# Strict Mode

* A component that is Part of React
* During development it will render the components TWICE to check for BUGS

# Components

* The most fundamental part of React.
* Components are the building blocks of React.
* Each Component has its own
  + Data
  + Logic
  + Appearance
* Components Tree
* A screenshot of a computer

  Description automatically generated

# JSX

* Declarative Syntax
* Each component must return a block of JSX
* JSX in compiled by the BABEL Compiler

# JS logic inside Components

* We can write whatever JS logic we want in the component, outside the return statement

# Separation Of Concerns

* Logic and UI are tightly coupled in REACT.
* They are collocated.
* The separation of Concerns in REACT is
  + ONE COMPONENT PER FILE

# Styling React Components

* There are multiple ways to Style the application
  + Inline
  + External
  + SASS
  + Tailwind
* To define INLINE styles, we need to create a JS object {{}}
  + 1st pair of curly braces are to enter the JS mode
  + 2nd pair of curly braces are to create a JS object
* To use External CSS file
  + Import the css file
    - Import ‘./index.css’;

# Props

* We use props to pass information down the component tree, from Parents to Children
* Anything can be passed as a Prop
  + Single values
  + Arrays
  + Objects
  + Functions
  + Other components
* PROPS ARE IMMUTABLE – READ-only
* ONE-WAY data Flow
  + In React Apps, data can only be passed from PARENT to CHILDREN using PROPS
  + Makes apps more predictable and easier to understand
  + Easier to debug
  + Is more performant

# React Fragments

* Let’s us group HTML element without leaving any trace in the DOM tree
* <></>
* When we need a Key
  + <React.Fragment key={x}>
  + </React.Fragment}