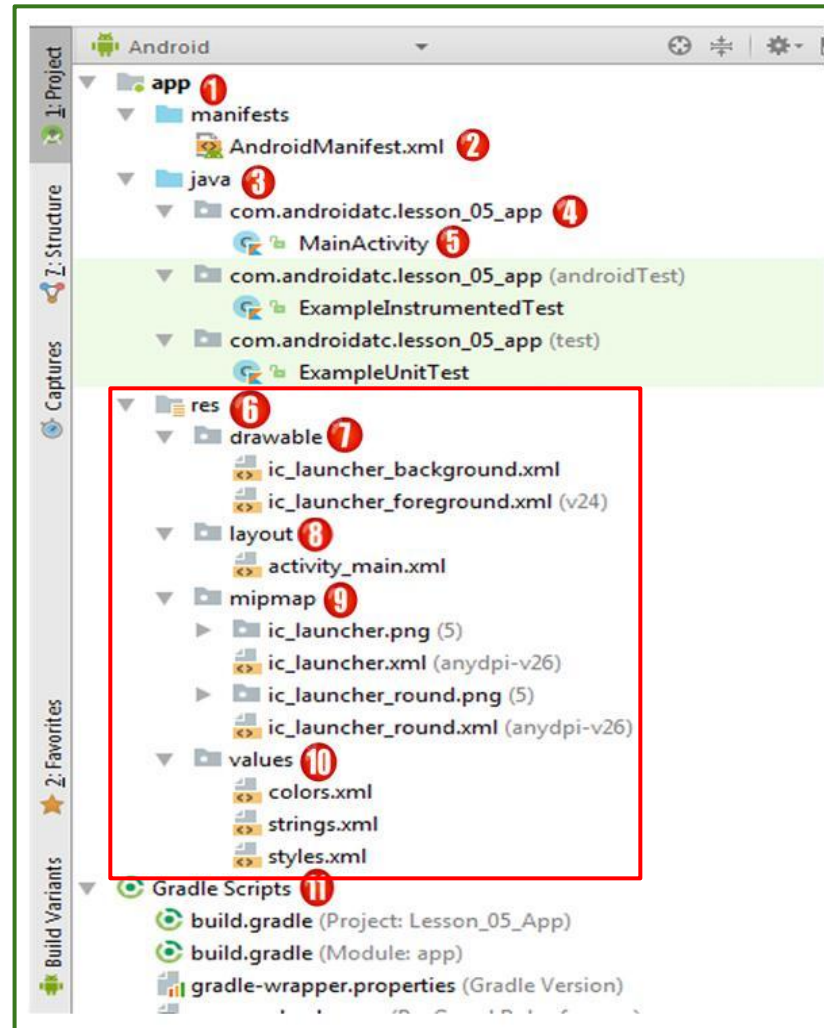


Desenvolvimento de Interfaces Android

Análise e Desenvolvimento de Software - 2020 / 4

Visão Geral dos Recursos



Visão Geral dos Recursos

```
MyProject/  
  src/  
    MainActivity.java  
  res/  
    drawable/  
      graphic.png  
    layout/  
      main.xml  
      info.xml  
    mipmap/  
      icon.png  
    values/  
      strings.xml
```

Recursos são os arquivos adicionais e o conteúdo estático usado pelo seu código, como bitmaps, definições de layout, strings da interface do usuário, instruções de animação, entre outras coisas.

Você deve fornecer recursos alternativos para configurações específicas do dispositivo, agrupando-os em diretórios especialmente nomeados.

Ao exteriorizar os recursos do aplicativo, é possível acessá-los usando códigos que são gerados na **classe R** do projeto.



<https://developer.android.com/guide/topics/resources/providing-resources?hl=pt-PT>

Visão Geral dos Recursos

Diretório	Tipo de recurso
color/	Arquivos XML que definem uma lista de estado de cores. Consulte Recurso de lista de estado de cores
drawable/	<p>Os arquivos Bitmap (.png, .9.png, .jpg, .gif) ou arquivos XML são compilados nos seguintes subtipos de recurso drawable:</p> <ul style="list-style-type: none">• Arquivos Bitmap• Nine-Patch (bitmaps redimensionáveis)• Listas de estado• Formatos• Drawables de animação• Outros drawables <p>Veja Recursos desenháveis.</p>
mipmap/	São arquivos drawable para diferentes densidades do ícone na tela de início. Para mais informações sobre o gerenciamento de ícones na tela de início com pastas mipmap/, consulte Visão geral do gerenciamento de projetos .
layout/	Arquivos XML que definem um layout de interface do usuário. Consulte Recurso de layout .

Visão Geral dos Recursos

Diretório	Tipo de recurso
values/	<p>São arquivos XML que contêm valores simples, como strings, números inteiros e cores.</p> <p>Enquanto os arquivos de recurso XML que estão em outros subdiretórios <code>res/</code> definem um único recurso com base no nome do arquivo XML, os arquivos no diretório <code>values/</code> descrevem vários recursos. Para cada arquivo neste diretório, cada filho do elemento <code><resources></code> define um único recurso. Por exemplo: um elemento <code><string></code> cria um recurso <code>R.string</code> e um elemento <code><color></code> cria um recurso <code>R.color</code>.</p> <p>Como cada recurso é definido com seu próprio elemento XML, é possível nomear o arquivo da forma que quiser e colocar tipos de recurso variados em um arquivo. No entanto, para esclarecer, você pode querer colocar tipos de recursos únicos em arquivos diferentes. Por exemplo, veja algumas convenções de nome de arquivo para recursos que podem ser criados neste diretório:</p> <ul style="list-style-type: none">• <code>arrays.xml</code> para matrizes de recurso (matrizes digitadas).• <code>colors.xml</code> para valores de cor• <code>dimens.xml</code> para valores de dimensão.• <code>strings.xml</code> para valores de string.• <code>styles.xml</code> para estilos. <p>Consulte Recursos de string, Recurso de estilo e Mais tipos de recursos.</p>

Calculadora de Gorjeta

Total da Conta:

Quantas Pessoas?

Gorjeta:

10%

Valor Gorjeta:

Total:

Total Por Pessoa:

Total da Conta

btxtTotalConta

Quantas Pessoas?

btxtPessoas

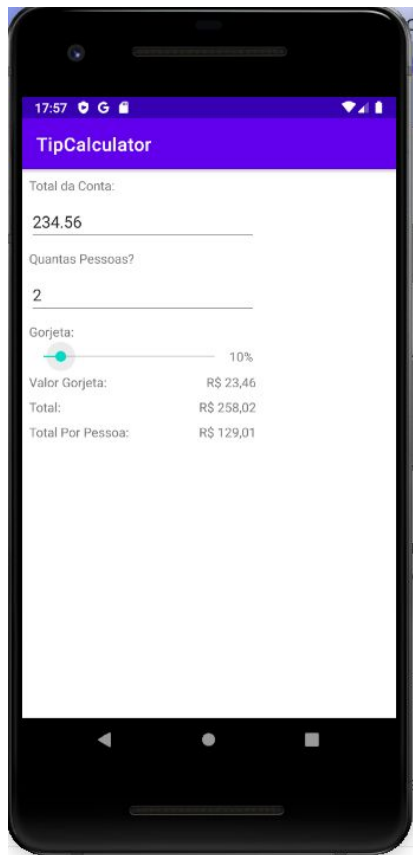
Gorjeta:

10%

Valor Gorjeta:

Total:

Total Por Pessoa:



Calculadora de Gorjeta

```
MainActivity.kt x
13 class MainActivity : AppCompatActivity(), View.OnFocusChangeListener, SeekBar.OnSeekBarChangeListener
14
15 //-----
16 private lateinit var txtTotalConta : EditText
17 private lateinit var txtPessoas : EditText
18 private lateinit var skGorjeta : SeekBar
19 //-----
20 private var formatador = NumberFormat.getCurrencyInstance()
```

Calculadora de Gorjeta

```
MainActivity.kt x
55  override fun onFocusChange(p0: View?, p1: Boolean) {
56      1      TODO( reason: "Not yet implemented")
57  }
58
59  override fun onProgressChanged(p0: SeekBar?, p1: Int, p2: Boolean) {
60      TODO( reason: "Not yet implemented")
61  }
62
63  override fun onStartTrackingTouch(p0: SeekBar?) {
64      2      TODO( reason: "Not yet implemented")
65  }
66
67  override fun onStopTrackingTouch(p0: SeekBar?) {
68      TODO( reason: "Not yet implemented")
69  }
70  }
```


Calculadora de Gorjeta

```
MainActivity.kt x
22  override fun onCreate(savedInstanceState: Bundle?) {
23      super.onCreate(savedInstanceState)
24      setContentView(R.layout.activity_main)
25      //-----
26      txtTotalConta = this.findViewById<EditText>(R.id.txtTotalConta)
27      txtTotalConta.setOnFocusChangeListener(this)
28      //-----
29      txtPessoas = this.findViewById<EditText>(R.id.txtPessoas)
30      txtPessoas.setOnFocusChangeListener(this)
31      //-----
32      skGorjeta = this.findViewById<SeekBar>(R.id.skGorjeta)
33      skGorjeta.setOnSeekBarChangeListener(this)
34      //-----
35  }
```

Calculadora de Gorjeta

```
MainActivity.kt
37 private fun atualizaDadosConta() {
38
39     1 if (txtTotalConta.text.toString().isEmpty()
40         && txtPessoas.text.toString().isEmpty()) {
41
42         var valorConta = txtTotalConta.text.toString().toDouble()
43         var qtdPessoas = txtPessoas.text.toString().toInt()
44
45         2 //-----
46         var lblValorRealGorjeta = this.findViewById<TextView>(R.id.lblValorRealGorjeta)
47         var valorRealGorjeta = valorConta * skGorjeta.progress / 100
48         lblValorRealGorjeta.setText(formatador.format(valorRealGorjeta))
49
50         3 //-----
51         var lblValorRealConta = this.findViewById<TextView>(R.id.lblValorRealConta)
52         lblValorRealConta.setText(formatador.format(number: valorConta + valorRealGorjeta))
53         var lblValorRealPorPessoa = this.findViewById<TextView>(R.id.lblValorRealPorPessoa)
54         lblValorRealPorPessoa.setText(formatador.format(number: (valorConta + valorRealGorjeta) / qtdPessoas))
55     }
```

Calculadora de Gorjeta

```
MainActivity.kt
57 override fun onFocusChange(p0: View?, p1: Boolean) {
58     1 this.atualizaDadosConta()
59 }
60
61 override fun onProgressChanged(p0: SeekBar?, p1: Int, p2: Boolean) {
62
63     2 var lblPercentualGorjeta = this.findViewById<TextView>(R.id.lblPercentualGorjeta)
64     lblPercentualGorjeta.setText(skGorjeta.progress.toString() + "%")
65     //-----
66     this.atualizaDadosConta()
67 }
68
69 override fun onStartTrackingTouch(p0: SeekBar?) {
70 }
71
72 3 override fun onStopTrackingTouch(p0: SeekBar?) {
73 }
74 }
```



Exercício

1. Fazer funcionar as suas calculadoras do Copacabana Runners.
2. Implementar a Calculadora de Gorjeta apresentada nesta aula.