Assignment 01 - Inception

Owner | Pankaj Kumar

1: What is Emmet?

- `Emmet` is the essential toolkit for web-developers. It allows you to `type shortcuts` that are then expanded into full pieces of code for writing `HTML, XML, CSS and Other languages` based on an abbreviation structure most developers already use that expands into full-fledged HTML markup and CSS rules.
- In Visual Studio Code (VS Code), Emmet is built-in and can be accessed by typing abbreviations and then pressing the Tab key to expand them.
  For example, typing "div.container" and pressing Tab will expand to:

```
<div class="container"></div>
```

2: Difference between a Library and Framework?

- `library` is a collection of packages that perform specific operations whereas a `framework` contains the basic flow and architecture of an application. The major difference between them is the complexity. Libraries contain a number of methods that a developer can just call whenever they write code. React js is library and Angular is Framework.
- One key difference between libraries and frameworks is that libraries are called by the code, while frameworks call the code. In other words, when you use a library, you are in control of how and when it is used. With a framework, the framework is in control and dictates how the code should be organized and used.

3: What is CDN? Why do we use it?

What_is_a_CDN__.pdf

(OR)

- CDN(Content Delivery Network) are used by a wide range of websites and applications, including e-commerce sites, media and entertainment platforms, and software as a service (SaaS) providers.
- **Improving the speed of web pages:** By serving content from a server that is physically closer to the user, CDNs can significantly improve the loading speed of web pages.
- **Reducing bandwidth costs:** CDNs can reduce the amount of bandwidth required by a website or application by serving cached copies of content from servers located closer to the user.
- **Enhancing security:** CDNs can provide additional security measures such as DDoS (Distributed Denial of Service) protection and SSL(Secure Socket Layer) encryption, helping to protect websites and applications from attacks.
- **Improving availability:** CDNs can help to improve the availability of a website or application by providing multiple copies of content that can be served to users if one server goes down.

4: Why is React known as React?

- React is a JavaScript-based UI development library. Facebook and an open-source developer community run it. It is designed to be declarative, meaning that developers specify what the UI should look like based on the current state of the application, and React takes care of updating the UI as needed. React is designed to be efficient and flexible, and is widely used in the development of web and mobile applications.

5: What is crossorigin in script tag?

- The crossorigin attribute sets the mode of the request to an HTTP CORS Request.

- The purpose of crossorigin attribute is used to share the resources from one domain to another domain. Basically, it is used to handle the CORS request. It is used to handle the CORS request that checks whether it is safe to allow for sharing the resources from other domains.
- The crossorigin attribute on a `<script>` tag specifies that CORS is supported when loading an external script file from a third party server or domain. CORS is a standard mechanism used to retrieve files from other domains.

Syntax

```
<script crossorigin="anonymous|use-credentials">
```

6: What is difference between React and ReactDOM?

- React and ReactDOM are two separate libraries that are often used together in the development of web applications with React.
- `React` is a JavaScript library for building User Interfaces whereas `ReactDOM` is also JavaScript library that allows `React to interact with the DOM`.
- ReactDOM, on the other hand, is a library that provides an interface between React and the DOM (Document Object Model). The DOM is a tree-like structure that represents the HTML of a web page, and ReactDOM provides a set of functions that allow React components to be rendered to the DOM and updated efficiently.
- In short, React is a library for building user interfaces, while ReactDOM is a library for interacting with the DOM and rendering React components to the web page. While they are often used together, they serve different purposes and can be used independently of each other.
- The react package contains `React.createElement()`, `React.Component`, `React.Children`, and other helpers related to elements and component classes. You can think of these as the isomorphic or universal helpers that you need to build components. The react-dom package contains `ReactDOM.render()`, and in react-dom/server we have server-side rendering support with `ReactDOMServer.renderToString()` and `ReactDOMServer.renderToStaticMarkup()`.

```
const listElement = React.createElement( 'li', {
    className: 'list'
}, 'React.js' );

ReactDOM.render( listElement, document.querySelector( '#root' ) );
```

7: What is difference between react.development.js and react.production.js files via CDN?

- Development is the stage of an application before it's made public while production is the term used for the same application when it is made public.
- Development JS for development reasons. You have Source Maps, debugging and often times hot reloading ability in those builds.
- The production build, on the other hand, runs in production mode which means this is the code running on your client’s machine.
- Development build is several times (maybe 3-5x) slower than the production build.

8: What is async and defer?

`Async` - The async attribute is a `boolean attribute`. The script is downloaded in `parallel(in the background)` to parsing the page, and `executed as soon` as it is available (do not block HTML DOM construction during downloading process) and don’t wait for anything.

Syntax

```
<script async src="script.js"></script>
```

`Defer` - The defer attribute is a `boolean attribute`. The script is downloaded in `parallel(in the background)` to parsing the page, and `executed after the page` has finished parsing(when browser finished DOM construction). The `defer attribute` tells the browser `not to wait for the script`. Instead, the browser will continue to process the HTML, build DOM.

Syntax

```
<script defer src="script.js"></script>
```