Skill Training React JS

Narasimha

Schedule for React JS Training

Day1: State Management: Class Components

Day2: State Management: Functional Components, Hooks

Day3: Http Client Programming

Day4: Working with Forms, Validations, Services

Day5 : Redux

Day6: Unit Testing

Day7 : Routing – SPA in React JS

Day1 State Management in React JS



Narasimha

Index – Day1

- 1. State and Props in React Components
- 2. State management API: setState()
- 3. How to manage state in class components
- 4. Stateless component

React Components: Stateful / Stateless

Stateful Components

- Stateful / Container / Smart
- That means the stateful components are keeping track of changing data.
- Class based components are stateful

Stateless Components

- Stateless / presentational / dumb
- Concerned with presenting data how things look
- Usually are STATELESS but not always
- Most likely a 'child' of a container component
- Will make up the majority of your application

Components

- Class Components (stateful)
- Functional Components without hooks (stateless)
- Functional Components with hooks (stateful)



Working with Props

Sr. IT Trainer/Consultant

Working with props

- "Props" is a special keyword in React
- It stands for properties
- It used for passing data from one component to another.
- props are used like attributes in html tags.
- props data is read-only, which means that data coming from the parent should not be changed by child components.

Working with props

```
<Demo uname="Narasimha" />

<div>
    Welcome to {this.props.uname}
</div>
```

Practice Hands-Ons

Narasimha

How to manage the state?

Working with state in Class Components

- state is an instance member of React Component.
- It can be defined as an object of a set of properties.
- It holds the information that may change over the lifetime of the component.
- By default every component will get state object as null from parent component (React.Component).
- We initialize state in component constructor with required properties.

```
this.state = { uname : "Scott"};
```

Usage of setState() method

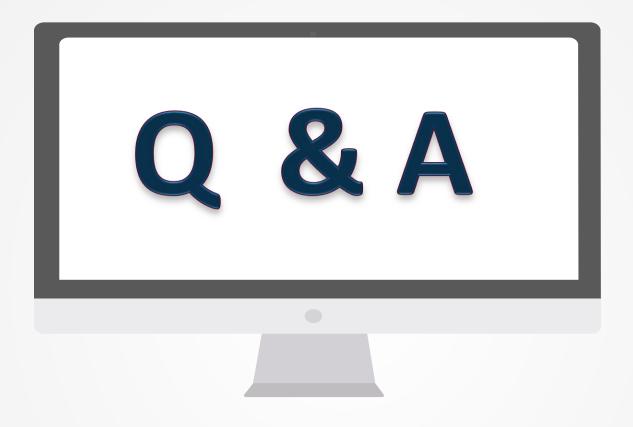
- setState() schedules an update to a component's state object.
- When state changes, the component responds by re-rendering.
- setState() is the only legitimate way to update state after the initial state setup.

Usage of setState() method

```
class Demo extends React.Component
    constructor()
          this.state = { name : "Scott" }; // Initialize State
          this.buttonClick = this.buttonClick.bind(this);
   buttonClick() {
          this.setState( { name : "Scott" } ); // Update State
```

Stateless Components

Sr. IT Trainer/Consultant



Narasimha