它是对 Google V8 引擎进行了封装的运行环境；
- 3、 简单的说 node.js 就是把浏览器的解释器封装起来作为服务器运行平台；
- 4、 用类似 JavaScript 的结构语法进行编程，在 node.js 上运行。

考点 57 运用 Node.js 的情况。

难度系数:★★☆☆☆

【腾讯云与智慧产业事业部】

网易专家解析:

Node.js 是异步的、事件驱动的、非阻塞的和单线程的，使得它成为开发以下应用程序的完美候选：

- 1、 实时应用程序，如聊天和提供实时更新的应用程序；
- 2、 将视频或其他多媒体内容流式传输给大量观众的流式应用程序；
- 3、 其他 I/O 密集型应用程序，如协作平台；
- 4、 遵循微服务架构的网络后端。



考点 58 对于 require 模块加载机制的理解。

难度系数: ★★★☆☆

【腾讯微信事业部】

网易专家解析:

require 的模块加载机制分为以下四步:

- 1、先计算模块路径;
- 2、如果模块在缓存里面，取出缓存;
- 3、加载模块;
- 4、输出模块的 exports 属性即可。

考点 59 对于 Node 事件循环流程的理解。

难度系数: ★★☆☆☆

【字节跳动互动娱乐事业部 / 百度 web 前端研发部】

网易专家解析:

在进程启动时，Node 便会创建一个类似于 while(true)的循环，每执行一次循环体的过程我们成为 Tick。每个 Tick 的过程就是查看是否有事件待处理，如果有就取出事件及其相关的回调函数，然后进入下一个循环，如果不再有事件处理，就退出进程。

考点 60 在每个 tick 的过程中，判断是否有事件需要处理的方法。

难度系数: ★★★☆☆

【腾讯网络媒体事业部 / 阿里产业电商事业部】

网易专家解析:

- 1、每个事件循环中有一个或者多个观察者，而判断是否有事件需要处理的过程就是向这些观察者询问是否有要处理的事件。
- 2、在 Node 中，事件主要来源于网络请求、文件的 I/O 等，这些事件对应的观察者有文件 I/O 观察者，网络 I/O 的观察者。



3、事件循环是一个典型的生产者/消费者模型。异步 I/O，网络请求等则是事件的生产者，源源不断地为 Node 提供不同类型的事件，这些事件被传递到对应的观察者那里，事件循环则从观察者那里取出事件并处理。

4、在 Windows 下，这个循环基于 IOCP 创建，在*nix 下则基于多线程创建。



02 Web 服务及 Koa

考点 61 V8 的内存限制的了解以及这样设计的原因。

难度系数: ★★★☆☆

【字节跳动互动娱乐事业部 / 字节跳动互动娱乐事业部】

网易专家解析:

64 位系统下是 1.4GB， 32 位系统下是 0.7GB。因为 1.5GB 的垃圾回收堆内存，V8 需要花费 50 毫秒以上，做一次非增量式的垃圾回收甚至要 1 秒以上。这是垃圾回收中引起 JavaScript 线程暂停执行的事件，在这样的花销下，应用的性能和影响力都会直线下降。

考点 62 对于事件监听的理解。

难度系数: ★★★☆☆

【网易有道事业部 / 京东电子文娱事业部】

网易专家解析:

Node.js 的事件监听也可能出现的内存泄漏。例如对同一个事件重复监听，忘记移除 (removeListener)，将造成内存泄漏。这种情况很容易在复用对象上添加事件时出现，所以事件重复监听可能收到如下警告：emitter.setMaxListeners() to increase limit。

例如，Node.js 中 Agent 的 keepAlive 为 true 时，可能造成的内存泄漏。当 Agent keepAlive 为 true 的时候，将会复用之前使用过的 socket，如果在 socket 上添加事件监听，忘记清除的话，因为 socket 的复用，将导致事件重复监听从而产生内存泄漏。

原理上与前一个添加事件监听的时候忘了清除是一样的。在使用 Node.js 的 http 模块时，不通过 keepAlive 复用是没有问题的，复用了以后就会可能产生内存泄漏。所以，你需要了解添加事件监听的对象的生命周期，并注意自行移除。



03 企业级 Node.js 框架

考点 63 webSocket 与传统的 http 相比的优势。

难度系数:★★★☆☆

【网易有道事业部 / 阿里产业电商事业部】

网易专家解析:

WebSocket 与传统的 http 相比的优势有以下三点:

- 1、客户端与服务器只需要一个 TCP 连接，比 http 长轮询使用更少的连接；
- 2、WebSocket 服务端可以推送数据到客户端；
- 3、更轻量的协议头，减少数据传输量。

考点 64 spawn 在创建子进程的时候，第三个参数有一个 stdio 选项，对于这个选项的作用与默认的值的理解。

难度系数:★★★☆☆

【百度移动互联网事业部 / 字节跳动互动娱乐事业部】

网易专家解析:

- 1、选项用于配置在父进程和子进程之间建立的管道。
- 2、默认情况下，子进程的 stdin、 stdout 和 stderr 会被重定向到 ChildProcess 对象上相应的 subprocess.stdin、subprocess.stdout 和 subprocess.stderr 流。
- 3、这相当于将 options.stdio 设置为 ['pipe', 'pipe', 'pipe']。



考点 65 对于 libuv 的理解。

难度系数: ★★★☆☆

【百度移动互联网事业部】

网易专家解析:

Libuv 是 Node.js 的多平台支持库，主要用于异步 I/O。它主要是为 Node.js 开发的，随着时间的推移，它被广泛用于其他系统，如 Luvit、pyuv、Julia 等。Libuv 基本上是对依赖于平台的 libev/IOCP 的抽象，为用户提供基于 libev 的 API。Libuv 的一些重要特性是：

- 1、 支持全功能事件循环；
- 2、 文件系统事件；
- 3、 异步文件和文件系统操作；
- 4、 异步 TCP 和 UDP 套接字；
- 5、 子进程。

考点 66 Node.js 处理子线程的方法。

难度系数: ★★★☆☆

【阿里产业电商事业部 / 腾讯云与智慧产业事业部】

网易专家解析:

一般来说，Node.js 是一个单线程进程，不暴露子线程或线程管理方法。但是仍然可以使用 spawn() 将子线程用于某些特定的异步 I/O 任务，这些任务在后台执行并且通常不执行任何 JS 代码或阻碍应用程序中的主事件循环。如果仍然想在应用程序中使用线程概念，必须明确地包含一个名为 ChildProcess 的模块。



考点 67 在 Node.js 上下文中解释 REPL。

难度系数: ★★★☆☆

【腾讯微信事业部 / 百度移动生态事业部】

网易专家解析:

REPL 在 Node.js 的代表: R-Read, E-Eval, P-Print, L-Loop。它代表一个计算机环境, 例如窗口控制台或 Unix/Linux shell, 可以在其中输入任何命令, 然后系统可以输出响应。默认情况下, Node.js 与 REPL 环境捆绑在一起。REPL 可以执行以下列出的任务:

Read: 读取用户的输入, 将其解析为 JavaScript 数据结构, 然后将其存储在内存中;

Eval: 接收并评估数据结构;

Print: 打印最终结果;

Loop: 循环提供的命令, 直到按两次 CTRL+C;

考点 68 解释 Node.js 中间件概念。

难度系数: ★★★☆☆

【百度移动互联网事业部 / 字节跳动互动娱乐事业部】

网易专家解析:

一般来说, 中间件是一个接收请求和响应对象的函数。换句话说, 在应用程序的请求-响应循环中, 这些函数可以访问各种请求和响应对象以及循环的下一个函数。中间件的 next 功能是借助一个变量来表示的, 通常命名为 next。中间件功能最常执行的任务是:

- 1、执行任何类型的代码;
- 2、更新或修改请求和响应对象;
- 3、完成请求-响应循环;
- 4、调用堆栈中的下一个中间件。



考点 69 解释控制流的工作。

难度系数: ★★☆☆☆

【腾讯微信事业部】

网易专家解析:

在 Node.js 中，控制流函数基本上是在异步函数调用之间执行的代码。以下是执行它必须遵循的步骤：

- 首先，必须控制执行顺序；
- 然后，需要收集所需的数据；
- 其次，必须限制并发；
- 完成后，必须调用程序的下一步。



工程化

第五章

01 构建艺术

02 持续集成与部署

03 Git操作

01 构建艺术

考点 70 请简单描述 Bundle, Chunk, Module。

难度系数: ★★☆☆☆

【腾讯互动娱乐事业部 / 百度智能生活事业部】

网易专家解析:

简单描述如下:

- 1、Bundle: 是由 WebPack 打包出来的文件;
- 2、Chunk: 代码块, 一个 Chunk 由多个模块组合而成, 用于代码的合并和分割;
- 3、Module: 是开发中的单个模块, 在 WebPack 的世界, 一切皆模块, 一个模块对应一个文件,

WebPack 会从配置的 entry 中递归开始找出所有依赖的模块。



02 持续集成与部署

考点 71 请简单描述常见的 Plugin。

难度系数: ★★☆☆☆

【腾讯微信事业部 / 百度移动生态事业部】

网易专家解析:

常见的 Plugin 有以下六种:

- 1、Define-Plugin: 定义环境变量;
- 2、Html-WebPack-Plugin: 简化 Html 文件创建;
- 3、Uglifyjs-WebPack-Plugin: 通过 UglifyES 压缩 ES6 代码;
- 4、WebPack-parallel-uglify-Plugin: 多核压缩, 提高压缩速度;
- 5、WebPack-Bundle-analyzer: 可视化 WebPack 输出文件的体积;
- 6、Mini-Css-Extract-Plugin: Css 提取到单独的文件中, 支持按需加载。



考点 72 请简单描述常见的 Loader。

难度系数:★★★☆☆

【腾讯网络媒体事业部 / 阿里产业电商事业部】

网易专家解析:

常见的 Loader 有以下八种:

- 1、File-Loader: 把文件输出到一个文件夹中，在代码中通过相对 Url 去引用输出的文件；
- 2、Url-Loader: 和 File-Loader 类似，但是能在文件很小的情况下以 base64 的方式把文件内容注入到代码中去；
- 3、Source-map-Loader: 加载额外的 Source Map 文件，以方便断点调试；
- 4、Image-Loader: 加载并且压缩图片文件；
- 5、Babel-Loader: 把 ES6 转换成 ES5；
- 6、Css-Loader: 加载 Css，支持模块化、压缩、文件导入等特性；
- 7、Style-Loader: 把 Css 代码注入到 JavaScript 中，通过 DOM 操作去加载 Css；
- 8、Eslint-Loader: 通过 Eslint 检查 JavaScript 代码。



03 Git 操作

考点 73 请简单描述 Git 中的“裸存储库”。

难度系数:★★★☆☆

【百度搜索研发事业部】

网易专家解析:

Git 中的“裸”存储库只包含版本控制信息而没有工作文件（没有工作树），并且它不包含特殊的 .Git 子目录。相反，它直接在主目录本身包含 .Git 子目录中的所有内容，其中工作目录包括：

- 1、一个 .Git 子目录，其中包含你的仓库所有相关的 Git 修订历史记录；
- 2、工作树，或签出的项目文件的副本。

考点 74 请简单描述 Git Config 的功能。

难度系数:★★☆☆☆

【网易雷火事业部】

网易专家解析:

Git Config 的功能描述如下：

- 1、Git 使用你的用户名将提交与身份相关联。Git Config 命令可用来更改你的 Git 配置，包括你的用户名；

使用如下：

- 1、Git Config –global user.name "Your Name": 此命令将添加用户名；
- 2、Git Config –global user.email "Your E-mail Address": 此命令将添加电子邮件 ID。



考点 75 请简单描述 Git Stash 的作用。

难度系数: ★★★☆☆

【美团互联网云事业部】

网易专家解析:

Git Stash 使用场景如下:

通常情况下, 当你一直在处理项目的某一部分时, 如果你想要在某个时候切换分支去处理其他事情, 事情会处于混乱的状态。问题是, 你不想把完成了一半的工作的提交, 以便你以后就可以回到当前的工作。解决这个问题的答案是 Git Stash。

Git Stash 描述如下:

Stash 会将你的工作目录, 即修改后的跟踪文件和暂存的更改保存在一堆未完成的更改中, 你可以随时重新应用这些更改。

考点 76 请简单描述 Git Pull 和 Git Fetch 的区别。

难度系数: ★★★☆☆

【百度智能生活事业部】

网易专家解析:

二者都是在实际开发中很常见的命令, 区别如下:

Git Pull 命令从中央存储库中提取特定分支的新更改或提交, 并更新本地存储库中的目标分支。

Git Fetch 也用于相同的目的, 但它的工作方式略有不同。当你执行 Git Fetch 时, 它会从所需的分支中提取所有新提交, 并将其存储在本地存储库中的新分支中。如果要在目标分支中反映这些更改, 必须在 Git Fetch 之后执行 Git merge。只有在对目标分支和获取的分支进行合并后才会更新目标分支。



计算机网络

第六章

- 01 网络协议
- 02 网络请求实战
- 03 网络安全与攻防
- 04 浏览器状态同步和路由
- 05 工具链和其他

01 网络协议

考点 77 对于 IP 地址分类的理解。

难度系数: ★★☆☆☆

【百度搜索研发事业部】

网易专家解析:

一般可以这么认为, IP 地址=网络号+主机号。

网络号: 它标志主机所连接的网络地址表示属于互联网的哪一个网络。

主机号: 它标志主机地址表示其属于该网络中的哪一台主机。

IP 地址分为 A, B, C, D, E 五大类:

- 1、A 类地址(1~126): 以 0 开头, 网络号占前 8 位, 主机号占后面 24 位;
- 2、B 类地址(128~191): 以 10 开头, 网络号占前 16 位, 主机号占后面 16 位;
- 3、C 类地址(192~223): 以 110 开头, 网络号占前 24 位, 主机号占后面 8 位;
- 4、D 类地址(224~239): 以 1110 开头, 保留位多播地址;
- 5、E 类地址(240~255): 以 11110 开头, 保留位为将来使用。

考点 78 对于五层体系结构的理解。

难度系数: ★★☆☆☆

【腾讯互动娱乐事业部 / 百度智能生活事业部】

网易专家解析:

包含以下五层体系结构:

- 1、应用层: 对应于 OSI 参考模型的 (应用层、表示层、会话层)。
- 2、传输层: 对应 OSI 参考模型的的传输层。
- 3、网络层: 对应 OSI 参考模型的的网络层。
- 4、数据链路层: 对应 OSI 参考模型的的数据链路层。



5、物理层：对应 OSI 参考模型的物理层。

考点 79 对于 TCP/IP 四层模型的理解。

难度系数：★★☆☆☆

【网易有道事业部 / 腾讯技术工程事业部】

网易专家解析：

TCP/IP 包含以下四层模型：

- 1、应用层：对应于 OSI 参考模型的（应用层、表示层、会话层）。
- 2、传输层：对应 OSI 的传输层，为应用层实体提供端到端的通信功能，保证了数据包的顺序传送及数据的完整性。
- 3、网际层：对应于 OSI 参考模型的网络层，主要解决主机到主机的通信问题。
- 4、网络接口层：与 OSI 参考模型的数据链路层、物理层对应。



02 网络请求实战

考点 80 对于 HTTP 原理与请求的过程的理解。

难度系数:★★★☆☆

【美团互联网云事业部】

网易专家解析:

HTTP 是一个基于 TCP/IP 协议来传递数据的超文本传输协议，传输的数据类型有 HTML，图片等。以下为 HTTP 请求过程：

- 1、客户端进行 DNS 域名解析，得到对应的 IP 地址；
- 2、根据这个 IP，找到对应的服务器建立连接（三次握手）；
- 3、建立 TCP 连接后发起 HTTP 请求（一个完整的 HTTP 请求报文）；
- 4、服务器响应 HTTP 请求，客户端得到 HTML 代码；
- 5、客户端解析 HTML 代码，用 HTML 代码中的资源(如 js, css, 图片等等)渲染页面；
- 6、服务器关闭 TCP 连接（四次挥手）。



03 网络安全与功防

考点 81 对于 DOS、DDOS、DRDOS 攻击的理解。

难度系数:★★★☆☆

【阿里社区电商事业部】

网易专家解析:

- 1、DOS: (Denial of Service), 翻译过来就是拒绝服务，一切能引起 DOS 行为的攻击都被称为 DOS 攻击。最常见的 DOS 攻击就有计算机网络宽带攻击、连通性攻击。
- 2、DDOS: (Distributed Denial of Service), 翻译过来是分布式拒绝服务。是指处于不同位置的多个攻击者同时向一个或几个目标发动攻击，或者一个攻击者控制了位于不同位置的多台机器并利用这些机器对受害者同时实施攻击。常见的 DDOS 有 SYN Flood、Ping of Death、ACK Flood、UDP Flood 等。
- 3、DRDOS: (Distributed Reflection Denial of Service), 中文是分布式反射拒绝服务，该方式靠的是发送大量带有被害者 IP 地址的数据包给攻击主机，然后攻击主机对 IP 地址源做出大量回应，从而形成拒绝服务攻击。



04 浏览器状态同步和路由

考点 82 对称加密与非对称加密的理解。

难度系数: ★★☆☆☆

【阿里社区电商事业部 / 百度信息技术工程部】

网易专家解析:

对称加密：指加密和解密使用同一密钥，优点是运算速度较快，缺点是如何安全将密钥传输给另一方。常见的对称加密算法有：DES、AES 等。

非对称加密：指的是加密和解密使用不同的密钥（即公钥和私钥）。公钥与私钥是成对存在的，如果用公钥对数据进行加密，只有对应的私钥才能解密。常见的非对称加密算法有 RSA。



05 工具链和其他

考点 83 对于 Forward 和 Redirect 的理解。

难度系数: ★★★☆☆

【网易雷火事业部】

网易专家解析:

直接转发方式 (Forward) , 客户端和浏览器只发出一次请求, Servlet、HTML、JSP 或其它信息资源, 由第二个信息资源响应该请求, 在请求对象 request 中, 保存的对象对于每个信息资源是共享的。

间接转发方式 (Redirect) , 实际是两次 HTTP 请求, 服务器端在响应第一次请求的时候, 让浏览器再向另外一个 URL 发出请求, 从而达到转发的目的。

考点 84 对于 HTTP 协议是无状态的理解。

难度系数: ★★★☆☆

【腾讯微信事业部 / 百度移动生态事业部】

网易专家解析:

当浏览器第一次发送请求给服务器时, 服务器响应了; 如果同个浏览器发起第二次请求给服务器时, 它还是会响应, 但是呢, 服务器不知道你就是刚才的那个浏览器。简言之, 服务器不会去记住你是谁, 所以是无状态协议。





- 悄 悄 变 强 大 -