### 1. Why is it necessary to start component names with a capital letter?

In React, it is necessary to start component names with a capital letter. If we start the component name with lower case, it will throw an error as an unrecognized tag. It is because, in JSX, lower case tag names are considered as HTML tags.

### 2. What are fragments?

In was introduced in React 16.2 version. In React, Fragments are used for components to return multiple elements. It allows you to group a list of multiple children without adding an extra node to the DOM.

render() {

**return** (

    <React.Fragment>

      <ChildA />

      <ChildB />

      <ChildC />

    </React.Fragment>

  )

}

render() {

**return** (

    <>

      <ChildA />

      <ChildB />

      <ChildC />

    </>

  )

}

### 3. Why are fragments better than container divs?

* Fragments are faster and consume less memory because it did not create an extra DOM node.
* Some CSS styling like CSS Grid and Flexbox have a special parent-child relationship and add <div> tags in the middle, which makes it hard to keep the desired layout.
* The DOM Inspector is less cluttered.

### 3. How to apply validation on props in React?

Props validation is a tool which helps the developers to avoid future bugs and problems. It makes your code more readable. React components used special property PropTypes that help you to catch bugs by validating data types of values passed through props, although it is not necessary to define components with propTypes.

We can apply validation on props using App.propTypes in React component. When some of the props are passed with an invalid type, you will get the warnings on JavaScript console. After specifying the validation patterns, you need to set the App.defaultProps.

**class** App **extends** React.Component {

          render() {}

}

Component.propTypes = { /\*Definition \*/};

### 4. What is create-react-app?

### Create React App is a tool introduced by Facebook to build React applications. It provides you to create single-page React applications. The create-react-app are preconfigured, which saves you from time-consuming setup and configuration like Webpack or Babel. You need to run a single command to start the React project, which is given below.

### $ npx create-react-app my-app

This command includes everything which we need to build a React app. Some of them are given below:

* It includes React, JSX, ES6, and Flow syntax support.
* It includes Autoprefixed CSS, so you don't need -webkit- or other prefixes.
* It includes a fast, interactive unit test runner with built-in support for coverage reporting.
* It includes a live development server that warns about common mistakes.
* It includes a build script to bundle JS, CSS, and images for production, with hashes and source maps.

### 5.  How can you create a component in React?

There are two possible ways to create a component in React:

**Function Components:** This is the simplest way to create a component in React. These are the pure JavaScript functions that accept props object as the first parameter and return React elements:

function Greeting({ message }) {

  return **<h1>**{`Hello, ${message}`}**</h1>**

}

**Class Components:** The class components method facilitates you to use ES6 class to define a component. The above function component can be written as:

class Greeting extends React.Component {

  render() {

    return **<h1>**{`Hello, ${this.props.message}`}**</h1>**

  }

}

### 6.When do we prefer to use a class component over a function component?

If a component needs state or lifecycle methods, we should use the class component; otherwise, use the function component. However, after React 16.8, with the addition of Hooks, you could use state, lifecycle methods, and other features that were only available in the class component right in your function component.

### 7. Is it possible for a web browser to read JSX directly?

Web browsers can't read JSX directly. This is because the web browsers are built to read the regular JS objects only, and JSX is not a regular JavaScript object.

If you want a web browser to read a JSX file, you must transform the files into a regular JavaScript object. For this purpose, Babel is used.