Assignment 1

▼ Difference between library and framework

Framework	Library
Framework, on the other hand has defined open or unimplemented functions or objects which the user writes to create a custom application	A programmer calls the library when and where he needs it. He controls it
Angular	React

▼ What is CDN? Why do we use it?

- A CDN is a content delivery network. It is a system of distributed servers that deliver web content to users based on their geographic location.
- CDNs are used to improve website performance by reducing latency and providing a more reliable connection.

▼ Why is React known as React?

> 💡 React is named React **because of its ability to react to changes in data.** When the data in a React component changes, React will automatically re-render the component so it reflects the new data

▼ What is cross-origin in the script tag?

> 💡 Provides support for <u>CORS</u>, defining how the element handles cross-origin requests, thereby enabling the configuration of the CORS requests for the element's fetched data.

- **Cross-Origin Resource Sharing**
  (<u>CORS</u>) is an <u>HTTP</u> -header based mechanism that allows a server to indicate any <u>origins</u> (domain, scheme, or port) other than its own from which a browser should permit loadingresources.

▼ What is difference between React and ReactDOM

React	ReactDOM
This is the core library	This is used to show on browser
Used to create react elements	Used to show react elements on the browser
Eg : react.createElement, react.createClass	Eg: ReactDOM.render()

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

Both React and ReactDOM are available over a CDN.

```
<script crossorigin src="https://unpkg.com/react@18/umd/react.development.js"></script>
<script crossorigin src="https://unpkg.com/react-dom@18/umd/react-dom.development.js"></script>
```

The versions above are only meant for development, and are not suitable for production.
Minified and optimized production versions of React are available at:

```
<script crossorigin src="https://unpkg.com/react@18/umd/react.production.min.js"></script>
<script crossorigin src="https://unpkg.com/react-dom@18/umd/react-dom.production.min.js"></script>
```

The development version contains helper function such as warnings whereas, in production

all comments are removed and dont include helper function .

```
// If we compare size when we do npm install

-- During Development
react-devleopment.js == 106KB
react-dom-devlopment.js == 909KB

--During production
react.production.js == 16KB
react-dom.proudction.js == 119KB
```
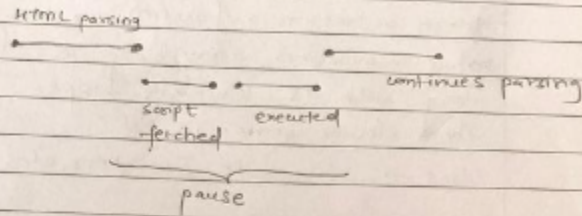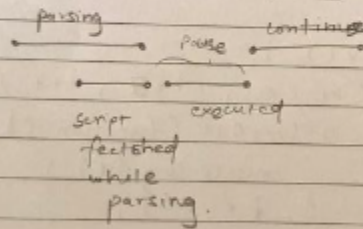
▼ What is async and defer?

Async	Defer
Script is fetched while HTML is parsing and executed after it is fetched and then again parsing is started	Script is fetched while HTML parsing and executed after the HTML Parsing is finished, hence there is no pause
It does not guarantee that the scripts are executed in order	Scripts are executed here in order
We can use this when the scripts which we are using are independent of our code. Eg ; google analytics	We can use this when we want scripts to be executed in an order, meaning if one script is dependent on the other.

Async Vs defer attributes

→ Normal → < script src = "   . >

HTML parsing

continues parsing

script   executed
fetched

pause

→ Async. → < script async src = "   ' >

parsing   continue

Pause

executed

script
feclched   Async does
while   not gurantee
parsing.   that the
                 scripts are
                 executed in order

⇒ Defer.

parsing completed

script   executed
fected   after
during   HTML
parsing   parsing done.