

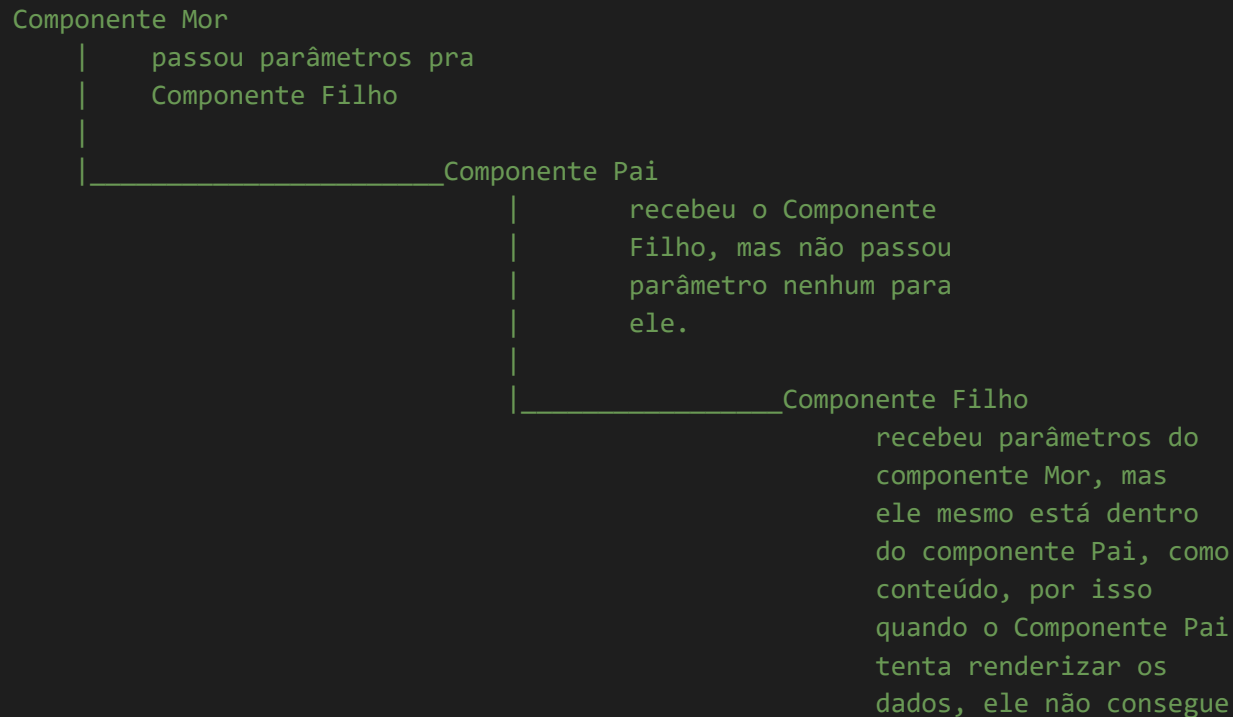
```
import React from 'react'
import {StyleSheet, Text, View} from 'react-native'
import Filhos from './Filhos'
```

```
/* PROPS.CHILDREN:
```

O props.children é uma forma de renderizarmos as propriedades que foram passadas para um Componente Filho que está dentro de um Componente Pai, porém os parâmetros foram passados por um Componente Mor que renderizou o Componente Pai, e não pelo Componente Pai, ficando impossível que o Componente Pai renderize as propriedades do Componente Filho.

Esquisito né?

Veja o diagrama embaixo e você vai entender...



Para resolver esse problema, usamos o props.children, veja como...

```
*/
```

```

export default props => {
  return (
    <View style={styles.background}>
      <Text style={styles.texto}>Família {props.familia}</Text>
      {/* Veja que o componente Família é o pai, que vai receber o componente Filhos, mas para renderizá-los, ele precisa de
      usar o props.children */}
      {props.children}
      {/* Veja que quando passamos valores diretamente a um filho ele renderiza sem problemas, ele só não renderiza quando é de
      um componente Mor para um filho. */}
      <Filhos nome="Intruso" familia='Intruso' />
    </View>
  )
}

const styles = StyleSheet.create({

  background: {
    margin: 20,
  },
  texto: {
    color: 'black',
    fontSize: 24,
    fontWeight: 'bold',
    marginBottom: 10,
    marginTop: 30,
  }
})

```

COMPONENTE FILHOS.JS...

```

import React from "react";
import { StyleSheet, Text } from "react-native";

export default ({nome, familia}) => {

```

```

const intruso = 'Intruso'
return (
  nome !== intruso
    ? <Text style={styles.texto}>{nome} {familia}</Text>
    : <Text style={styles.intruso}>{nome} {familia}</Text>
  )
}

const styles = StyleSheet.create({

  texto: {
    color: 'black',
    fontSize: 16,
  },
  intruso: {
    color: 'red',
    fontSize: 16,
  },
})

```

ARQUIVO DO APP.JS...

```

import React from 'react'
import Familia from './src/componentes/propsPontoChildren/Familia'
import Filhos from './src/componentes/propsPontoChildren/Filhos'

const familia1 = "Silva"
const familia2 = "Gusmão"

export default () => {
  return (
    /* Coloque seus componentes abaixo...*/
    <>
      { /* Neste caso App.js faz o papel de componente Mor */}
      <Familia familia={familia1}>

```

```
    { /* Note que os componentes Filhos estão sendo passados como conteúdo... */}  
    <Filhos nome="Isadora" familia={familia1} />  
    <Filhos nome="Felipe" familia={familia1} />  
    <Filhos nome="Adriano" familia={familia1} />  
  </Familia>  
  <Familia familia={familia2}>  
    <Filhos nome="Anabela" familia={familia2} />  
    <Filhos nome="Raique" familia={familia2} />  
    <Filhos nome="Janderson" familia={familia2} />  
  </Familia>  
</>  
)  
}
```

RESULTADO NO CELULAR...

The image shows a mobile application running on an emulator and the Visual Studio Code editor with the source code.

Mobile App Preview:

- Família Silva**
 - Isadora Silva
 - Felipe Silva
 - Adriano Silva
 - Intruso Intruso
- Família Gusmão**
 - Anabela Gusmão
 - Raique Gusmão
 - Janderson Gusmão
 - Intruso Intruso

Visual Studio Code Editor:

EXPLORADOR

- CURSO-DE-REACT
 - node_modules
 - src
 - componentes
 - consoles
 - detalhesComponentes
 - detalhesPropriedades
 - propsPontoChildren
 - JS Familia.js
 - JS Filhos.js
 - ThisIsButton
 - ThisIsFragment
 - ThisIsPlataform
 - ThisIsSafeAreaView
 - estilos
 - exercicios
 - fundamentos
 - truques-na-manga \ Rend...
 - JS RenderizacaoCondiciona...
 - .buckconfig
 - .eslintrc.js
 - .flowconfig
 - .gitignore
 - .node-version
 - .prettierrc.js
 - .ruby-version
 - .watchmanconfig
 - JS App.js

JS App.js

```
4 import React, { useState } from 'react';
5 const familia1 = "Silva";
6 const familia2 = "Gusmão";
7
8 export default () => {
9   return (
10     /* Coloque seus componentes abaixo... */
11     <>
12       /* Neste caso App.js faz o papel de componente Mor... */
13       <Familia familia={familia1}>
14         /* Note que os componentes Filhos estão sendo passados como conteúdo... */
15         <Filhos nome="Isadora" familia={familia1}/>
16         <Filhos nome="Felipe" familia={familia1}/>
17         <Filhos nome="Adriano" familia={familia1}/>
18       </Familia>
19       <Familia familia={familia2}>
20         <Filhos nome="Anabela" familia={familia2}/>
21         <Filhos nome="Raique" familia={familia2}/>
22         <Filhos nome="Janderson" familia={familia2}/>
23       </Familia>
24     </>
25   );
26 }
```

TERMINAL

```
BUILD SUCCESSFUL in 4m 49s
38 actionable tasks: 2 executed, 36 up-to-date
info Connecting to the development server...
info Starting the app on "emulator-5554"...
Starting: Intent { cmp=com.componentes/.MainActivity }
```