

Projeto Loja básica com React Native

- Crie um novo projeto no <https://snack.expo.dev/>, em meus My snakes. Chamado “projeto-loja-basica”.
- Renomear o arquivo App.js, para App.tsx
- Vá a pasta components e apague os arquivos existentes nela.
- Vá até a pasta assets e apague o conteúdo dela.
- Você deve está com o ambiente igual a imagem abaixo.



- Vá até a pasta do seu computador, c:\LanschoolFile e descompacte o arquivo Loja.zip.
- Neste zip têm dois arquivos, data.ts e hero.jpg. Faça o upload dos dois para o seu App.
- Substitua o View, por um SafeAreaView. Aproveite e já importe o componente e um StatusBar também;
- Altere o código para o código abaixo, cuidado com os erros de digitação.

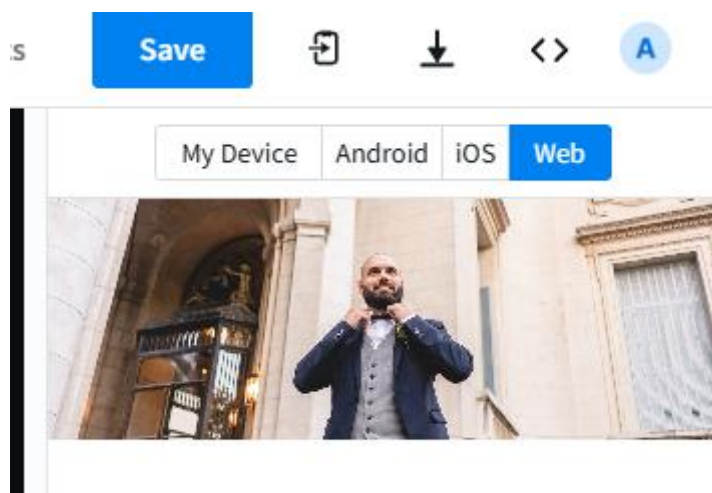
```
import { View, SafeAreaView, StatusBar, StyleSheet, Image } from 'react-native';

function App() {
  return (
    <SafeAreaView style={styles.container}>
      <Image
        source={require('./assets/hero.jpg')}
        resizeMode="cover"
        style={styles.hero}
      />
    </SafeAreaView>
  );
}

const styles = StyleSheet.create({
  container: {
    marginTop: StatusBar.currentHeight || 0
  },
  hero: {
    width: '100%',
    height: 120
  }
});

export default App;
```

- O Resultado deve ser o mesmo da imagem abaixo:



- Se tudo ocorreu bem, vamos para o mesmo passo, inserir o nome dos produtos e valor.
- Realize as importações necessárias para prosseguir como mostra a imagem abaixo

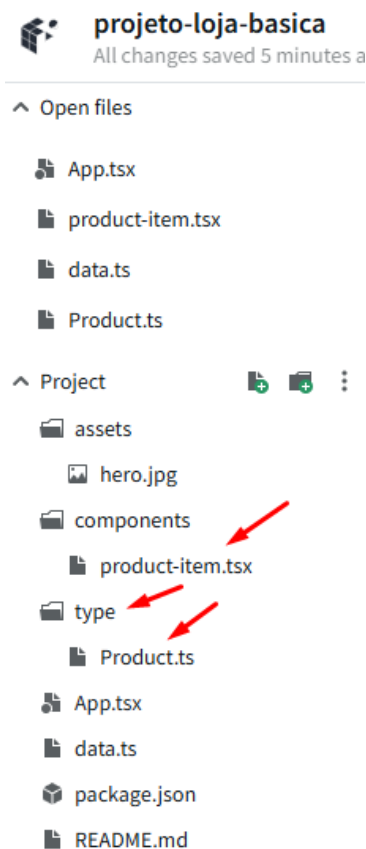
```

1  import { View, SafeAreaView, StatusBar, StyleSheet, Image, Text, FlatList } from 'react-native';
2
3  import { list } from './data';
4
5  function App() {
6    return (
7      <SafeAreaView style={styles.container}>
8
9        <Image
10         source={require('./assets/hero.jpg')}
11         resizeMode="cover"
12         style={styles.hero}
13       />
14
15       <View style={styles.area}>
16         <Text style={styles.h1}>Produto</Text>
17         <FlatList
18           data={list}
19           renderItem={() => {
20             keyExtractor={item => item.id.toString()}
21           }}
22         />
23       </View>
24     </SafeAreaView>
25   );
26 }
27


```

- Em seguida altere o seu código da linha 15 a 22.

- Agora crie uma pasta chamada "type" e nela um arquivo chamado "Product.ts", em seguida vá a pasta componentes e crie um arquivo chamado "product-item.tsx"



- Em seguida realize as alterações abaixo:

 **projeto-loja-basica**
All changes saved 6 minutes ago. [See previous saves.](#) ✓

Open files


- App.tsx
- product-item.tsx
- data.ts
- Product.ts

Project

- assets
 - hero.jpg
- components
 - product-item.tsx
- type
 - Product.ts
- App.tsx

```
export type Product = {  
  id: number;  
  image: string;  
  name: string;  
  price: number;  
}
```

- Em product-item.tsx:

 **projeto-loja-basica**
All changes saved 7 minutes ago. [See previous saves.](#) ✓

Open files

- App.tsx
- product-item.tsx
- data.ts
- Product.ts

Project

- assets
 - hero.jpg
- components
 - product-item.tsx

```
1 import { View } from 'react-native'  
2  
3 export const ProductItem = () => {  
4   return (  
5     <View>  
6  
7     </View>  
8   );  
9 }
```

- Agora vá no App.tsx e realize as alterações:

```
1  import { View, SafeAreaView, StatusBar, StyleSheet, Image, Text, FlatList } from 'react-native';
2
3  import { list } from './data';
4
5  import { ProductItem } from './components/product-item';
6
7  import { Product } from './type/Product';
8
9  function App() {
10   return (
11     <SafeAreaView style={styles.container}>
12       <Image
13         source={require('./assets/hero.jpg')}
14         resizeMode="cover"
15         style={styles.hero}
16       />
17
18       <View style={styles.area}>
19         <Text style={styles.h1}>Produto</Text>
20
21         <FlatList
22           data={list}
23           renderItem={({item}: {item: Product }) => (<ProductItem />)}
24           keyExtractor={item => item.id.toString()}
25         />
26       </View>
27     </SafeAreaView>
28   );
29 }
30
31
32 }
```

- Por fim, realize as últimas alterações no App.tsx

```
8
9  export const ProductItem = (props: Props) => {
10    const handleAddToCart = () => {
11      alert("Você está comprando o produto:" + props.product.name);
12    }
13
14
15    return (
16      <View style={styles.container}>
17        <Image
18          source={{ uri: props.product.image }}
19          style={styles.image}
20        />
21
22        <View style={styles.info}>
23
24          <Text style={styles.name}> {props.product.name}</Text>
25          <Text style={styles.price}> R$ {props.product.price.toFixed(2)}</Text>
26
27          <Button
28            title="Comprar"
29            onPress={handleAddToCart}
30          />
31        </View>
32      </View>
33    );
34  }
35
36 }
```

- No Styles

```
38  const styles = StyleSheet.create({
39    container: {
40      flexDirection: 'row',
41      marginBottom: 20
42    },
43    image: {
44      width: 100,
45      height: 100
46    },
47    info: {
48      flex: 1,
49      marginLeft: 20
50    },
51    name: {
52      fontWeight: 'bold',
53      fontSize: 21,
54      marginBottom: 10
55    },
56    price: {
57      color: '#FF0000',
58      fontSize: 16,
59      marginBottom: 10
60    }
61  });
```