What did you learn yesterday?

* Learn fs lib in nodeJS

What excites you about coding.

* The moment that you solve the problem and achieve some function.

What is a recent technical challenge you experienced and how did you solve it.

* The problem I meet recently is how to realize upload function on website and save the upload document in the local server. First, I will watch relative video on YouTube. And then I will google NodeJS FS core lib, to check how the api work. What function it call.

When building a new web site or maintaining one, can you explain some techniques you have used to increase performance?

* 1. Check whether there is some JS code that can be re-used many time
* 2. Minimize the number of HTTP request
* 3. Cache frequently use document.

Can you describe some SEO best practices or techniques you have used lately?

* Put key word inside the <Head/>
* Put you domain name in google engine

Can you explain any common techniques or recent issues solved in regards to front-end security?

* For example MD5, which is password encryption technology. Which is unbreakable

Talk about your preferred development environment.

* I would said NodeJS. Frontend, it has ReactJS, Vue, AngularJS this frontend framework, Backend, it has NodeJS, which contain famous lib, like express, NestJS to build web applications.

Which version control systems are you familiar with?

* Git

Can you describe your workflow when you create a web page?

* 1. Basic Idea what this web page going to do, separate its function part.(Chat with backend, what the data structure would send to this page, what data would look like and build the fake data lib)
* 2. UI design, or used UI component directly,
* 3. Based on the page, separate by component. Do the basic components first, which is most requentlly used, like the button, avatar, input. And then used this basic component to build compose, which is bigger size.
* 4. Build demo frontend with fake data

How would you optimize a website's assets/resources?

* Use inline image, put the image on img store lib. Get the image through api
* Use correct image style

Name 3 ways to decrease page load (perceived or actual load time).

* Condemn JS and CSS
* Use CDN network speed up
* Reduce repeat code.

If you jumped on a project and they used tabs and you used spaces, what would you do?

* I would use tabs to replace the spaces that I used in the project. Because to make the project united and code simple is much important for me.

If you could master one technology this year, what would it be?

* I would said C++, because C++ is the most efficiently language based on the assembly language. But how to handle its pointer is very difficult and annying.

What resources do you use to learn about the latest in front end development and design?

* I will watch relative video first
* After learning rough idea, I will go to see the document to understand how it work
* If I met some problem, I will go stackoverflow to search similar questions

What skills are needed to be a good front-end developer?

* HTML/CSS/JS
* Git
* Browser developer tools
* Testing/debugging tools
* SEO
* Communication skills

What role do you see yourself?

* Front-end developer
* Full stack developer
* CTO

**Explain the difference between cookies, session storage, and local storage?**

* The lifecycle different

1. Cookie has expire time, or after close the browser, cookie disappear
2. LocalStorage: need to be clean by person
3. SessionStorage: only valiable in current webpage, close the browser also be disappear.

* Data size

1. Cookie: 4KB
2. Local&session Storage: 5MB

--------------------------------------- CSS ----------------------------------------

Explain the CSS “box model” and the layout components that it consists of

* Content - The content of the box, where text and images appear
* Padding - A transparent area surrounding the content (i.e., the amount of space between the border and the content)
* Border - A border surrounding the padding (if any) and content
* Margin - A transparent area surrounding the border (i.e., the amount of space between the border and any neighboring elements)

What is CSS selector specificity and how does it work?

* Css selector is to select the dom element in HTML
* Most often use selector is ID selector, class selector,

What's the difference between "resetting" and "normalizing" CSS? Which would you choose, and why?

* Due to we have lots of browser on the market, like Firefox, Chrome …. The default setting is different.
* Resetting is more like clean all the style.
* Normalizing is like to base on rule, clean the unnecessary one.

Describe Floats and how they work.

* Floats is CSS position property. Unlike absolute elements. Which are removed from the flow of the page
* It can be set left, right and both.

Describe z-index and how stacking context is formed.

* Z-index property in css controls the vertical stacking order of elements that overlap
* Based on the number of z-index, higher number show first.

Have you ever used a grid system, and if so, what do you prefer?

* 1. It could used to component define.
* Like [a, a, a, b]
* [a, a, a, b]
* And assign the a & b to correct component.

Are you familiar with styling SVG?=

* yes, SVG is kind of drawing by code.
* You can change its color and opacity based on code
* If you want to used SVG, you need to import {ReactComponent as xxx} from

Can you give an example of an @media property other than screen?

* Yes, @media property is used to fix different screen size to open web website

Explain how a browser determines what elements match a CSS selector.

* For example with selector p span, browsers will find all <span> element and then find the <p>element.
* If use key or class selector, it would be much faster

What is the CSS display property and can you give a few examples of its use?

* Display property is how it show like. Block, inline-block, inline, none.
* Block: take the whole line
* Inline: you can’t set its height and width, which is based on it children width and height
* Inline-block: contain the advantage between block and inline

Is there any reason you'd want to use translate() instead of absolute positioning, or vice-versa? And why?

* The reason why I need to use translate() is because I need to add some animation.

Can you explain the difference between px, em and rem as they relate to font sizing?

* Px
* Em: is based on its parent box
* Rem: is root, 1rem = 10px

Can you give an example of a pseudo class? Can you provide an example use case for a pseudo class?

* :after, :before
* :first-line

What is the difference between a block level element and an inline element. Can you provide examples of each type of element?

* Block for whole line
* Inline for particular part

-----------------------------------------JS -----------------------------------------

difference between var and let?

* Var is global scope, let is local or block scope.
* Let is unique, you can’t define twice using same name, which will throw an error
* In strict mode, it will recommend developer use let instead of var.

What's the difference between a variable that is: null, undefined or undeclared

* Undefined is a variable that has been declare but no value exists, and is a type of itself”undefine”
* Null is a value of a variable and is a type of object

What language constructions do you use for iterating over object properties and array items?

* For loop
* For … in …
* For each
* Map
* Reduce,

Can you describe the main difference between the Array.forEach() loop and Array.map() methods and why you would pick one versus the other?

* The return value is different, map will return the array after processing, but forEach will return undefine
* Map can chain other methods but forEach can’t do this, due to the return is different.
* forEach would mutate the array on which it is call, but map will

Explain the difference between: function Person(){}, var person = Person(), and var person = new Person()?

* Function Person() {} is to define a function
* Var person = Person() is a value of person function return
* Var person = new Person() a new instance of an object

My program is 2+2 with 2 bachelor degrees, one from China, one from UW. I am not allowed to switch to a CS major. But I am so interested in programming, and I took CS courses from UW CS department that required for the CS minor, and I also take advanced CS courses to fulfill my dream to become a top software developer.

I gain 2 bachelor degree and 1 minor degree in China and UWaterloo. I think this is one of my advantage. Frontend developer not only need hard code to solve the software problem. but it also requires good arts aesthetic and UI/UX experience. Furthermore, I also have relative experience on backend development. I am familiar with Nodejs, use express and nextjs to build backend platform.

**Explain what is the scope? What is the difference between var, let, and const in javascript?**

Scope is whether you have the accessability of the variable and function. In the other word, what variables you can access to. Two kind of scope, global scope and local scope.

Global scope: if a variable is declared outside all function or curly braces { }， it is said to be defined in the global scope.

Local scope: two kinds of local scope, one is function scope, and the other one block scope.

* Function scope is declare inside the function
* Block scope is declare inside the curly braces.

Const: const definition, one you define, you can change it. Most used on global scope.

Var and let: let is kind of update of var. let is more used on block scope, like for loop, which is a perfect sample. Var can use on global and function scope.

**What happens when you type a URL in the browser and press enter?**

1. Analysis the URL, check whether the URL is legal URL, or just key word search
2. DNS search, in this step, First check the browser and OS cache to check whether there is IP address relative to this URL. If no, then go to router cache, if still can’t find it, then go to root DNS, until find the final result
3. TCP connection, once we get the IP address, then we can build the connection between server and browser. TCP three hand shake connection.
4. After connected the server successfully, the request will sent to the server side. Server side handle the request based on the request to return response to the client side
5. Browser receive the response, check the header, cache the resource, handle the data
6. Render website page

**Where does javascript execute?**

Javascript is most often run on webpages in side the browser, but it can also run server-side, which is NodeJS

**If Javascript Is Single Threaded, How Is It Asynchronous?**

There is an example to explain this. Image JS is a person, everyday he wake up, he need to wash his face, but he also feel hungry. He open the microwave and push the food inside the microwave. And then he go to wash his face. When he done he go back to kitchen and get and eat the food.

Single Threaded mean that only JS can do all the process. Once JS have async operation, then JS will let other to handle the async operation, JS will handle the result once it return back.

The other example to explain JS must be a single threaded is that if JS is multi-thread, then one thread create DOM and other delete DOM. The browser have no idea which thread he should listen for.

**Hoisting?**

Hoisting mean you can use the function or variable before declare it.

For example, I have a function call addOne, which I didn’t declare it yet. But I can still use this function before I declare. That is because function declaration will move forward and save in cache, and then call the function.

Num = 6

Console.log(num)

Var num;

**JavaScript — is it Compiled or Interpreted?**

JavaScript is an interpreted language, not a compiled language. A program such as C++ or Java needs to be compiled before it is run.

JavaScript has no compilation step. Instead, an interpreter in the browser reads over the JavaScript code, interprets each line, and runs it.

If use ts, we need compile. It will translate the TS code to Nodejs code.

**Closures?**

Closures mean the variable inside the function can be read by its child function. Each function is a close package. You can’t read the variable inside the function directly.

Why we use closures?

1. If you want to read the internal function variable, then you can return child function, which contain the function variable.
2. If you want some function variable don’t destroy, after finish the function

JS进阶问题

1. **What’s different between undefined and null？**

* Undefined: a variable declared, but no value has been assigned a value;
* var demo;
* alert(demo); //shows undefined
* alert(typeof demo); //shows undefined
* Null: null is an assignment value, you can assign it to a variable. Null is similar to bool , string, and number.
* var demo = null;
* alert(demo); //shows null
* alert(typeof demo); //shows object

1. **why use && operator**

&& is logical operators in JS, It will return the first operand that is falsy. If no falsy operand was found, return the latest operand.

1. **Why use || operator**

|| is logical operators in JS. It was used to prevent some variable is undefined. It will return the first operand that is truthy, if no truthy operand was found, return the latest operand.

1. **DOM是什么**

DOM mean Documental Object Module. When browser read the HTML document, it will create a big tree structure object.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta http-equiv="X-UA-Compatible" content="ie=edge">

<title>Document Object Model</title>

</head>

<body>

<div>

<p>

<span></span>

</p>

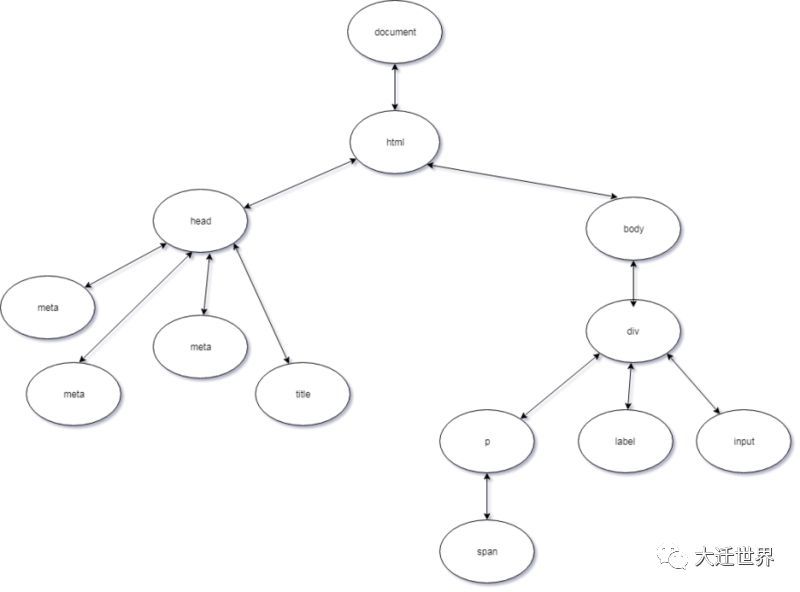
<label></label>

<input>

</div>

</body>

</html>



1. **What is different between preventDefault vs. stopPropagagtion.**

* **stopPropagation**: prevents propagation of the same event from being called. Propagation means bubbling up to parent elements or capturing down to child element
  + **I have two div, div1 and div2**. Div2 is div1’s child. They both has an onclick alert function. Normally, if I click div2, then div1 & div2 alert function will be triggered. If I add event.stopPropagation() in div2, then only div2 alert function will be triggered.
* **preventDefault**: prevent default action occur., like the submit button in the form.
  + Button in side the form, it’s default action will submit. If you don’t want to use it’s default action, then you preventDefault to stop it occur.

1. **Why `obj.someprop.x` will cause an error**

const obj = {};

console.log(obj.someprop.x);

because obj.someprop is undefined. Therefore, undefined don’t have the properties x, which will cause an error.

1. **What is different between == and ===**

==: is used to compare value, it could use to compare the value with different data type.

=== is used to strict comparation. Normally, it was used to compare the same data type value. To compare whether their address is same. Special example:

Let a = { a : 1};

Let b = { a : 1};

Let c = a;

Console.log(a === b) // false

Console.log(a === c) // true

1. **!! operator why use it.**

!! can mandatory switch right side value to bolen value.

1. **What is Hoisting**

Hoisting mean you can use the function or variable before declare it.

For example, I have a function call addOne, which I didn’t declare it yet. But I can still use this function before I declare. That is because function declaration will move forward and save in cache, and then call the function.

Num = 6

Console.log(num)

Var num;

If you use strict mode, you can write code like this

1. **Explain what is the scope? What is the difference between var, let, and const in javascript?**

Scope is whether you have the accessability of the variable and function. In the other word, what variables you can access to. Two kind of scope, global scope and local scope.

Global scope: if a variable is declared outside all function or curly braces { }， it is said to be defined in the global scope.

Local scope: two kinds of local scope, one is function scope, and the other one block scope.

* Function scope is declare inside the function
* Block scope is declare inside the curly braces.

Const: const definition, one you define, you can change it. Most used on global scope.

Var and let: let is kind of update of var. let is more used on block scope, like for loop, which is a perfect sample. Var can use on global and function scope.

1. **Closures**

Closures mean the variable inside the function can be read by its child function. Each function is a close package. You can’t read the variable inside the function directly.

Why we use closures?

1. If you want to read the internal function variable, then you can return child function, which contain the function variable.
2. If you want some function variable don’t destroy, after finish the function
3. **What is falsyValues in JS**

Const falsyValue = [“”, 0, null, undefined, NaN, false]

1. **What is “use strict” for?**

“use strict” module add some limitation when our coding. It can prevent some bug earlier

* Variable can be used after declare

The reason why use strict model,

* Strict mode makes it easier to write secure JS
* Strict mode change previously accepted “bad syntax” into real errors

1. What is the keyword of “this” in JavaScript

This: refers to an object, that object which is executing the current bit of javascript code. In other words, ever js function while executing has a reference to its current execution context, called this.

const carDetails = {

name: "Ford Mustang",

yearBought: 2005,

getName(){

return this.name;

},

isRegistered: true

};

console.log(carDetails.getName()); // Ford Mustang

1. **What is IIFE, why use it?**

IIFE: 'Immediately Invoked Function Expression', use racket to include a function, another racket include the parameter of the function.

Like this (function (a, b) { dosomething… } ) (aVal, bVal);

Why use it?

1. A variable defined in an IIFE can’t be accessed from the outside. It can be accessed in the enclosing block, which mean is safe
2. Loop index, it could executing asynchronous tasks inside a loop. Solve the async problem without using Promise and async/await
3. **Why use Function.prototype.bind?**

This was most use in React, in class. Bind() is used to bind the function to the class. Then you can use the function by this.function()

1. **What is High-order functions?**

Functions that taking functions as arguments or return a function.

function higherOrderFunction(param,callback){

return callback(param);

}

Callback function is kind of high-order functions

1. Array.prototype.map & Array.proptotype.filter & Array.proptotype.reduce

Map() create a new array, calls the provided function once for each element in an array, in order

Filter() create a new array, filter the element not fit for functions.

Reduce() executes a provided function for each value of the array (from left-to-right).

1. ES6有哪些新特性

* 箭头函数
* 类
* 模板字符串
* 加强的对象字面量
* 对象解构
* Promise
* 生成器
* 模块
* Symbol
* 代理
* Set
* 函数默认参数
* Res和展开
* 块作用域

1. What is arrow function.

Arrow function is more concise and convenience to use compare to the regular function expression.

In classic function expression, the `this` keyword is bound to different value based on the context in which it is called. With arrow functions, you don’t need to bind, because it bind automatic.

When should use arrow function

* When you always need to bind the function to this.
* Simple callback function, which save you lots of space and readable.

1. 主流前端请求方案有哪些，各有哪些利弊

Axios

Ajax

Frech

1. 如何解决回调地狱
2. Axios和Promise之间的关系是什么
3. Promise 和 Callback的区别是什么
4. Js里面垃圾回收系统是怎样的
5. 为什么JS是单线程，但是可以处理异步
6. **Where does javascript execute?**

Javascript is most often run on webpages in side the browser, but it can also run server-side, which is NodeJS

1. **What happens when you type a URL in the browser and press enter?**
2. 响应头有哪些
3. 浏览器存储的方式
4. Cookie和Session的认识
5. 浏览器渲染的步骤
6. 页面渲染优化
7. Get和Post请求的区别
8. 介绍304过程
9. HTTP状态码
10. 请求头和响应头都有什么信息
11. 什么是类
12. 什么是模板字符串？Template literals
13. 什么是对象解构？
14. ES6导入和导出区别，ES6模块
15. 什么是set对象， 怎么工作
16. 什么是callback 方法
17. Promise是什么
18. Async/Await是什么，如何工作的?
19. 。。。 有两个名字： rest参数和扩展运算符，它们之间的区别
20. 什么是默认参数
21. 隐式和显式转换有什么区别
22. 什么是NaN, 如何检查
23. 如何判断是否为数组
24. 什么AJAX
25. 如何在JS中创建兑现
26. In运算符使用
27. 哪些方法可以处理jS中的异步
28. 函数表达式和函数声明之间有什么区别？
29. Type of null是什么？为什么返回object
30. New 关键字有什么用
31. 什么时候不适用箭头函数
32. Object.freze()和const的区别