

Buổi 10 – React Native 3

Thiết kế giao diện bằng cách dùng vòng lặp

1. Nguyên tắc: Thiết kế theo ma trận hàng cột

	0	1	2	3
0				
1				
2				
3				
4				
5				

```
for(int i=0;i<6;i++)  
{  
  for(int j=0;j<4;j++)  
  {  
    push(đẩy dữ liệu);  
  }  
}
```

2. Vị trí bố trí vòng lặp:

Trong hàm **render();**

Ngoài hàm **return();**

3. Button trong React Native

TouchableWithoutFeedback

TouchableOpacity

Sự kiện trong button: **onPress**

```

10 //3. Vung thiet ke giao dien
11 render()
12 {
13     //Vi tri xu ly vong lap giao dien
14     //1.Tao cac button
15     let rows=[];//khai bao cac dong
16     let nums=[[1,2,3],[4,5,6],[7,8,9],['.','0','=']];//khai bao cac so de dien
17     for(let i=0;i<4;i++)//so dong
18     {
19         let row=[];//1 dong chua cac con so
20         for(let j=0;j<3;j++)
21         {
22             //dua cac button vao cac o
23             row.push(
24                 <TouchableOpacity style={styles.btn} key={nums[i][j]}>
25                     <Text style={styles.textBtn}>{nums[i][j]}</Text>
26                 </TouchableOpacity>
27             );
28         }
29         //dua row vao mang rows
30         rows.push(
31             <View style={styles.row} key={i}>{row}</View>
32         );

```

```

32     );
33 }
34 //2.Tao cac phep tinh
35 let ops=[];//mang cac phep tinh
36 let operator=['+','-','*','/','DEL'];
37 for(let i=0;i<5;i++)
38 {
39     ops.push(
40         <TouchableOpacity style={styles.btn} key={operator[i]}>
41             <Text style={styles.textBtn}>{operator[i]}</Text>
42         </TouchableOpacity>
43     );
44 }
45 //-----
46 return(
47     <View style={styles.container}>
48         <View style={styles.result}></View>
49         <View style={styles.calculation}></View>
50         <View style={styles.button}>
51             <View style={styles.numbers}>{rows}</View>
52             <View style={styles.operator}>{ops}</View>
53         </View>
54     </View>

```

