# React Basics 1

## import syntax

```
import React, { Component } from 'react';
```

## export Syntax

```
Code Sampleexport default App;
```

## Components Types

### Class Based Component

```
class App extends Component {
render () {
return (
<div>
Hello {this.props.name}
</div>
)
}
}
//Note that when returning multiple html
elements,wrap it inside div and insite
brackets of return st
```

### Functional components

```
const Header =(props) => {
return(
<div>
Hello {this.props.name}
</div>
)
}
//props are passed as the first
parameter to a function.
```

## Using Constructor inside react

```
class App extends Component {

constructor(props)
{
```

```
super(props);
}

//Rest of the code
```

## States

```
//State Declaration
class App extends Component {

state = {
toggle: true,
name: 'Akash',
}

//Using state
<p>{this.state.name}</p>
```

## Props

```
//First pass props to child component
<Video name="Akash" />

//recieve props in child component as
render () {
<p>{this.props.name}</p>
}
//Note that props is an array which contains all the props passed for that component
```

## Children

```
<AlertBox>
<h1>You have pending notifications</h1>
</AlertBox>

class AlertBox extends Component {
render () {
return <div>
{this.props.children}
</div>
}
}
//Children are passed as the children property.
```

# 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()
```

# React Basics 3

## Event Handling using arrow function

```
class Trial extends Component {

logHandler = () => {
console.log('Event triggered');

}
render() {
return (
<div>
<div onClick={this.logHandler}>Click me
to log</div>
</div>
);
}
}
```

## With values passed

```
class Trial extends Component {

logHandler = (name) => {
console.log('Event triggered',name);

}
render() {
return (
<div>
<div onClick={() => {this.logHandler(
'Akash')}}>Click me to log</div>
</div>
);
}
}
```

## With arrow,parameters and Bind

```
class Trial extends Component {

logHandler = (name) => {
console.log('Event triggered',name);
}
render() {
return (
<div>
<div onClick={this.logHandler.bind(this,'Akash')}>Click me to log</div>
</div>
);}}
```

## With normal method and bind

```
class Trial extends Component {

logHandler(name) {
console.log('Event triggered',name);

}
render() {
return (
<div>
<div onClick={this.logHandler.bind(this,
'Akash')}>Click me to log</div>
</div>
);}}
```

## With normal method and arrow function in event

```
class Trial extends Component {

logHandler(name) {
console.log('Event triggered',name);

}
render() {
return (
<div>
<div onClick={() => this.logHandler(
'Akash')}>Click me to log</div>
</div>
);}}
```

Note that .bind() is more fast than arrow inside event.

# React Basics 4

## PropTypes

## First install prop-types

```
npm install prop-types --save
```

## Import proptypes

```
import PropTypes from 'prop-types';
```

## Proptypes Syntax

```
static propTypes = {
movie: PropTypes.shape({
title: PropTypes.string.isRequired,
}),
desc: PropTypes.string
}

//Note that isRequired make sure that prop is required and shape method is for nested
object
```

## Default props syntax

```
static defaultProps = {
desc: 'Description not availabel'
}

//default props do not support nested objects
```