React Basics 1

Conditional Rendering with &&

```
{this.state.toggle &&
<p>This should show and hide</p>
}

//This will render p tag if this.state.toggle is true otherwise not
```

Refs in DOM

```
class App extends Component {

submit = () => {
console.log(this.text.value);

}

render() {
return (
<div className="App">

<input type="text" ref={(input) => this.text = input} />
<button onClick={this.submit}>Show Value</button>
</div>
);
}
}

//Notice that inside input tag we give prop ref and give arrow function bu passing
value of any var then we create a variable inside arrow function using this.text =
input. then we handle that this.text var in onclick event method.

//Using refs is uncontrolled, we cannot check to see if ip given is in lowerspaces or
without spaces
```

Controlled ip and two way binding

```
class App extends Component {

state = {
input:"Hello",
}
updateInput = (event) => {
this.setState({
input: event.target.value,
})
}

render() {
return (
<div className="App">

<h3>{this.state.input}</h3>

<input type="text" onChange={this.updateInput} value={this.state.input}/>


<button onClick={this.submit}>Show Value</button>
</div>
);
}
}


//Flow of program is first when page is loaded, value will be hello as we have given input value as hello by linking it to state, Then when we change ip , onchange method is called which again set the state to current event.target.value , as we change state using set state, value prop inside input will point to new state and get updated. Note that if we hardcode value of ip as string "Hello" , input field will not change as it always is hello
//Two way binding is simply printing new input value using h3 tag
//This is controlled ip as we can check in updateInput method if ip contains no spaces using
input: event.target.value.trim()
```