```
 1 package com.example.calculadoraimc
 2
 3 import android.os.Bundle
 4 import androidx.activity.ComponentActivity
 5 import androidx.activity.compose.setContent
 6 import androidx.activity.enableEdgeToEdge
 7 import androidx.compose.foundation.background
 8 import androidx.compose.foundation.layout.*
 9 import androidx.compose.foundation.text.
   BasicTextField
10 import androidx.compose.foundation.text.
   KeyboardOptions
11 import androidx.compose.material3.Button
12 import androidx.compose.material3.Scaffold
13 import androidx.compose.material3.Text
14 import androidx.compose.runtime.*
15 import androidx.compose.ui.Alignment
16 import androidx.compose.ui.Modifier
17 import androidx.compose.ui.graphics.Color
18 import androidx.compose.ui.text.input.KeyboardType
19 import androidx.compose.ui.tooling.preview.Preview
20 import androidx.compose.ui.unit.dp
21 import androidx.compose.ui.unit.sp
22 import com.example.calculadoraimc.ui.theme.
   CalculadoraimcTheme
23 import androidx.compose.foundation.Image
24 import androidx.compose.ui.res.painterResource
25 import com.example.calculadoraimc.R
26
27
28 class MainActivity : ComponentActivity() {
29     override fun onCreate(savedInstanceState: Bundle
   ?) {
30         super.onCreate(savedInstanceState)
31         enableEdgeToEdge()
32         setContent {
33             CalculadoraimcTheme {
34                 Scaffold(modifier = Modifier.
   fillMaxSize()) { innerPadding ->
35                     IMCCalculator(modifier = Modifier
   .padding(innerPadding))
```

```
36                 }
37             }
38         }
39     }
40 }
41
42 @Composable
43 fun IMCCalculator(modifier: Modifier = Modifier) {
44     var altura by remember { mutableStateOf("") }
45     var peso by remember { mutableStateOf("") }
46     var resultado by remember { mutableStateOf("") }
47
48      Box(
49         modifier = modifier
50             .fillMaxSize()
51             .background(Color.Cyan)  // Define o
   fundo branco
52             .padding(16.dp),
53         contentAlignment = Alignment.TopCenter  //
   Centraliza no topo
54     ) {
55         Column(
56             modifier = Modifier
57                 .fillMaxWidth()
58                 .padding(top = 50.dp),  //
   Espaçamento para posicionar mais para cima
59             horizontalAlignment = Alignment.
   CenterHorizontally,
60             verticalArrangement = Arrangement.
   spacedBy(16.dp)  // Espaçamento entre os itens
61         ) {
62             Text(
63                 text = "Calculadora de IMC",
64                 fontSize = 24.sp,
65                 modifier = Modifier.padding(bottom =
   16.dp)
66             )
67
68             // Campo de entrada para altura
69             BasicTextField(
70                 value = altura,
```

```
 71                  onValueChange = { altura = it },
 72                  modifier = Modifier
 73                      .fillMaxWidth(0.8f)
 74                      .padding(8.dp),
 75                  keyboardOptions = KeyboardOptions(
    keyboardType = KeyboardType.Number),
 76                  decorationBox = { innerTextField ->
 77                      Box(
 78                          Modifier
 79                              .fillMaxWidth()
 80                              .padding(8.dp)
 81                      ) {
 82                          if (altura.isEmpty()) {
 83                              Text("Digite sua altura
     (m)", fontSize = 16.sp)
 84                          }
 85                          innerTextField()
 86                      }
 87                  }
 88              )
 89
 90              // Campo de entrada para peso
 91              BasicTextField(
 92                  value = peso,
 93                  onValueChange = { peso = it },
 94                  modifier = Modifier
 95                      .fillMaxWidth(0.8f)
 96                      .padding(8.dp),
 97                  keyboardOptions = KeyboardOptions(
    keyboardType = KeyboardType.Number),
 98                  decorationBox = { innerTextField ->
 99                      Box(
100                          Modifier
101                              .fillMaxWidth()
102                              .padding(8.dp)
103                      ) {
104                          if (peso.isEmpty()) {
105                              Text("Digite seu peso (
    kg)", fontSize = 16.sp)
106                          }
107                          innerTextField()
```

```
108                     }
109                 }
110             )
111 
112             // Botão para calcular IMC
113             Button(
114                 onClick = {
115                     val imc = calcularIMC(altura, peso)
116                     resultado = when {
117                         imc < 18.5 -> "Seu IMC é $imc: Abaixo do peso"
118                         imc in 18.5..24.9 -> "Seu IMC é $imc: Peso normal"
119                         imc in 25.0..29.9 -> "Seu IMC é $imc: Sobrepeso"
120                         else -> "Seu IMC é $imc: Obesidade"
121                     }
122                 },
123                 modifier = Modifier.fillMaxWidth(0.6f)
124             ) {
125                 Text("Calcular IMC", fontSize = 16.sp)
126             }
127 
128             // Exibição do resultado
129             Text(
130                 text = resultado,
131                 fontSize = 18.sp,
132                 modifier = Modifier.padding(top = 16.dp)
133             )
134 
135             // Exibição da imagem centralizada
136             Image(
137                 painter = painterResource(id = R.drawable.foto1),
138                 contentDescription = "Tabela de IMC",
```

```
139                     modifier = Modifier
140                         .fillMaxWidth(0.8f)
141                         .padding(top = 16.dp)
142                 )
143             }
144         }
145 }
146
147 // Função para calcular o IMC
148 fun calcularIMC(altura: String, peso: String):
    Double {
149     val alturaDouble = altura.toDoubleOrNull() ?:
    return 0.0
150     val pesoDouble = peso.toDoubleOrNull() ?: return
     0.0
151     return pesoDouble / (alturaDouble * alturaDouble
    )
152 }
153
154 @Preview(showBackground = true)
155 @Composable
156 fun IMCCalculatorPreview() {
157     CalculadoraimcTheme {
158         IMCCalculator()
159     }
160 }
161
162
163
```