Assignment – 10

Owner – Mohit R

1. Explore all the ways of writing CSS.

- Native CSS - all components's styles in a single file index.css
- SAAS and SCSS - Syntactical CSS - it is then converted to css
- Inline - style attribute - pass object - {{backgroundColor : "red"}}
- Component Library - MaterialUI, Bootstrap, Base Web UI, Ant design, chakra UI
- styled components
- CSS Framework - Tailwind

2. How do we configure tailwind?

- Steps to use tailwind css in app :
- Install tailwind css

```
npm install -D tailwindcss postcss
npx tailwindcss init
```

- Configure PostCSS

```
{
  "plugins": {
    "tailwindcss": {}
  }
}
```

- Configure your template paths

```
module.exports = {
  content: [
    "./src/**/*.{html,js,ts,jsx,tsx}",
  ],
  theme: {
    extend: {},
  },
  plugins: [],
}
```

- Add the Tailwind directives to your CSS

```
@tailwind base;
@tailwind components;
@tailwind utilities;
```

3. In tailwind.config.js, what do all the keys mean (content, theme, extend, plugins)?

- **Content**
  - The content section is where you configure the paths to all of your HTML templates, JS components, and any other files that contain Tailwind class names.
- **theme**
  - We can customize color palette, spacing scale, typography scale, or breakpoints using theme section of your tailwind.config.js file
- **plugins**
  - the plugins section allows you to register plugins with Tailwind that can be used to generate extra utilities, components, base styles, or custom variants.

4. Why do we have .postcssrc file?

- Tool for transforming CSS with JavaScript
- Basically, if we want to transform CSS inside JavaScript we use postcss
- Create a .postcssrc file in your project root and enable the tailwindcss plugin.
- We write the below code in our .postcssrc file

```
{
"plugins": {
"tailwindcss": {}
}
}
```