



Reskilling 4Employment Software Developer

Acesso móvel a sistemas de informação

Bruno Santos

bruno.santos.mcv@msft.cesae.pt

Tópicos

- LinearLayout

Tipos de layout

- Existem dois tipos de layout principais:
 - RelativeLayout
 - LinearLayout

RelativeLayout

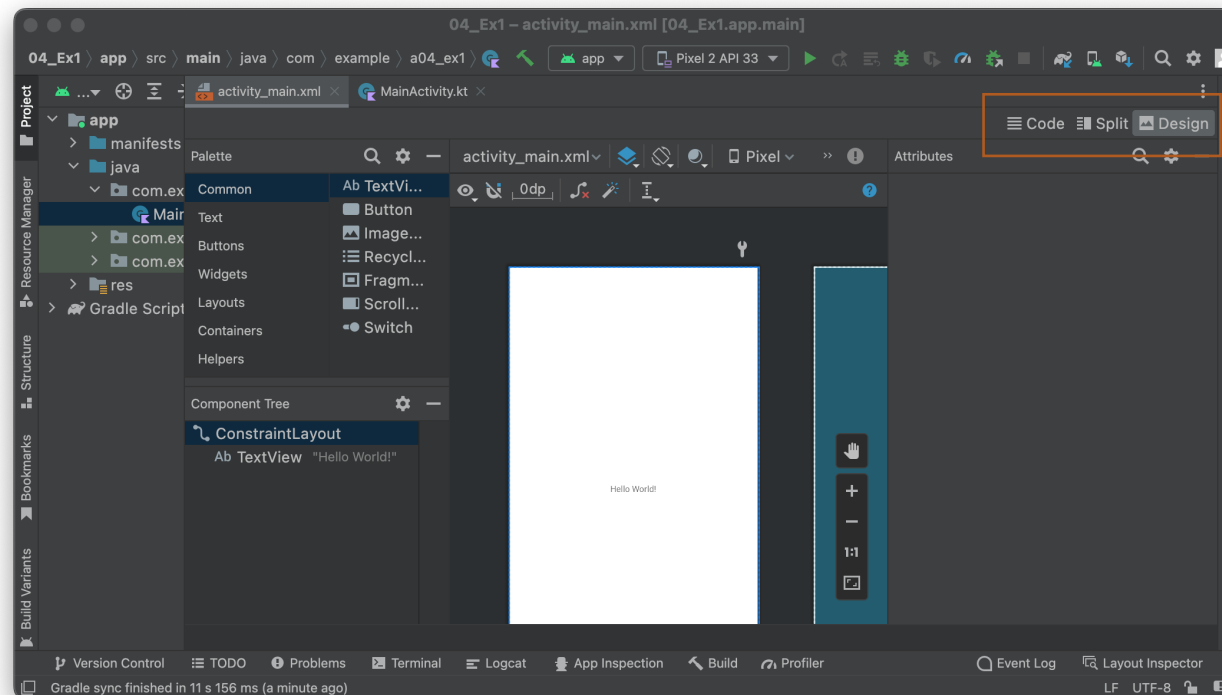
- No caso do RelativeLayout os elementos estão relacionados uns com os outros, sendo que cada um apresenta a sua posição relativamente a outros elementos

LinearLayout

- No caso do LinearLayout os elementos estão relacionados com o layout em si e são colocados segundo uma orientação (orientation) que pode ser vertical ou horizontal.
- Vamos começar com este tipo de layout!

LinearLayout

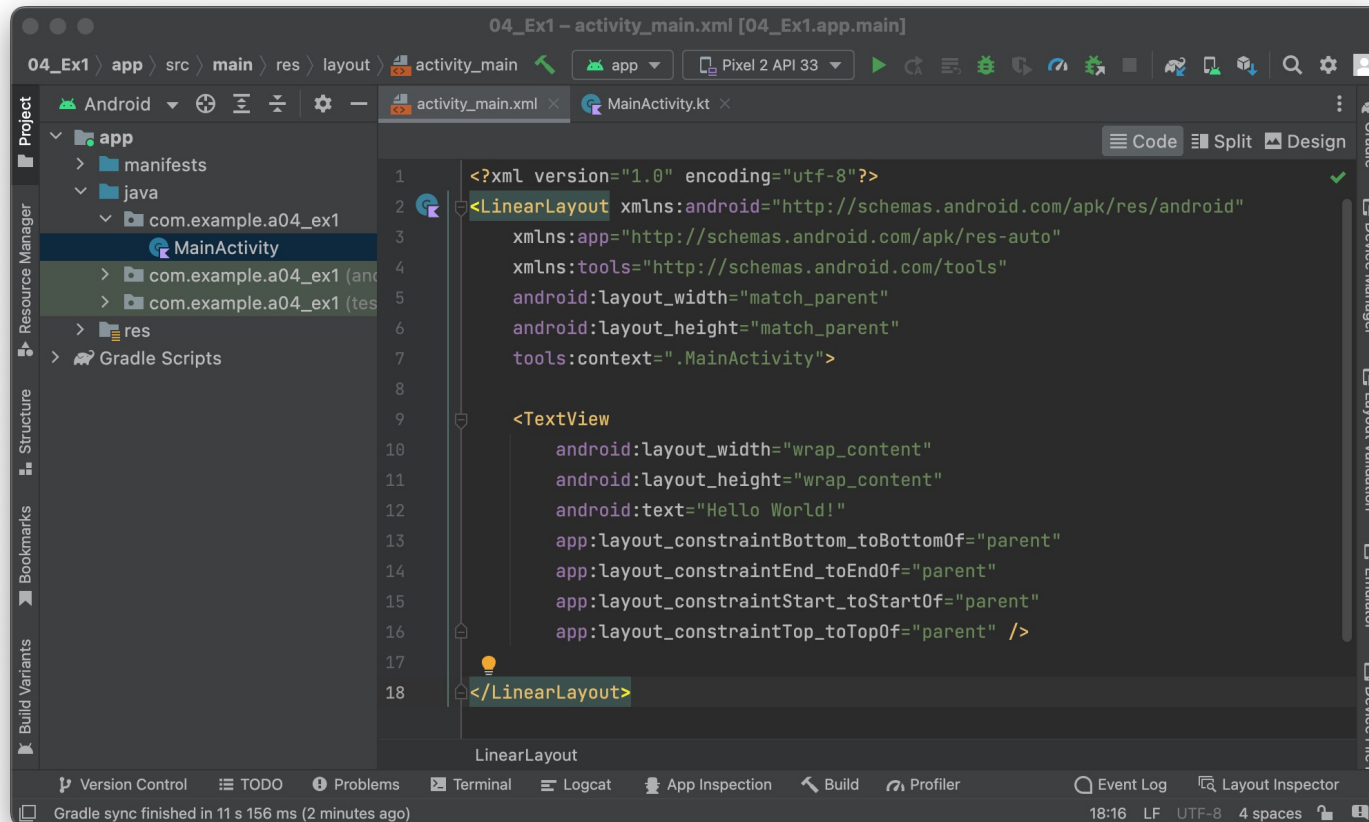
- Numa nova aplicação vamos alterar no ficheiro `activity_main.xml` a vista para Code:



LinearLayout

- Alteramos a linha:
`android.support.constraint.ConstraintLayout`
- Para apenas:
`LinearLayout`

LinearLayout



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

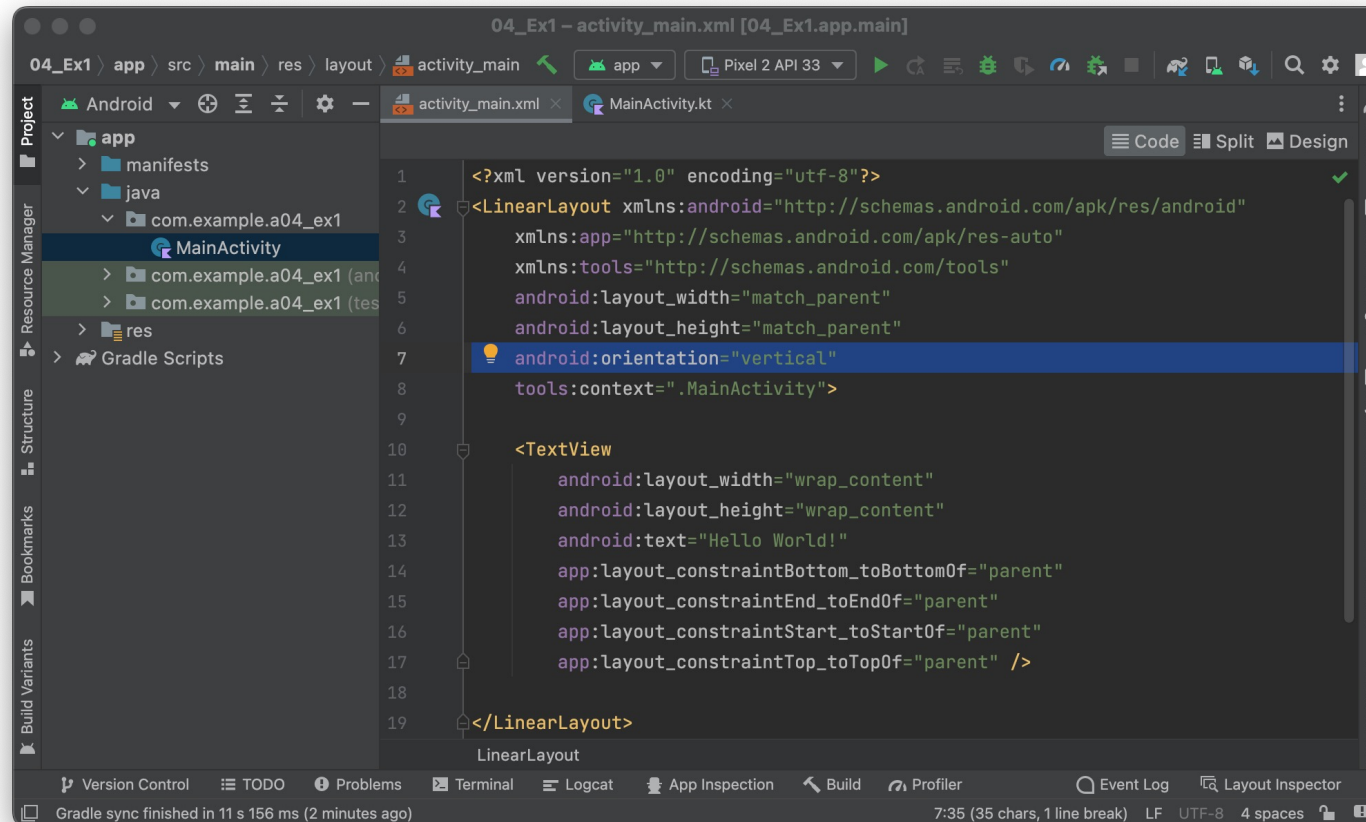
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</LinearLayout>
```


LinearLayout

- Acrescentamos dentro das propriedades do LinearLayout a propriedade orientation e colocamos a opção vertical (pode ser vertical ou horizontal).

LinearLayout



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello World!"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

</LinearLayout>
```

LinearLayout

- Dentro do LinearLayout temos um elemento criado: TextView, vamos eliminá-lo e criar um novo elemento, também TextView, para isso basta escrever <TextView e dar Enter.
- Automaticamente foi criado o seguinte código:

```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"
```

LinearLayout

- As duas propriedades essenciais de cada elemento são o width (largura) e o height (altura). Para cada uma das propriedades podemos selecionar as opções:
 - match_parent – ocupa todo o espaço do elemento pai
 - wrap_content – ocupa o espaço necessário para apresentar o conteúdo do elemento

LinearLayout

- Antes de alterar as propriedades anteriores vamos criar a propriedade text e atribuir-lhe o valor “Isto é uma TextView” e o alinhamento (gravity) para center para serem percetíveis as alterações.
- Vamos alterar largura e altura para wrap_content, para ocupar apenas o tamanho necessário para o elemento.

LinearLayout



```
<TextView  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:gravity="center"  
    android:text="Isto é uma TextView" />
```

LinearLayout

- Agora vamos alterar apenas a largura para `match_parent`, para ocupar todo o elemento pai (LinearLayout).

LinearLayout



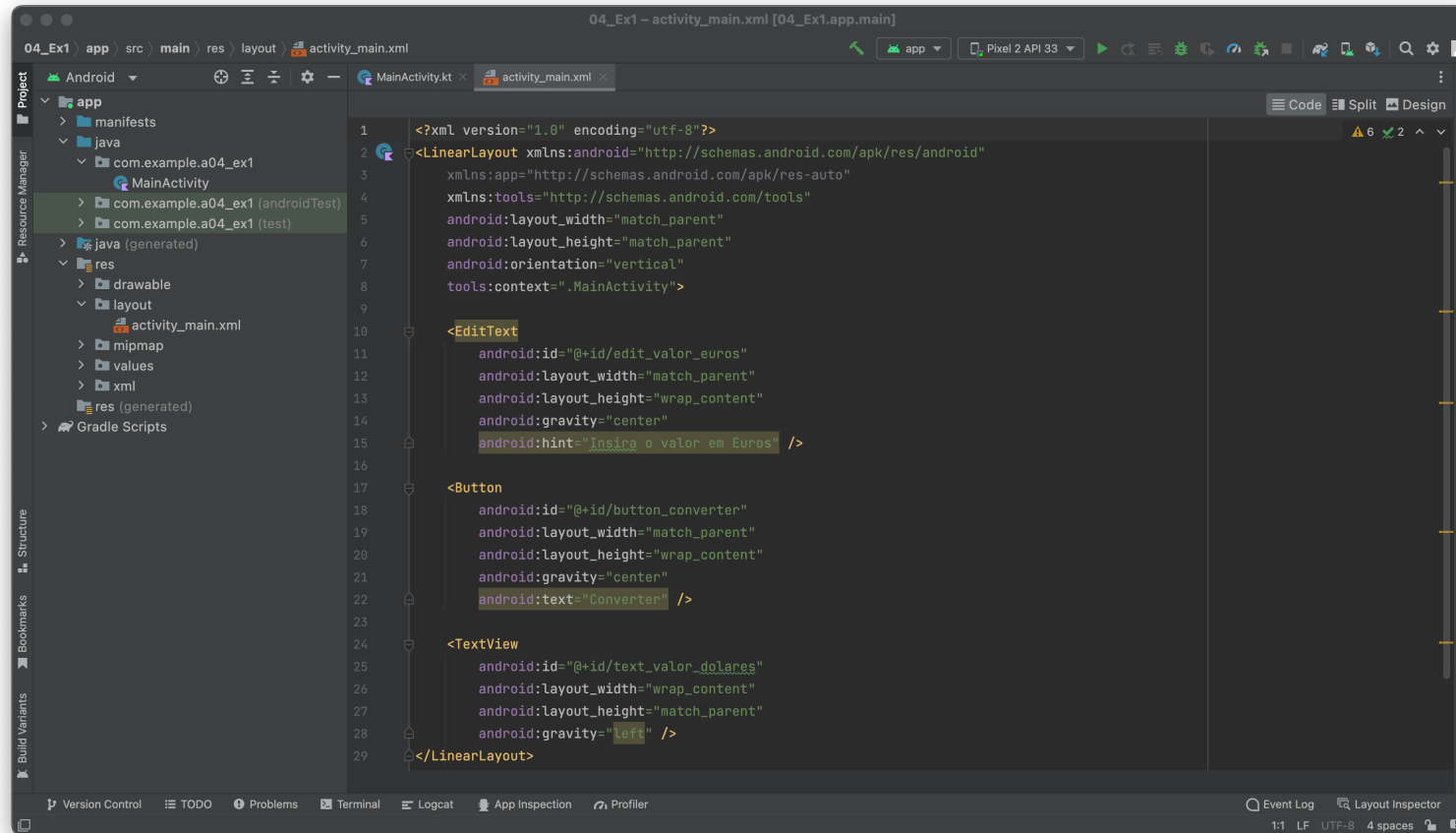
```
<TextView  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:gravity="center"  
    android:text="Isto é uma TextView" />
```



Exercício 1

- Vamos recriar o exercício do conversor de Euro para Dólar colocando os vários elementos no layout:
- 1 EditText
 - ID: edit_valor_euros (importante deve aparecer @+id/edit_valor_euros)
 - hint: Insira o valor em Euros
 - gravity: center
 - layout_width: match_parent
 - layout_height: wrap_content
- 1 Button:
 - ID: button_converter
 - text: Converter
 - gravity: center
 - layout_width: match_parent
 - layout_height: wrap_content
- 1 TextView:
 - ID: text_valor_dolares
 - gravity: left
 - layout_width: wrap_content
 - layout_height: wrap_content

Exercício 1

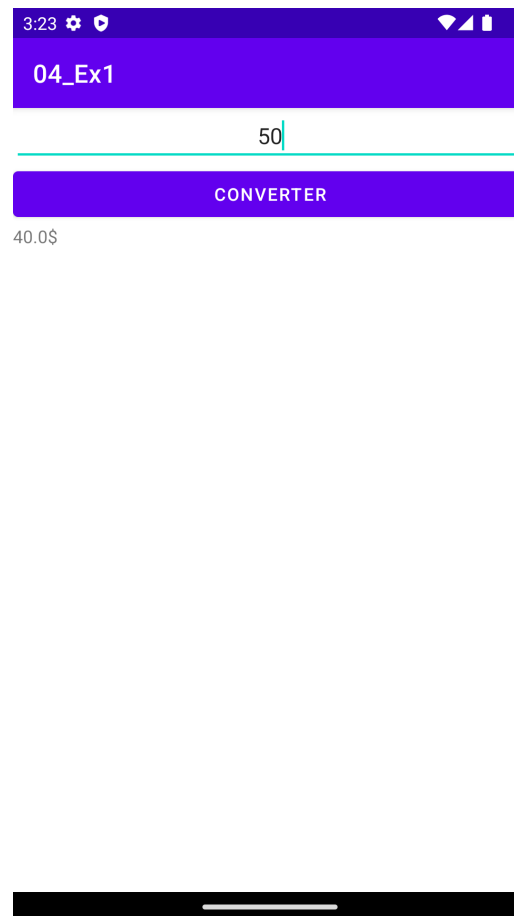


```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3   xmlns:app="http://schemas.android.com/apk/res-auto"
4   xmlns:tools="http://schemas.android.com/tools"
5   android:layout_width="match_parent"
6   android:layout_height="match_parent"
7   android:orientation="vertical"
8   tools:context=".MainActivity">
9
10  <EditText
11    android:id="@+id/edit_valor_euros"
12    android:layout_width="match_parent"
13    android:layout_height="wrap_content"
14    android:gravity="center"
15    android:hint="Insira o valor em Euros" />
16
17  <Button
18    android:id="@+id/button_converter"
19    android:layout_width="match_parent"
20    android:layout_height="wrap_content"
21    android:gravity="center"
22    android:text="Converter" />
23
24  <TextView
25    android:id="@+id/text_valor_dolares"
26    android:layout_width="wrap_content"
27    android:layout_height="match_parent"
28    android:gravity="left" />
29 </LinearLayout>
```

Exercício 1

- Utilize o conteúdo do ficheiro MainActivity.kt para programar a aplicação e teste a mesma.

Exercício 1

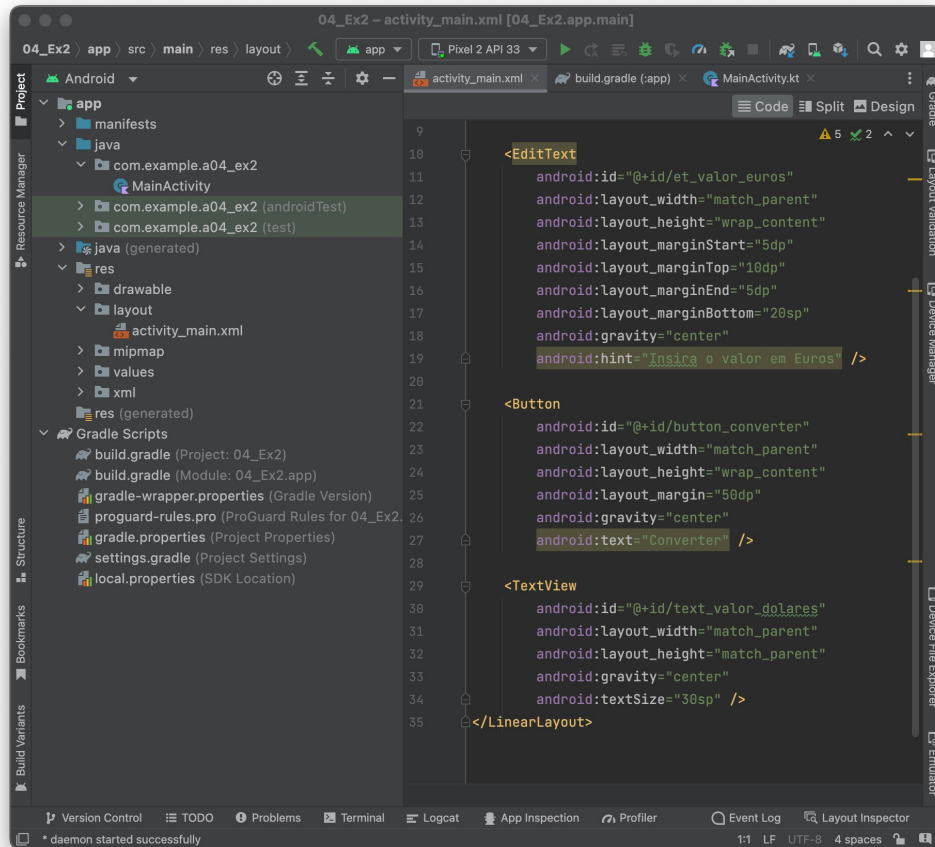


The screenshot shows a mobile application interface with a dark blue background. At the top, a status bar displays the time 3:23, a gear icon for settings, a shield icon for security, and signal/battery indicators. Below the status bar, a dark blue header contains the text "04_Ex1". A light blue input field is positioned below the header, containing the number "50" with a green cursor. A light blue button labeled "CONVERTER" is located below the input field. At the bottom of the screen, the text "40.0\$" is displayed. A black horizontal bar with a white line is visible at the very bottom, representing the home indicator on an iPhone.

Exercício 2

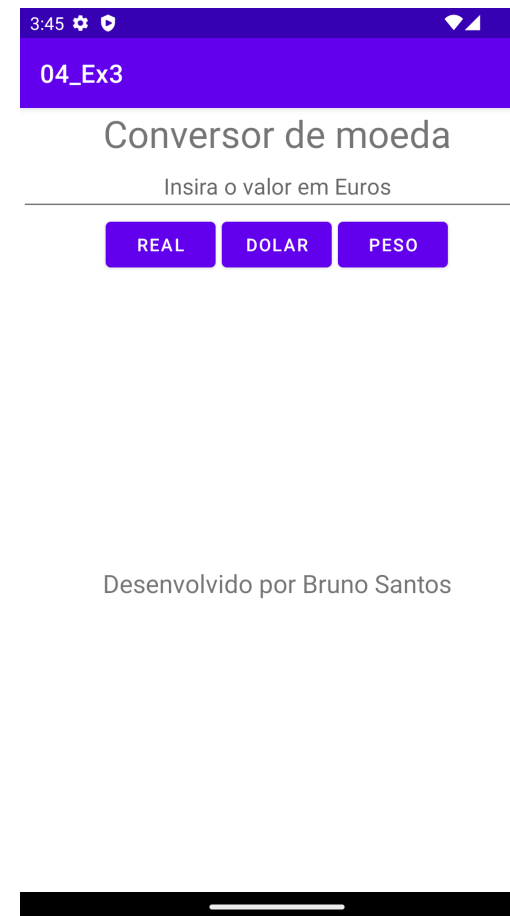
- Altere ou crie uma nova aplicação semelhante à anterior e realize as seguintes operações:
 1. Acrescente as seguintes margens ao elemento EditText:
 1. Top: 10dp
 2. Start: 5dp
 3. End: 5dp
 4. Bottom: 20dp
 2. Acrescente uma margem a toda a volta de 50dp ao elemento Button
 3. Altere os parâmetros `layout_width` e `layout_height` to TextView para `match_parent`, o parâmetro `gravity` para `center` e acrescente o `textSize` com o valor 30sp.

Exercício 2



LinearLayout

- Vamos criar um novo conversor de moeda que permita converter um valor em Euros para Reais, Dólares ou Pesos, com um layout semelhante ao seguinte:



LinearLayout

- Antes de mais, os três Button presentes no layout não estão ordenados na vertical como todos os outros elementos, mas sim na horizontal, isto é explicado pela utilização de um segundo LinearLayout dentro do primeiro LinearLayout.

LinearLayout

- Com esta disposição estamos a incluir o segundo LinearLayout como um dos elementos na vertical do primeiro layout e a garantir que todos os elementos dentro dele estejam na horizontal.
- Acrescente dentro do primeiro layout os seguintes elementos por ordem:

LinearLayout

- TextView
 - layout_width: match_parent
 - layout_height: wrap_content
 - text: Conversor de Moeda
 - gravity: center
 - textSize: 30sp
- EditText
 - ID: edit_valor_euros
 - layout_width: match_parent
 - layout_height: wrap_content
 - hint: Insira o valor em euros
 - gravity: center

LinearLayout

- LinearLayout
 - layout_width: match_parent
 - layout_height: wrap_content
 - orientation: horizontal
 - gravity: center
- TextView
 - layout_width: match_parent
 - layout_height: match_parent
 - text: Desenvolvido por: <nome do formando>
 - gravity: center
 - textSize: 30sp

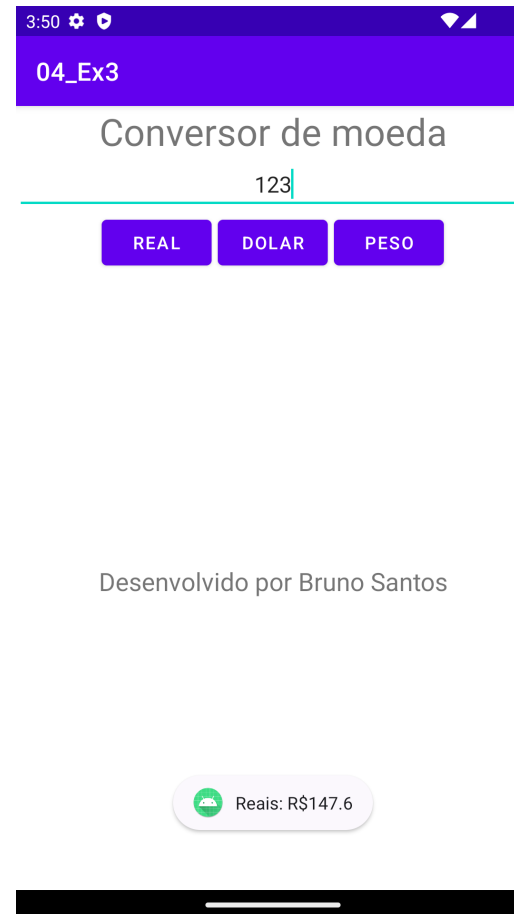
LinearLayout

- Dentro do segundo LinearLayout acrescente os seguintes elementos por ordem:
- Button
 - ID: button_real
 - layout_width: wrap_content
 - layout_height: wrap_content
 - text: Real
- Button
 - ID: button_dolar
 - layout_width: wrap_content
 - layout_height: wrap_content
 - margin_horizontal: 5sp
 - text: Dolar
- Button
 - ID: button_peso
 - layout_width: wrap_content
 - layout_height: wrap_content
 - text: Peso

Exercício 3

- Programe o conversor de moeda de forma a que preenchido o valor em Euros, quando clicado num dos três botões apareça num Toast o respetivo valor em reais, dólares ou pesos.
- Considere como taxa de conversão:
- 1 euro (€) = 4,4 reais (R\$)
- 1 euro (€) = 1,20 dólares (\$)
- 1 euro (€) = 31,5 pesos (\$)

Exercício 3



Desenvolvido por Bruno Santos

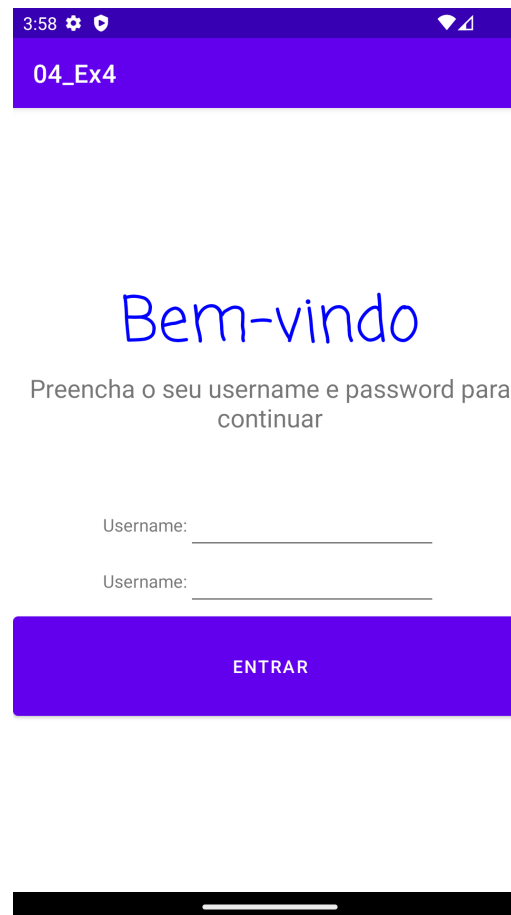
Exercício 4

- Crie uma nova aplicação que apresente ao utilizador um formulário de criação de registo numa aplicação. O layout deve ficar semelhante ao do slide seguinte.
- Aquando do clique no botão o login deve ser validado. Caso o username seja user e a password seja pass deve aparecer uma mensagem a dizer login válido, caso contrário deve aparecer login inválido. Em ambos os casos o texto presente nos campos de username e password devem ser apagados.

Exercício 4

- Na primeira TextView o fontFamily é casual, o textColor é azul (ver código hexadecimal) e o textSize 50sp.
- Na segunda TextView o textSize é 20sp e tem uma margem abaixo de 50dp
- Os dois EditText têm como minWidth 200dp
- O EditText da password tem como inputType o valor textPassword
- O Button tem um padding de 30dp

Exercício 4



3:58

04_Ex4

Bem-vindo

Preencha o seu username e password para continuar

Username:

Username:

ENTRAR