Assessment:
Neha Noor

Part 1: HTML

1. **Explain the difference between block-level and inline elements with examples.**

Block Level Elements:

- Elements Which takes 100% width and make a block.
- Next Element will display below the block element.
- We can apply height and width to block elements.

Example: <p></p>, <h1></h1>, <div></div>

Inline Elements:

- Elements which take width according to their content.
- They do not take 100% width.
- Next element will display next to inline element.
- We can not apply height and width to block elements.

Example: <span></span>, <img /> etc.

2. How do you make an image accessible to screen readers?

To make the image accessible we can add an anchor tag. Then image tag between it. So that it would be assessble to screen reader.

3. What is the purpose of the `meta` tag in HTML?

Meta tags are tags for describing metadata of web page.

They declare in Title tag of web page.

A meta tag is used to display keywords and description about web page which help in **SEO** of website.

Similarly , a meta tag is used to define responsiveness by defining **veiwport** attribute and it’s initial scale.

Example: <meta author=”Neha Noor” description=”Associate Software Developer”/>

4. **Explain the difference between `<div>` and `<span>` tags.**

<div>

- <div> is a block element.
- It is used to make block of code in a web page.
- And to apply css in a block by using css selectors within <div> tag like class and ID.

Example: To make a box we wrap all elements of box within a div. And then we can apply css and well as javascript to make it dynamic.

<span>

- <span> is an inline element.
- It is used to make styling of content within a line.

Example: I have to change font colour of first word of heading, then i will wrap it into <span> tang. Then i can apply inline css.

5. **What are semantic HTML elements, and why are they important?**

- Semantic HTML elements are the elements which have special meaning and they are indicating a special purpose.
- Some of the major semantic HTML elements are <header>, <footer> , <nav>, <aside>, <audio>, <video>, <section> etc.
- For instance, <header> indicating that it demonstrate the top body of website, similary for the case of <footer> , it specify the foot section of a website.
- They are mainly used for **SEO optimization** of a website.

6. **What are HTML5 web storage options and how do they differ from cookies?**

HTML5 provides different options for web storage. Some of the main web storage options are given below:

- **Local Storage:** It provide storage at the browser for a specific limit. The stored data do not delete even when web page is closed.
- It is used to store data which need efficieny and latency of website. But don’t store sensitive data.
- **Session Storage:** It is used to store data for a limited time period. When session expired, data would be vanished.
- **IndexedDB :** It is a database in a browser , which is used to store such type of data which needs imediate retrival.
- It increase latency and efficiency of website.
- **Cookies:** Cookies are different from all above web storage options, because cookies is a piece of data send by server to the client(browser) which store essential information about client’s user.
- It stores essential information about client visit on website on the borwser.
- It would expires and delete after a specified time period.

Part 2: CSS

1.What is the box model in CSS, and how does it affect the layout of elements?

- Box model is a way to positon an element in a web page.
- Box Model consist of three major components: Margin, padding and border.
- Padding is space which we spacify around content of element.
- Border is a layer outside padding.
- Margin is spacing outside the border. Actually padding specify how much space an element display with other elements and the webpage.

2. **Explain the difference between `class` and `id` selectors in CSS.**

Class selector

- Class is a **CSS selector** to apply css by targeting through class name.
- Classes are not uniquely indentified a component.
- Class selctor is used for applying css by targeting any class name.
- There can be more than one class in an element as well as more than one element can contain the same class for styling purpose.
- We can also targeted classes for applying javascript.
- In javascript, we use:

  document.getElementById(.classname) to get all elements with specific classname.

3. **How can you create a responsive web design using CSS?**

Using CSS we can make responsive web pages by applying different best practices:

We can use Grid Veiw, to display elements in 2 demension (rows and colums).

We can use vw unit to specify height and width other than px, which display height and width according to the width of screen.

We can specify meta tag with veiwport attribute, which helps in resposiveness.

And most important , after applying all best practices **we have to apply Media Queries by specifing different screen's width to make page resposive through css.** If we are not using any other CSS framework.

4. What are CSS Flexbox and Grid, and when would you use each?

CSS Flexbox:

- CSS Flexbox is a method to display elements either in rows or colums.
- We have to apply Flexbox property to make differnt sections in a web page.

- For instance, I have to make 3 coulms, then i can use flexbox.

CSS Grid Veiw:

- It is similar to Flexbox , but the major differnce is that it is 2 Dimensional and we can specify rows as well as colums simultaneously.
- It is much flexible than flexbox.
- It is used to make dynamic sections too.

5. **How does the `position` property work in CSS? Describe the differences between `static`, `relative`, `absolute`, `fixed`, and `sticky`.**

Positions property deifne how are where an element should appear in the web page.

Static: It is a default property of HTML elements.

Relative: It is used to specify the position in relative to webpage and other HTML elements. We can apply top, right, left, bottom properties , if an element is relative.

Absolute: An element with absolute position become child of the element having relative position. It positioned top, right, bottom, left according to parent element. To make an absolute element , we have to specify postion: relative; to its parent element.

Fixed: To fixed on specific postion on screen. We can apply top, right, left, bottom properties to elements having fixed position.

Sticky: To stick it on the screen at specific postion.we can apply top, right, left, bottom properties to elements having sticky position.

6. **What is the difference between `margin` and `padding` in CSS?**

Padding: Spacing of an element around its content (space inside border in case of border).

Margin: Spacing of an element to other elements and web page (space outside border in case of border).

7. H**ow can you center a div horizontally and vertically using CSS?**

For center a div,

- Create a div.
- Apply width to a specifc width (For instance width: 50%;).
- Set margin: auto;

8. **What is the purpose of media queries in CSS? Provide an example.**

Media queries are used to make responsive web pages for different screen sizes.

Example:

- I have to create a card crousel.
- For Large screens , i have to display 4 cards in a column.
- For medium screens, i have yo display 2 cards.
- And for small ones, one card in a coulmn.
- By using i can specify the number of coulmn by specifing particular width.

@media(width<750px){

Div{

gird-tempate -colums: 1fr;

}

}

9. Explain the concept of CSS specificity and how it affects the application of styles.

10. What are CSS preprocessors, and what advantages do they offer?

- CSS preprocessor is also known as **SaSS and ScSS**.
- It is a css library to apply css much effectively.
- By using CSS preprocessor, we make components, import components into different files as well as it also increase **resuability** of styling.
- We can specify make varibles to maintain **consistency** and **integrity** throughout the web page.

Part 3: JavaScript

1.**Explain the difference between `var`, `let`, and `const` in JavaScript.**

Var

- var is a variable which have function scope.
- It is hoisted so that we can use it before declaration.
- It can declared as well as initialized again.
- It was used before ES6.
- It support all old browsers.

Let

- let introduced in ES6.
- It have block scope.

- It can’t be hoisted. Hositing of let casue Temporal Dead Zone (TDZ) and RefferenceError.
- It can initialze but can’t declared again.

const

- const introduced in ES6.
- It have block scope.
- It can’t be hoisted. Hositing of const casue Temporal Dead Zone (TDZ) and RefferenceError.
- It can’t initialze and declared again.
- We have to initialze const varible when we declare it.
- We used const to declare objects and arrays.
- It just make a constant reference, we can add or change properties of objects and arrays when we are using const.

2. **What is the event loop in JavaScript?**

Event loop is a process to trigger events in a loop (iterative mode).

3. **How does JavaScript handle asynchronous operations? Explain using callbacks, promises, and async/await.**

Javascript is a single threaded language.

It execute a statement at a time.

But it control the execution of block of code by applying asynchronus operations.

Asynchronous operations is a way in which a block of code is not wait for the executing of another block of code.

Callbacks:

- Callback is a function which passes as a parameter of another function.
- The callback function waits, until its containing funtion complete its execution.
- We can have nested callback functions to work it asynchronously.
- But nested callback function arise a situation known as callback hell.
- Which make a code ambigous.
- To overcome this dificulty , javascript introduce an other way for adsynchronous operations. That is promises.

Promises

- Promises is a javascript built in object having two callbacks; resolve and reject.
- Promises have three states:

Pending: Neither completed nor rejected.

Fullfiled: Task completed successfully.

Rejected: Task rejected

- Resolve callback execute when task is fullfiled and rejected callback execute if task is rejected.
- We also use .then and .catch methods to for resolve and reject callbacks.

Async/Await:

- Async/Await is discovered on the top of promises.
- They simplify the way in which promises used.
- Async function return a promise.
- And await is used to stop the execution until any specif task is done (Like fetching of data from API).

4. **What are closures in JavaScript, and how do they work?**

When we create nested javascript functions , then the most inner nested child would have access to all varibles of its parent function. This is called Lexical scope of that function. And that function is known as clouser function.

A clouser function can access all variables of its parent functions.

5. **Describe the concept of prototypal inheritance in JavaScript.**

Prototypal inheritance is a way of defining that an object is a prototype of its object type.

For instance,

I have created an object contructor function to create multiple objects having same properties.

If i would have to add property in the constructor. I have to add it on protype of contructor. Like:

myFunction.prototype.greet = (firstname) => return this.firstame;

Here myFunction is name of object function constructor.

6. What is the difference between == and === in JavaScript?

== is used to compare value not type. Like;

Let a = 5;

If (a == ‘5’)   //output: true

Javascript have default ability to convert string into number if number it is quotations.

=== compare value as well as type. Like;

```
Let a = 5;

If (a === ‘5’)   //output: false
```

7. Explain how you would create and manipulate objects in JavaScript.

There are different ways to create an manupulate objects in javacript:

Method 1: Object literals

We can make objects by make simply object literal. like;

```
const  obj = {}
```

Then we can add value as follows:

```
obj.name = “Neha”;
```

Method 2: By using new Keyword

For instance;

```
const  myObj = new Object();

myObj.name = “Neha Noor”
```

This method is not recommended because new keyword cause misleadings and may arise error.

Method 3: By making object constructor function

For instance;

```
Function myObj(firstname, lastname){

This.firstname = firstname;

this.lastname = lastname;

}

myObj(“Neha”, “Noor”);
```

If we want to make differnt objects having similar properties, we perfer this method.

8. **What is the difference between `map()`, `forEach()`, `filter()`, and `reduce()` array methods?**

map() : It is used to apply manipulation to every element of array then save to another array. Like if i want to multiply every element of array with 2 and then save it into anoter array.

forEach(): forEach() is used to iterate an array and display all elements of an array.

filter(): It is a function to all elements from array which fullfil specific condition. Like; if i have to find all numbers greater than 10.

reduce() : It is used to reduce size of array by eliminating last element.

9. **How do you handle errors in JavaScript?**

- We use try and catch block to handle errors in javascript.
- try block contain the block of code which we want to execute. It execute of there is no error.
- catch block have a parameter to pass error. This block execute if we encounter error. We can also display error on the console by getting error parameter.

10. **What are JavaScript modules, and how do you use them?**

- Javascript allow modularization for easability and usability of code.
- We can make modules make it easy to handle.
- We can use different modules by importing it.
- Javascript classes also allow us to make a modular structure of code by making multiple classes and extends to other classes.

11. **What is variable hoisting?**

- Variable hoisting is a javascript default behaviour to make declaration of variables and functions on the top of the scope.
- If we initialize a variable before declare it , javascript declare it by default on the top of the scope.
- For the case of var ,it can't make any issue.
- But let and const varibales make a reference error , because it would leads access of variable which is not initilize yet.
- For the case of let and const, it leads temporal dead zone.

12. **What is the temporal dead zone?**

If let and const variables are not declared, then javascript default behaviour move declaration to the top of the scope, but the variable is not initialized yet. This causes ReffernceError. And such situation leads to temporal dead zone.

13. **What will be the output of the following code?**

```
console.log(typeof null);
```

typeof null is equal to **object**.

14. **What will be the output of the following code?**

```
var foo = function bar() { return 12; };

console.log(typeof bar());
```

Output would be **Object**.

Part 4: Vue

1. **Explain the Vue instance lifecycle hooks.**

Lifecycle hook is used to specify execution of block of code at specific state of its lifecycle. Some important lifecycle hooks are given below:

1. Created
2. beforeCreated
3. Mounted
4. beforeMounted
5. Activated
6. beforeActivated
7. errorCaptured

2. **What are directives in Vue, and how do they work? Provide examples of common directives.**

Directives are special attributes which allow html template to manipulate according to vue instace.Some important directives are given below:

1. Var-if   (for cnoditional rendering)
2. Var-else (if ‘if ’ condition is not fullfiled)
3. var-for(for iteration)

3. **How does two-way data binding work in Vue?**

Two way binding is a way to change Model if View changes and vice versa. We use two-way binding to bind input fields with data properties to reterive data. We use v-model directive for this purpose.

4. Explain the difference between `v-if` and `v-show` directives.

v- if: Render element on the base of condition.

v- show: Manage display property of element according to condition.

5. **How do you handle events in Vue?**

In vue js, events handled by binding event handler in attribute. And make a specific function which trigger when event occurs.

6. **What is Vue Router, and how do you configure it?**

- Vue router is used to make routing within different pages and sections of a web page.
- To make routing in vue js, we have to install vue-router library.
- Then import it on a page where we would declare all routes.
- Then import all companents , which routes we have to make by including their path slug.

7. **How do you manage state in Vue applications?**

In vue js, we use vuex to manage state globally. It contain different options: state, getter, mutations and actions. We store global data on the store. Then manage the state by applying manipulations and operations on other options.

8. **What are Vue mixins and how do they work?**

Vue mixins are used to make a module.The module would contain all options: data, methods, watcher etc. And all these options would be encapsulate in a single function. Then we can export that mixin function anywhere in the applicaiton. It is used for reusability of code.

9. **What is the difference between computed and method?**

Computed properites: They are used for complex manipualtion of data. They cached results. If any dependency changes , then the computed function needs to re evaluate. Otherwise, it would display cached result.

Methods: Methods are used to add simple functions for basic manipulation. They can’t cached results.

10: **What are the ways to transfer data properties in components?**

1. By using props
2. By using inject/provider
3. By using $emit

Part 5: GIT

1. **What is Git and how does it differ from other version control systems?**

Git is a version control system. It uses differents commads to get or commit repositry to or from remote repository. It is easy to use that's why majority prefer git on other version control systems.

2. **Explain the basic workflow of Git.**

When we work on git, we would have to do following:

- We would have to clone a existing repo or push a branch.
- To create a new branch we would use the command: **git branch [branch name]**
- Then if we want to add it on staging: **git add [file name]**
- To commit it: **git comit [file name]**
- To merge it with any branch : **git merge main.**
- And **git pull [branch name]**

3. **What are commits in Git, and how do you create one?**

The commits is changes on a brach we made but not yet merge or push into repository.

To create commits:

First we would have to : **git add [file name]**   // branch is staging now

Then: **git comit [branch name]**   //Now changes on a branch is comited

4. **What is a branch in Git, and why are branches used?**

There is tree like structure in git. If there are difffent team members working on the same project, then every memeber make a branch to work individualy , then commit this branch to main branch to changes visible to everyone.

5. **How do you clone project?**

To clone an existing project, we would have to add following command:

git clone [repository url]

6. **What is a merge conflict, and how do you resolve it?**

Merge conflict is situation arrises:

- If there's a branch, someone make changes in a file.
- Then merge it with the main brach.
- Someone else make changes in the same file and same line and trying to merge it with the main branch.
- Such situation causes merge conflict.
- To solve merge conflict we can manually see, which code should be replaced . So an experienced developer resolve such conflicts by examine the situation.

7. **Can you describe the difference between `git merge` and `git rebase`?**

Git merge: It is merging a branch with another branch.
Git rebase: It is comiting a branch on the top of another branch.

Part 6: System Design

1. **Design a basic watcher system in JavaScript that observes changes to an object's properties and triggers a callback function**

Designing a watcher system in javascript involves the following steps:

Step #1: Make data reactive:

- To make data properties reactive. We can use Object.defineProperty as well as Proxies.
- Within Object.defineProperty and proxies, we have to add getter and setter function.
- **Getter function:** To get a property of object.
- **Setter function:** In this, first we have to compare old value with new value. If new value is changed then we would have to notify the watcher function.
- When watcher is notified for changes, first it access Dependency checker class.
- Dependency checker class contain all informations about data dependencies.

Step #2: Make a Dependency Checker Class

- Here we would use an array to push dependency data into array.
- This Dependency class is called when watcher is notified.

Step #1: Make a watcher class

- Watcher class call every element from depency array of dependency checker class. And change data property and comit all changes.
- All changes would store in a stack.
- After complete updating all dependencies, Watcher re render it and comit all changes in the virtual DOM.

That’s how, watcher worked on Javascript internally.

Scenario Based Question

1. **You are tasked with building a real-time collaborative text editor similar to Google Docs. Users should be able to edit documents simultaneously and see each other's changes in real time.**

Answer the following questions from above scenario

- **How will you handle real-time updates and ensure data consistency?**

For handling real time updates we can use websokets for real time updates simultaneoulsy changes made by others.It would also be a great technique for such editor.

- **What strategies will you use to minimize latency and manage concurrent edits?**

  To minimize latency i can use Vue reactive system. Vuex would be great option for that scenerio.

  To manage concurent edits, i can use vuex by making a global store for entire application. At that store data would be update concurently. And it’s a better way to edit concurent edits.

- **How will you implement the user interface to display real-time changes and user cursors?**

  For user cursor i can add textarea. And i can bind it with data using two-way binding (v-model) technique. It would ensure integrity in both model and view.

- **What performance optimizations are necessary for handling large documents?**

  For handling large documents, we can optimize performance by ensuring quick retrival of data.

  We can use AJAX, which is a fast way to load a part of web page without loading it.

  As well as we can use web storage like (local storage, session storage and indexed DB) to store data temprarily.

Management Based Questions

1. **Quality Control Issues**

Scenario: A manufacturing company is experiencing high defect rates in one of its product lines, leading to customer complaints and returns.

Questions:

1. **What steps would you take to identify the root cause of the quality issues?**

   To find root cause of quality issue, I can do SWOT analysis of whole project. To see where chances of risks and how i can mitigate impacts of these risks. And make quality improvable.

2. **How would you implement corrective actions to address the defects?**

   To address the defects, I can analysis it all over.

3. **What processes would you put in place to ensure ongoing quality control?**

   To control ongoing quality, i can use different quality management techniques like; Quality management .

4. **How would you communicate with customers about the steps being taken to resolve the issues?**

   We should have open communication. To discuss all painpoints. To ensure smooth understanding of whole procedure.

5. **What strategies would you use to prevent similar quality issues in the future?**

   To prevent such situation in future , we can use historical data for risk assessment techniques as well as to avoiding such scenerios to be occured.