

RENDER VỚI IF – ELSE  
VÀ MẪNG

# Render if – else dùng toán tử so sánh 3 ngôi

- Cú pháp:  
Điều kiện so sánh ? Kết quả khi đúng : kết quả khi sai
- Ví dụ: Khi count == 5 thì hiển thị tên dạng:

John Walker

Count: 5

Press Here

# Render if – else dùng toán tử so sánh 3 ngôi

```
export default function DemoRenderIfElse() {  
  let name="John Walker"  
  const[count, setCount]=useState(0)  
  const onPress = () => setCount(count => count + 1);  
  return (  
    <View style={styles.container}>  
      <Text style={styles.name}>{count==5 ? name : ''}</Text>  
      .....  
    </View>  
  )  
}
```

# Render if – else dùng hàm

```
export default function DemoRenderIfElse() {  
  let name="John Walker"  
  const[count, setCount]=useState(0)  
  const onPress = () => setCount(count => count + 1);  
  _renderName={()=>{  
    return <Text style={styles.name}>{count==5 ? name : ''}</Text>  
  }  
  return (  
    <View style={styles.container}>  
      {_renderName()}  
      .....  
    </View>  
  )  
}
```

# Render if – else dùng hàm

Hoặc:

```
export default function DemoRenderIfElse() {  
  let name="John Walker"  
  const[count, setCount]=useState(0)  
  const onPress = () => setCount(count => count + 1);  
  _renderName=()=>(  
    <Text style={styles.name}>{count==5 ? name : ''}</Text>  
  )  
  return (  
    <View style={styles.container}>  
      {_renderName()}  
      .....  
    </View>  
  )  
}
```

# Render if – else dùng hàm

Hoặc:

```
export default function DemoRenderIfElse() {  
  let name="John Walker"  
  const[count, setCount]=useState(0)  
  const onPress = () => setCount(count => count + 1);  
  _renderName = () => {  
    if (count == 5)  
      return <Text style={styles.name}>{name}</Text>  
    else  
      return <Text style={styles.name}>No name</Text>  
  }  
  return (  
    <View style={styles.container}>  
      {_renderName()}  
      .....  
    </View>  
  )  
}
```

# Render dữ liệu mảng dùng hàm map

- Hàm map: duyệt qua từng phần tử của mảng và thực hiện 1 hàm nào đó trên từng phần tử. Sau đó trả về 1 mảng mới
- Cú pháp:  
**Tênmảng.map(hàm có 2 tham số: data và index)**
- Trong đó:
  - Data: giá trị của phần tử đang duyệt
  - Index: chỉ số của phần tử đang duyệt
- Ví dụ:

```
const a=[1, 2, 3, 4, 5]
const b=a.map((data, index)=>{return data + 5})
console.log(b)
```

# Render dữ liệu mảng dùng hàm map

```
import { View, Text } from 'react-native'
import React from 'react'

export default function DemoRenderArray() {
  const names=[
    'Lê Văn An',
    'Nguyễn Toàn Nhã',
    'Trần Thanh Thúy',
    'Lưu Diệc Phi'
  ]
  return (
    <View>
      <Text>DemoRenderArray</Text>
      {names.map((data, index) => (
        <Text>{data}</Text>
      )))}
    </View>
  )
}
```



# Render dữ liệu mảng dùng hàm map

Hoặc:

```
import { View, Text } from 'react-native'
import React from 'react'

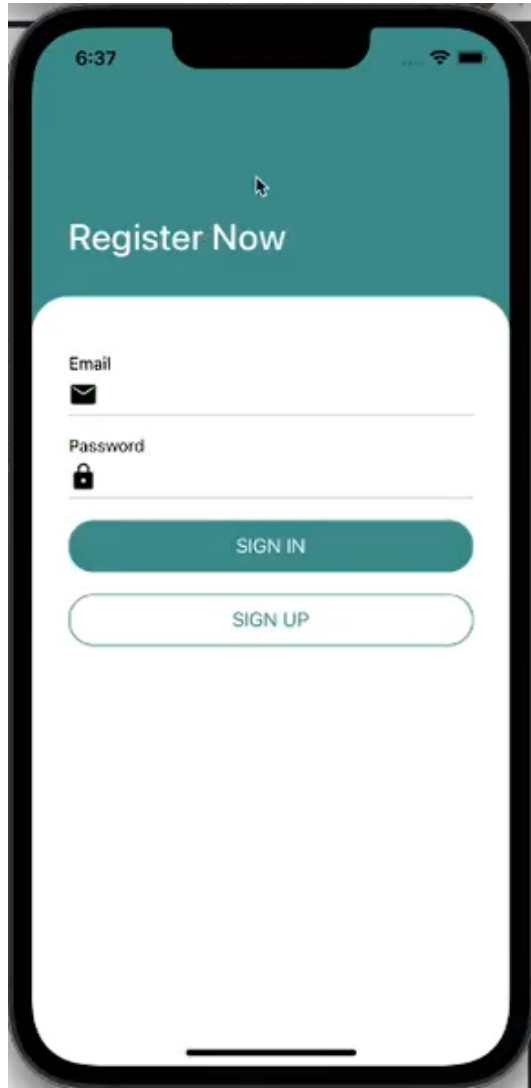
export default function DemoRenderArray() {
  const names=[
    'Lê Văn An',
    'Nguyễn Toàn Nhã',
    'Trần Thanh Thúy',
    'Lưu Diệc Phi'
  ]
  const _renderArray={()=>{
    return names.map((data, index) => (
      <Text style={{color:index==names.length-1 ? 'red' : 'black' }}>{data}</Text>
    ))
  }}
  return (
    <View>
      <Text>DemoRenderArray</Text>
      {_renderArray()}
    </View>
  )
}
```

# Render dữ liệu mảng dùng hàm map

Hoặc:

```
const _renderArray = () => {  
  return names.map((data, index) => {  
    let color = 'black'  
    if (index == names.length - 1) {  
      color = 'red'  
    }  
    return <Text style={{ color: color }}>{data}</Text>  
  })  
}  
return (  
  <View>  
    <Text>DemoRenderArray</Text>  
    {_renderArray()}  
  </View>  
)
```

# Excercise 1



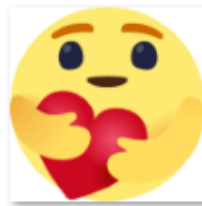
- Nút Sign in: return về 1 Text phía dưới cùng màn hình
  - Nếu Email là “user.demo.com” và Password là “demo” thì nội dung Text là: “Login successful”
  - Ngược lại: “Login failed”

# Excercise 2

Cho sẵn các file hình ảnh:



angry.png



care.png



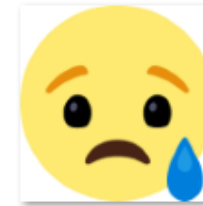
haha.png



like.png

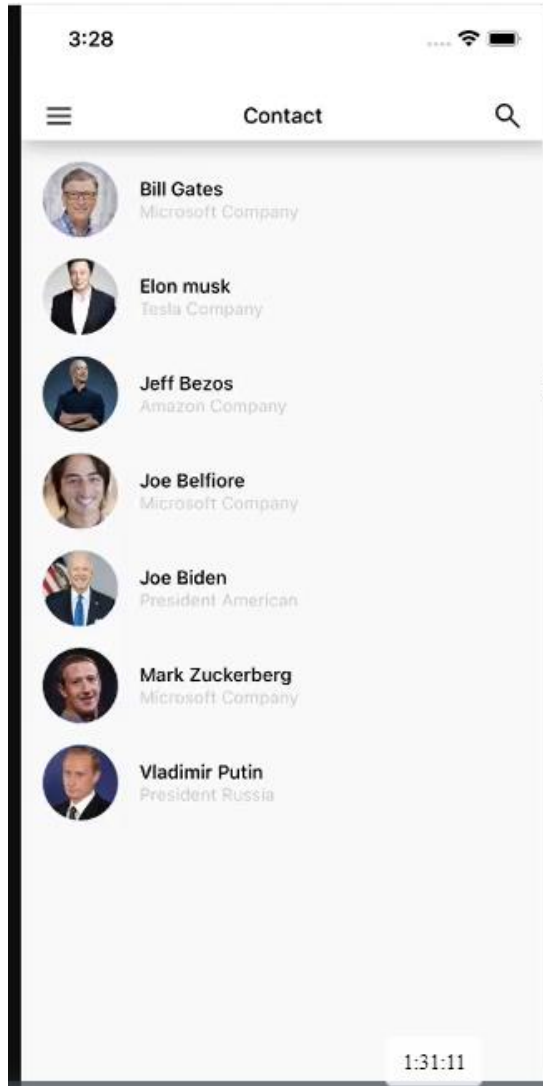


love.png



sad.png

# Excercise 3



```
datas = [  
  { title:"Bill Gates",    desc:"Microsoft Company",  
    image:require("../assets/images/billgates.jpeg")  },  
  { title:"Elon musk",    desc:"Tesla Company",  
    image:require("../assets/images/elon_musk.jpeg")  },  
  { title:"Jeff Bezos",    desc:"Amazon Company",  
    image:require("../assets/images/jeff.jpeg")  },  
  { title:"Joe Belfiore",    desc:"Microsoft Company",  
    image:require("../assets/images/joe_belfiore.webp")  },  
  { title:"Joe Biden",    desc:"President American",  
    image:require("../assets/images/joe_biden.jpeg")  },  
  { title:"Mark Zuckerberg",    desc:"Microsoft Company",  
    image:require("../assets/images/mark.jpeg")  },  
  { title:"Vladimir Putin",    desc:"President Russia",  
    image:require("../assets/images/putin.jpeg")  }  
]
```

Các file hình được cho sẵn

Yêu cầu: Viết riêng từng hàm cho từng phần của bố cục