Chapter 03 - Laying the Foundation

Topics

- JSX
- React.createElement vs JSX
- Benefits of JSX
- Behind the Scenes of JSX
- Babel & parcel role in JSX
- Components
- Functional Components
    - Composing Components

Assignment

- What is JSX?
  JSX is a html like syntax
- Superpowers of JSX
  The superpowers of JSX are
  1. JS integration
  2. Conditional Rendering
  3. looping
  4. Dynamic Attributes
- Role of `type` attribute in script tag? What options can I use there?
  This attribute indicates the type of script represented.
  its options are

1. `importmap`
2. `module`
3. `speculationrules`

- `{TitleComponent}` vs `{<TitleComponent/>}` vs `{<TitleComponent></TitleComponent>}` in JSX

the `{TitleComponent}` this syntax is used when we need to pass the component as a prop

`{<TitleComponent/>}` It is normal function component called and rendered

`{<TitleComponent></TitleComponent>}` This is also an normal component called and rendered , its main purpose is to render group of children between those tags

Coding Assignment:

- Create a Nested header Element using React.createElement(h1,h2,h3 inside a div with class “title”)
    - Create the same element using JSX
    - Create a functional component of the same with JSX
    - Pass attributes into the tag in JSX
    - Composition of Component(Add a component inside another)
    - `{TitleComponent}` vs `{<TitleComponent/>}` vs `{<TitleComponent></TitleComponent>}` in JSX
- Create a Header Component from scratch using Functional Components with JSX
    - Add a Logo on left
    - Add a search bar in middle
    - Add User icon on right
    - Add CSS to make it look nice

References

- Babel: https://babeljs.io/
- Attribute Type: https://developer.mozilla.org/en-US/docs/Web/HTML/Element/script#attr-type

- JS Modules: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Modules
- Babel Playground: https://babeljs.io/repl#
- React without JSX: https://reactjs.org/docs/react-without-jsx.html