Department of Computing

**CS-213: Advanced Programming**

**Class: BSCS 7A**

# Lab 11: React Native Calculator Application

**Date: 05 December, 2019**

**Time: 10:00-01:00pm & 02:00-05:00pm**

# Instructor: Dr. Sidra Sultana

**Lab Engineer: Ms. Ayesha Asif**

# 

# Lab 11: React Native Calculator Application

**Farukh Ahmed Shaikh (220247)**

**BSCS 7 A**

**Lab Task**

Create a basic calculator app in react native

|  |
| --- |
| App.js |
| import React, { Component } from 'react';  import {  View,  Text  } from 'react-native';  import Style from './Style';  import Button from './Button';  *// Define the input buttons that will be displayed in the calculator.*  const inputButtons = [  [1, 2, 3, '/'],  [4, 5, 6, '\*'],  [7, 8, 9, '-'],  ['C', 0, '=', '+']  ];  export default class App extends Component {  constructor(props) {  super(props);  this.initialState = {  previousInputValue: 0,  inputValue: 0,  selectedSymbol: null  };  this.state = this.initialState;  }  render() {  return (  <View style={Style.rootContainer}>  <View style={Style.displayContainer}>  <Text style={Style.displayText}>{this.state.inputValue}</Text>  </View>  <View style={Style.inputContainer}>  {this.\_renderInputButtons()}  </View>  </View>  );  }  \_renderInputButtons() {  let views = inputButtons.map((row, idx) => {  let inputRow = row.map((buttonVal, columnIdx) => {  return <Button  value={buttonVal}  highlight={this.state.selectedSymbol === buttonVal}  onPress={this.\_onInputButtonPressed.bind(this, buttonVal)}  key={'butt-' + columnIdx} />;  });    return <View style={Style.inputRow} key={'row-' + idx}>{inputRow}</View>;  });  return views;  }  \_onInputButtonPressed(input) {  switch (typeof input) {  case 'number':  return this.\_handleNumberInput(input);  default:  return this.\_handleStringInput(input);  }  }  \_handleNumberInput(num) {  let inputValue = (this.state.inputValue \* 10) + num;  this.setState({  inputValue: inputValue  });  }  \_handleStringInput(str) {  switch (str) {  case '/':  case '\*':  case '+':  case '-':  this.setState({  selectedSymbol: str,  previousInputValue: this.state.inputValue,  inputValue: 0  });  break;  case '=':  let symbol = this.state.selectedSymbol,  inputValue = this.state.inputValue,  previousInputValue = this.state.previousInputValue;  if (!symbol) {  return;  }  this.setState({  previousInputValue: 0,  inputValue: eval(previousInputValue + symbol + inputValue),  selectedSymbol: null  });  break;  case 'C':  this.setState(this.initialState);  break;  *// case 'c':*  *// this.setState({inputValue: 0});*  *// break;*  }  }  } |

|  |
| --- |
| Button.js |
| import React, { Component } from 'react';  import {  TouchableHighlight,  Text  } from 'react-native';  import Style from './Style';  export default class Button extends Component {    render() {  return (  <TouchableHighlight style={[Style.inputButton, this.props.highlight ? Style.inputButtonHighlighted : null]}  underlayColor="#505050"  onPress={this.props.onPress}>  <Text style={Style.inputButtonText}>{this.props.value}</Text>  </TouchableHighlight>  )  }    } |

|  |
| --- |
| Style.js |
| import { StyleSheet } from 'react-native';  var Style = StyleSheet.create({  rootContainer: {  flex: 1  },  displayContainer: {  flex: 4,  backgroundColor: '#505050',  justifyContent: 'center',  },  displayText: {  color: 'white',  fontSize: 75,  fontWeight: 'lighter',  textAlign: 'right'  },    inputContainer: {  flex: 3,  backgroundColor: '#D4D4D2',  },  inputButton: {  flex: 1,  alignItems: 'center',  justifyContent: 'center',  borderWidth: 0.5,  borderColor: '#1C1C1C'  },  inputButtonHighlighted: {  backgroundColor: '#FF9500'  },  inputButtonText: {  fontSize: 30,  fontWeight: 'lighter',  color: 'black'  },  inputRow: {  flex: 1,  flexDirection: 'row'  }  });  export default Style; |

**Screenshot:**

 