

Lab 4 : JavaScript for Mobile Device Programming

- ให้นักศึกษาทำการสร้าง New Project ใหม่ ชื่อ ProjectCal ใน Folder Mobile\<รหัสนักศึกษา>\ProjectCal แบบ Blank Project โดยใช้คำสั่ง

Name	Date modified	Type	Size
.expo	9/9/2020 6:57 PM	File folder	
.expo-shared	9/9/2020 4:51 PM	File folder	
.vscode	9/9/2020 5:08 PM	File folder	
assets	9/9/2020 4:51 PM	File folder	
node_modules	9/9/2020 5:04 PM	File folder	
src	9/9/2020 4:51 PM	File folder	
.gitignore	9/9/2020 4:51 PM	Text Document	1 KB
App	9/9/2020 6:21 PM	JavaScript File	1 KB
app	9/9/2020 4:51 PM	JSON File	1 KB
babel.config	9/9/2020 4:51 PM	JavaScript File	1 KB
package	9/9/2020 4:51 PM	JSON File	1 KB
yarn.lock	9/9/2020 4:51 PM	LOCK File	254 KB

- ให้นักศึกษาทำการสร้างโปรแกรมเครื่องคิดเลขบนอุปกรณ์เคลื่อนที่ ดังแสดงตามรูปข้างล่างนี้



บันทึกผลการทดลอง :

```
App.js  Keyboard.js  Button.js
App.js > App > fontSize
1  import { StatusBar } from "expo-status-bar";
2  import React, { useState } from "react";
3  import { StyleSheet, Text, View } from "react-native";
4  import Keyboard from "../src/Keyboard";
5
6  export default function App() {
7    const [text, setText] = useState("");
8    const [result, setResult] = useState("");
9    return (
10     <View style={styles.container}>
11       <View style={{ flex: 2 }}>
12         <Text style={styles.text}>{text}</Text>
13         <Text style={{...styles.text, fontSize: 40}}>{result}</Text>
14       </View>
15       <Keyboard setText={setText} text={text} setResult={setResult} />
16     </View>
17   );
18 }
19
20 const styles = StyleSheet.create({
21   container: {
22     flex: 1,
23     backgroundColor: "#fff",
24     flexDirection: "column",
25     marginTop: 25,
26   },
27   text: {
28     textAlign: "right",
29     fontSize: 30,
30     margin: 10
31   }
32 });
33
```

```

src > .\App.js > [0] Keyboard
1  import React from "react";
2  import { StyleSheet, Text, View } from "react-native";
3  import Button from "../Button";
4
5  const Keyboard = ({ setText, text, setResult }) => {
6    const setInfix = (e, text1) => {
7      setText(text+text1)
8    };
9    const deleteText = (e) => {
10     if (text !== "") {
11       text = text.slice(0, text.length - 1);
12       setText(text);
13     }
14   };
15   const calculate = (e) => {
16     try {
17       const sum = eval(text);
18       setResult(sum);
19     } catch (e) {
20       if (e instanceof SyntaxError) {
21         setResult("Error")
22       }
23     }
24   };
25   const setDot = (e, text1) =>{
26     let isSet = false
27     text.split("").forEach((value, index, arr) =>{
28       if (value === "."){
29         isSet = false
30       }
31       else if (isNaN(value)){
32         isSet = true
33         if (index === arr.length - 1){
34           isSet = false
35         }
36       }
37     })
38     if (isSet){
39       setText(text+text1)
40     }
41   }

```

```

App.js Keyboard.js X Button.js
src > Keyboard.js > Keyboard
42   return (
43     <View style={styles.container}>
44       <View style={styles.col1}>
45         <Button text1={"1"} setInfix={setInfix} />
46         <Button text1={"4"} setInfix={setInfix} />
47         <Button text1={"7"} setInfix={setInfix} />
48         <Button text1={"."} setInfix={setDot} />
49       </View>
50       <View style={styles.col1}>
51         <Button text1={"2"} setInfix={setInfix} />
52         <Button text1={"5"} setInfix={setInfix} />
53         <Button text1={"8"} setInfix={setInfix} />
54         <Button text1={"0"} setInfix={setInfix} />
55       </View>
56       <View style={styles.col1}>
57         <Button text1={"3"} setInfix={setInfix} />
58         <Button text1={"6"} setInfix={setInfix} />
59         <Button text1={"9"} setInfix={setInfix} />
60         <Button text1={"="} setInfix={calculate} />
61       </View>
62       <View style={styles.col2}>
63         <Button text1={"Del"} setInfix={deleteText} />
64         <Button text1={"+"} setInfix={setInfix} />
65         <Button text1={"-"} setInfix={setInfix} />
66         <Button text1={"*"} setInfix={setInfix} />
67         <Button text1={"/"} setInfix={setInfix} />
68       </View>
69     </View>
70   );
71 };
72
73 const styles = StyleSheet.create({
74   container: {
75     flex: 3,
76     flexDirection: "row",
77   },
78   col1: {
79     flex: 1,
80     flexDirection: "column",
81     backgroundColor: "#444",
82   },
83   col2: {
84     flex: 1,
85     flexDirection: "column",
86     backgroundColor: "#333",
87   },
88 });
89
90 export default Keyboard;
91

```

```
JS App.js JS Keyboard.js JS Button.js X
src > JS Button.js > [0] Button
1  import React from "react";
2  import { StyleSheet, Text, View, TouchableOpacity } from "react-native";
3
4  const Button = ({ text1, setInfix }) => {
5    return (
6      <TouchableOpacity
7        style={styles.container}
8        onPress={(e) => setInfix(e, text1)}
9      >
10     <Text style={{ fontSize: 32, color: "white" }}>{text1}</Text>
11     </TouchableOpacity>
12   );
13 };
14
15 const styles = StyleSheet.create({
16   container: {
17     flex: 1,
18     alignItems: "center",
19     justifyContent: "center",
20   },
21 });
22
23 export default Button;
24
```