Name: Yipeng Hong NUID: 001228971

1. **Explain HTML5 Web Storage APIs with example. (5 Points)**

HTML5 Web Storage APIs can let web application store data locally within browser instead of cookies which are included in server request. It can store amount of data and is more secure which means it will not cause any impact on any other application and it is convenient to test the function of the web application. All data can be use like a formal web application until the web application is restart (refresh page) and the data will be reset. For example, you can sign up in a web application without server, and the web application will get the information you entered and run as a formal application which you can find your information in the console. But the data will be gone when you refresh the page and it will not affect anything.

1. **Describe HTML5 semantic elements <header>, <article>, <section>, & <footer>. (5 Points)**

All of these four semantic elements represent the container.

<header> is usually used as a container for navigational links or introductory content. For heading element, logo or relative information, these are often contained in <header>. <header> should not be contained in some specific elements like <footer>.

<article> can contain the meaningful content (comment, articles etc.) independently which means it can be distributed independently without affecting other section.

<section> acts as a container for the different part of any other document. It can act as a container for a chapter, header etc.

<footer> is used to contain the information of its inside elements and act as a footer of other sections.

1. **What is the difference between inline, inline-block, and block? (5 points)**

inline: It only can be used to set the margin-left, margin-right, padding-left and padding-right of the content. The padding will overlap the lines above and under.

inline-block: It has the same function as inline, and it also acts as a container that can set the height and the width of the block. It can be used as a block to flow within content. It can be put in place between the text without line break.

block: It has the same function as inline-block, and besides that, it also has the line break. So, the block will be put in a new line itself.

1. **Explain grid layout with example. (10 points)**

Grid layout let page be divided into many regions and it also can define the relationship between different regions. It is similar to the table, but it allows each region to be positioned independently which means region can overlap others. Each region can have independent CSS.

For example, <div class=`` region``>

<div class=``first``>First</div><div class=``second``>Second</div><div class=``third``>Third</div>

</div>

.region {display: grid; grid-template-columns: repeat(3, 1fr);grid-gap:5px;grid-auto-rows:maxmun(100px,auto);}

.first{grid-column: 1/3;grid-row:1} .second{grid-column:2/4;grid-row:1/4} .third{grid-column: 2/3; grid-row: 2/4}

And then, you will see there are three blocks positioned in the page and these grid has some overlapped region.

1. **Explain CSS z-index with example. (5 points)**

Z-index decides which the stack order of elements. If an element has greater stack order, it will be positioned in front of another element which has lower stack order. But to be attention, z-index is only valid for the position elements.

For example, there are two same size pictures ( named `one` and `tow`) and both of them are fixed in the same position. If `one` has value 999 of z-index and `two` has 9999, then you will see the `two` staying in front and `one` is behind `tow`.

1. **Write a regular expression to match the href value of an anchor tag. Explain your regex expression in detail. E.g. ‘<a href=”**[**https://northeastern.edu**](https://northeastern.edu/)**”>Northeastern University</a>’. (10 Points).**

Answer: /href="([^"]+)"/g

href=" matches the characters href=" literally (case sensitive); [^"] matches a single character not present in the list; + matches unlimited times (at least one time); ( ) does not work here, just for isolation; "" beside the ( ) match the "" of the href link; g modifier: global. All matches (don't return after first match)

1. **Explain the difference between XMLHttpRequest and fetch API. (5 points)**

Fetch API design is an improvement of XHR’s.

Fetch API: Without implementing the CORS you can still get response from server. Use Cache with req and res object. For streaming response, you can access the low-level stream.

1. **Explain hoisting in Javascript with example. Explain if `let` and `const` are hoisted in ES6. (10 points)**

Javascript hoisting means a undelcared variable can be used and declaration can be after it. It means that if a name is declared in a scope, the identifier can refer to that variable within that scope.

For example : x =`Insert Part`; insertPart = document.getElementById(`insert`); insertPart.innerHTML = x; var x: String. For this example, it will work and the position of the insertPart will be inserted the value of x. But to be attention, here we use the `var`.

`let` and `const` are not hoisted in ES6 and ReferenceError exception will occur if the declaration is after the usage of the name.

1. **Give example for rest and spread operator in ES6. (5 points)**

Rest operator: function sum(…sumOfAll) { return sumOfAll.reduce((sum, next) => sum + next) }

Call sum (1, 3, 5) and will get the result 9.

Spread operator: var numArr = [1, 2, 3, 4, 5]; var [one, two, …remaining] = numArr;

console.log(one); // 1 (number)

cosole.log(tow); // 2 (number)

console.log(remaining); // [3, 4, 5] (array)

1. **What is the difference between library, polyfills, and framework? (5 points)**

Latest angular is based on the latest standards of web platform. So, it means that not all of the browser can support all features which support the latest browsers. So, by loading the polyfills, it will compensate the features that the browser need. But it does not mean that polyfills transform the browser.

Library is something like bootstrap which contains set of css files or jqueries. It can seem like the collection of common css files and jquery functions. It let people more convenient to design web, so people can spend more time focusing in the design of the function. But for some specific css or function, people still need to design by themselves.

Angularjs is kind of framework. A framework is a real structure aimed at supporting the building of something. Framework is a layered structure that let people more conveniently build up something. It also has some specific function that let it easy to make use of some API or the connection to other servers. It can combine several complicated stuffs and make the building easier.

1. **Explain typescript module and angular module with example. (5 points)**

Typescript module, using export and import, can group related logic, structure the code and prevent pollution of the global namespace ( can not be visible ouside ).

export class validation { isValid(s: String) { return s.length == 20; } }

export { validation }

import { validation } from `./Validation`;

let checkValid = validation

Angular module is a collection of files ( filters, services etc ). It is available to an application and it has to be registered before being used in application.

In app.module.ts, import a new module: import: [ BrowserModule, NewComponentModule ] -> it should import {NewComponentModule} from `./path`

NewComponentModule is the module that contains other component. In this Module, it should: export class NewComponentModule {}

1. **Explain data binding in angular. (10 points)**

Data binding is a convenient way to connect the binding sources and target HTML elements. It allows user to declare some values in component and directly show or pass the data by using data binding. There are three binding types depend on the direction of data flow: source-to-view, view-to-source, view-to-source-to-view (two-way binding). It can let people get and deal with the data directly. It make the way passing and showing data between different component more convenient. By using data binding, data can be dealt with within html or used in typescript.

1. **Explain callback hell in javascript and how to avoid it? (5 points)**

As we know, Javascript is single-threaded event loop. So, when you have lots of callback functions in the code, especially when you need to use loop or try-catch ( nested ), it will make the code hard to work. In order to avoid it, one of the best ways is to modularize ( use module ). Then, give the function a meaningful and readable name, declare functions beforehand and make use of Promise.

1. **Give an example to set request headers in HttpClientModule in angular 4. (5 points)**

this.http.get(`localhose:3000/api/users`)

.map(data => data.json()).toPromise() //Get data

this.http.pose(`localhose:3000/api/users`)

.map(data => data.json()).toPromise() //Post data

this.http.delete(`localhose:3000/api/users` + id)

.map(data => data.json()).toPromise() //Delete data

this.http.put(`localhose:3000/api/users` + id, newUser)

.map(data => data.json()).toPromise() //Delete data

1. **Explain MVC architecture with controller, view, model, and service layers. (10 points)**

MVC represents Model-view-controller and is a design pattern for developing web application. It isolates the logic from the user interface layer and supports separation of concerns. Controller receives requests, deal with these data and interact with the Model. Then, Controller and Model will work together to output the data to response requests. Then the View will get the data form the Controller and the Model and present these data in particular format through some templating systems. The module used in MVC is not necessarily from Service, but service can deal with and pass some specific data to Module. Service layers can receive and update the values, then let the application perform specific logic.