

browser fundamentals / architecture

- network component
 - used to send the request to the server
 - used to get the response from the server
- rendering component / engine
 - also known as a layout engine
 - used to render the HTML page including CSS (convert the html to JavaScript)
- javascript engine
 - the heart of any browser
 - used to execute the JavaScript code
- user interface component
 - used to display the user interface
- web-storage component
 - used to store the data in the browser
 - e.g. local storage, session storage, cookies

React

- a JS library used to develop Single Page Application
- Single Page Application
 - contains only one html page
 - gets loaded only once when user visits the website
 - once loaded, it sends the request to the server only to get the data
 - it executed only on the client side (inside a browser)
- react is used to develop client side applications
- developed by Facebook and open sourced for other developers
- features
 - has a component-driven architecture
 - used to developer SPA type applications
 - used virtual DOM for improving the application performance
 - it has eco-system: React Router, React Redux Toolkit, React Native

installation

used CDN links

- downloading the react library every time the page is loaded
- to use the CDN links:

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

```
<!-- Don't use this in production: -->
<script src="https://unpkg.com/@babel/standalone/babel.min.js"></script>
```

- react.development.js
 - used for developing the react application
 - used to create react components
- react-dom.development.js
 - used to render the react components inside the browser
- babel.min.js
 - used to convert the JSX code into JS code

using application managers

- using vite

```
# create react app using vite
> npm create vite@latest myapp

# go to the directory
> cd myapp

# install the dependencies
> npm install

# run the application
> npm run dev

# visit the url: http://localhost:5173/
```

data binding

- using the variable value inside a html tag
- in react, it will be done using interpolation
 - used the {} brackets for loading the variables value inside the html tag

```
const myvar = 100
<h3>{myvar}</h3>
```

- to render a simple variable use interpolation

```
const myvar = 100
```

```
<h3>{myvar}</h3>
```

- to render an object, split the object into properties and use interpolation to render those properties

```
const car = {
  model: 'triber',
  company: 'renault',
  price: 10
}

<div>
  <div>model = {car['model']}</div>
  <div>company = {car.company}</div>
  <div>price = {car.price}</div>
</div>
```

```
const car = {
  model: 'triber',
  company: 'renault',
  price: 10
}

const {model, company, price} = car

<div>
  <div>model = {model}</div>
  <div>company = {company}</div>
  <div>price = {price}</div>
</div>
```

- to render an array, use the map function to iterate over the array and use interpolation to render the properties of the object

```
const cars = [
  { model: 'triber', company: 'renault', price: 10 },
  { model: 'seltos', company: 'kia', price: 20 },
  { model: 'creta', company: 'hyundai', price: 30 },
]

const div = cars.map((car, index) => {
  return (
    <div key={index}>
```

```
    <div>model = {car['model']}</div>
    <div>company = {car.company}</div>
    <div>price = {car.price}</div>
  </div>
)
})
```