



# Reskilling 4Employment Software Developer

Acesso móvel a sistemas de informação

Bruno Santos

[bruno.santos.mcv@msft.cesae.pt](mailto:bruno.santos.mcv@msft.cesae.pt)

# Tópicos

- LinearLayout

# Tipos de layout

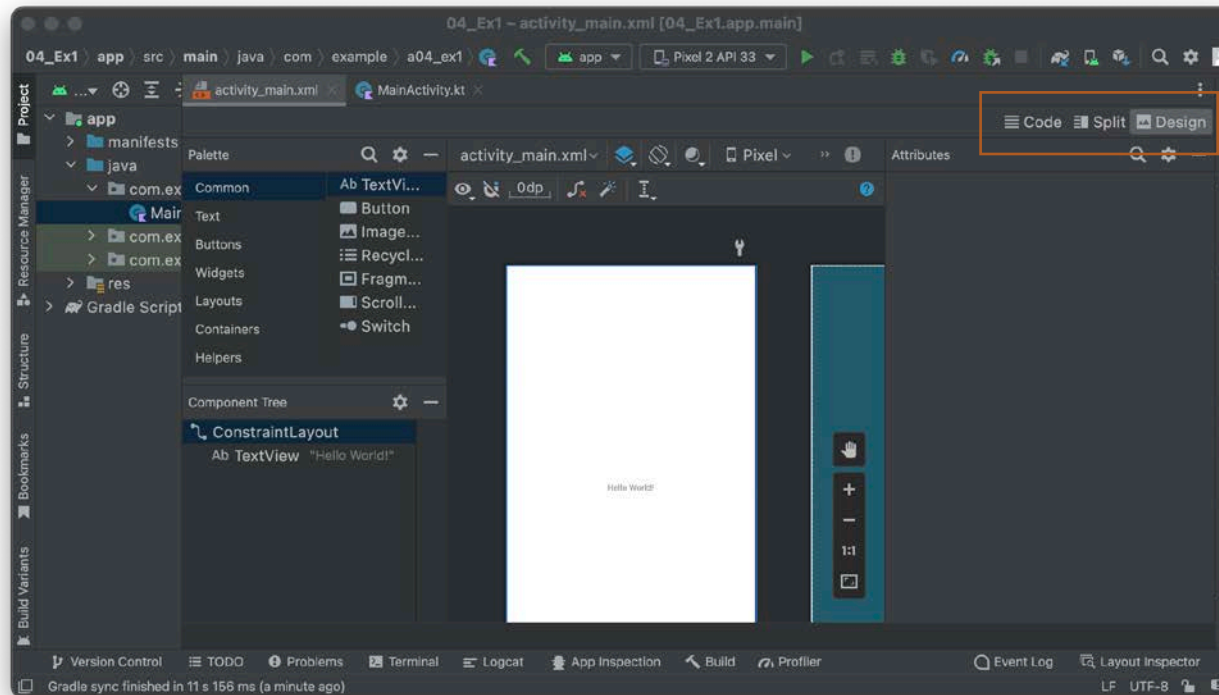
- Existem vários tipos de layout, como exemplo:
  - ConstraintLayout
  - RelativeLayout
  - **LinearLayout**
  - FrameLayout

# LinearLayout

- No 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

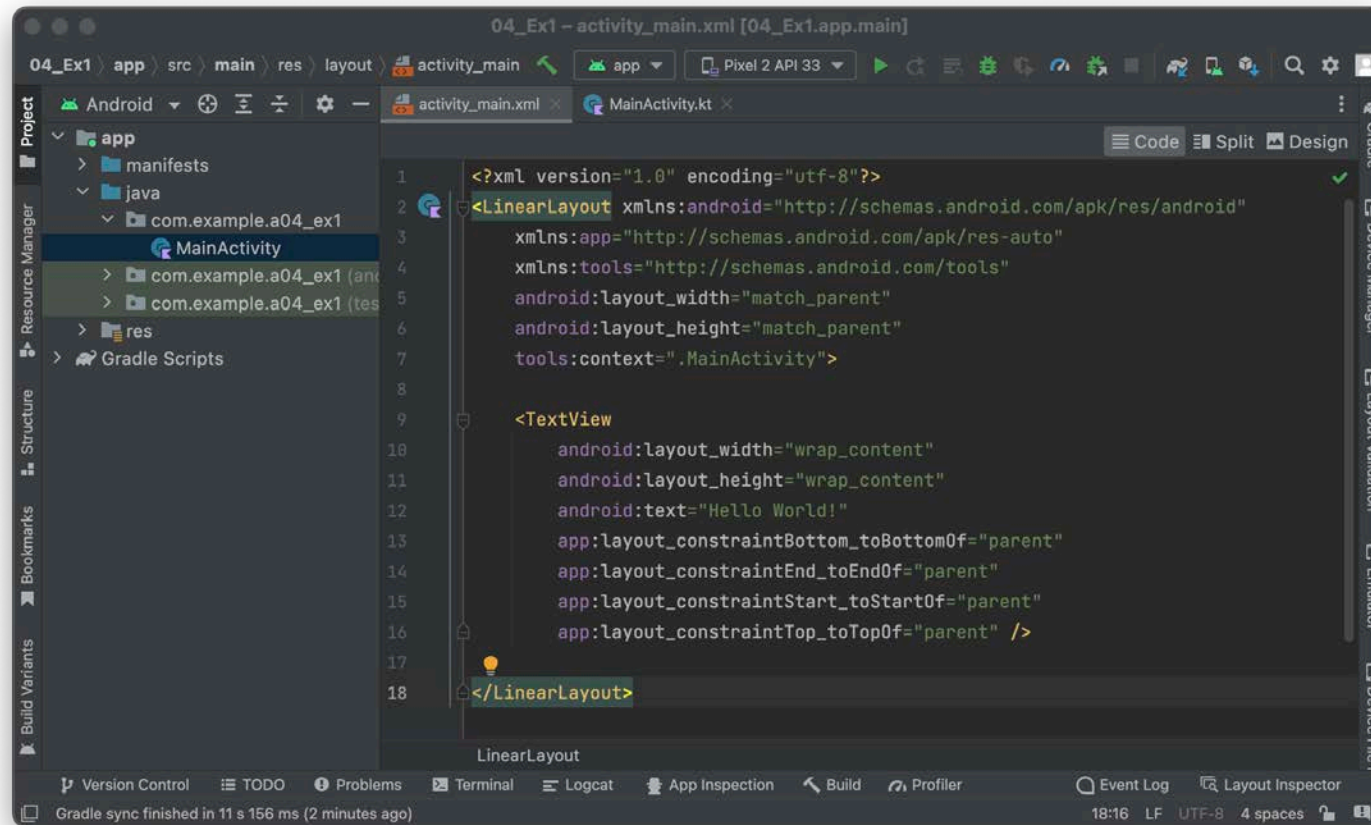
- Alteramos a linha:

`android.support.constraint.ConstraintLayout`

- Para apenas:

`LinearLayout`

# LinearLayout

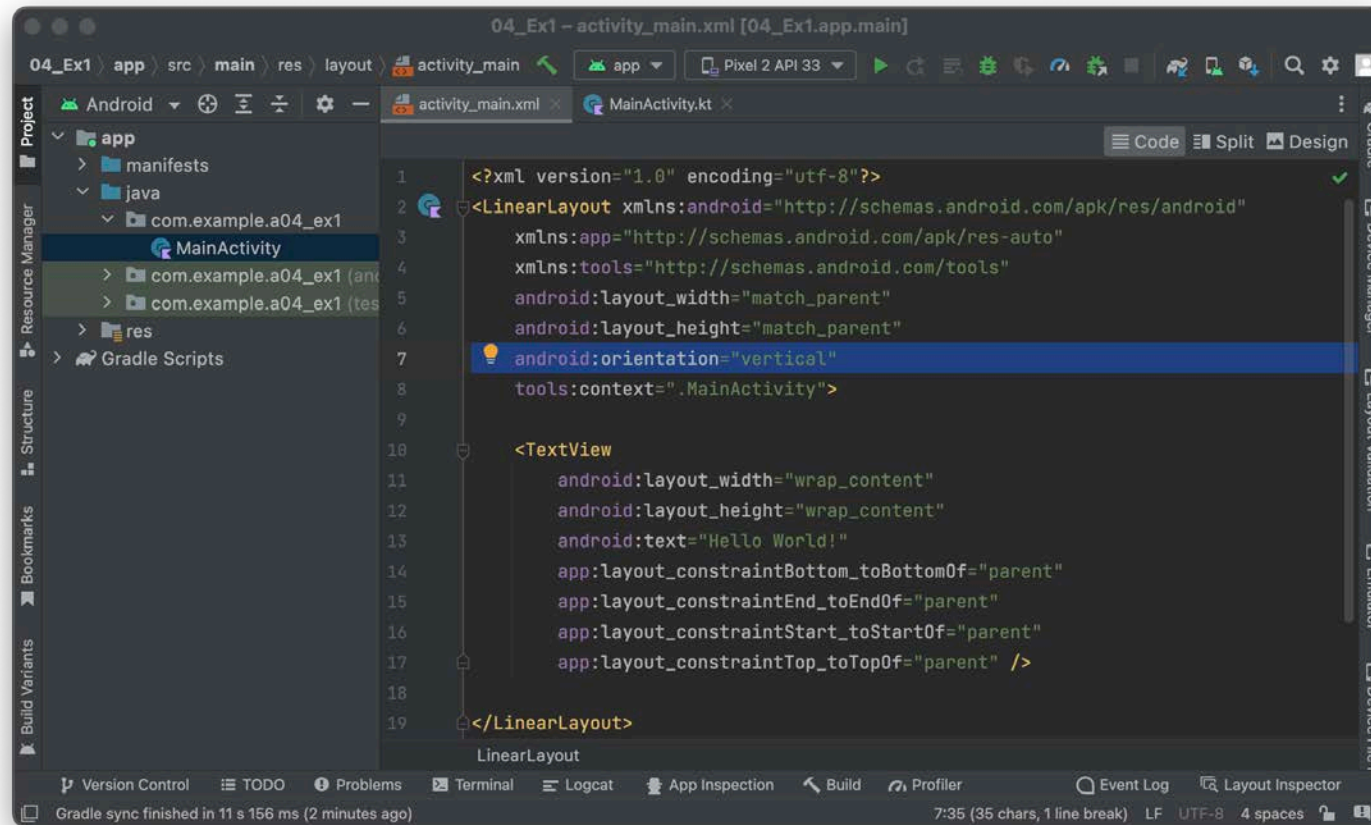


# LinearLayout

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



# 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 perceptíveis 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" />
```

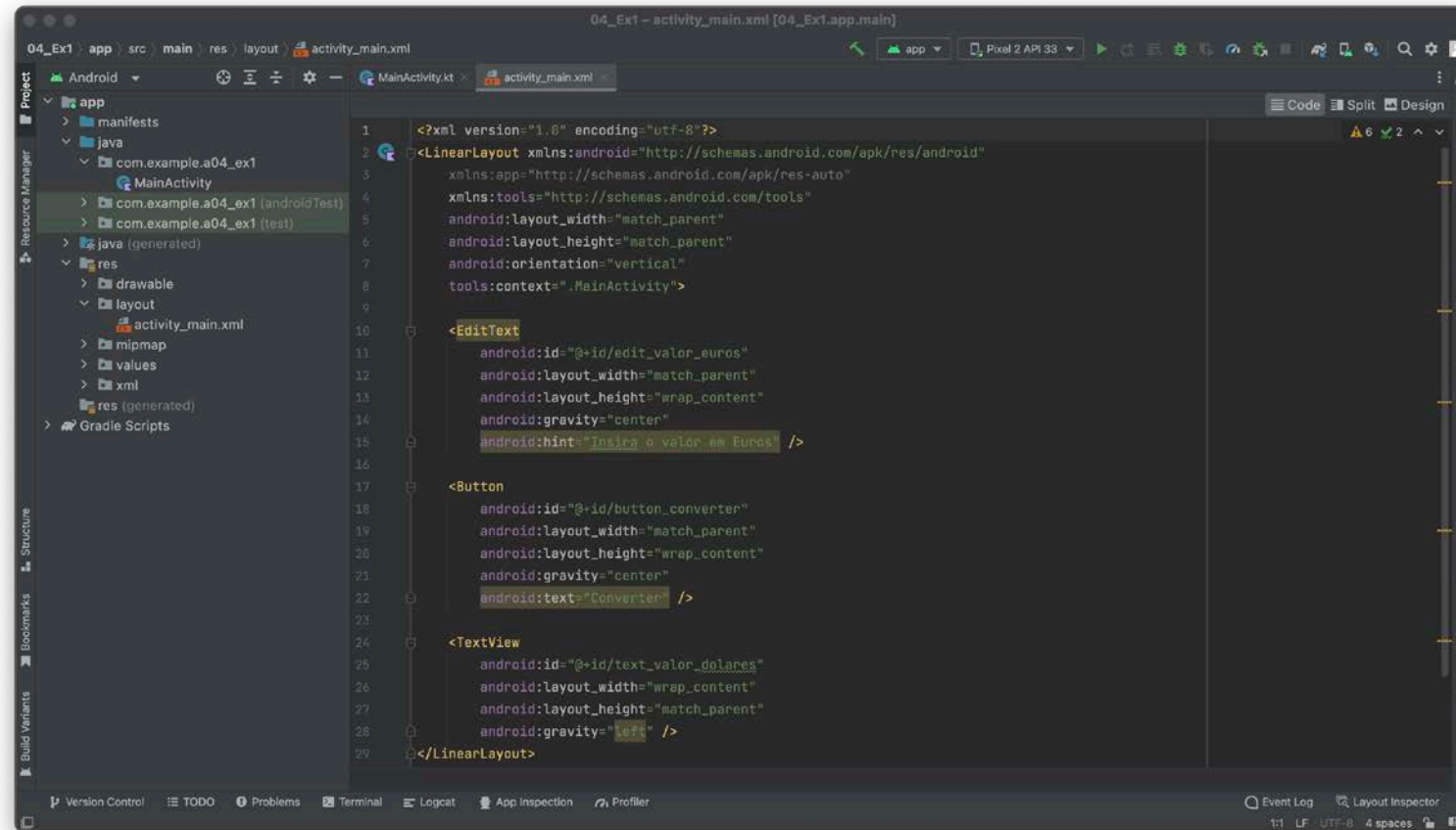


# Exercício 1

- 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



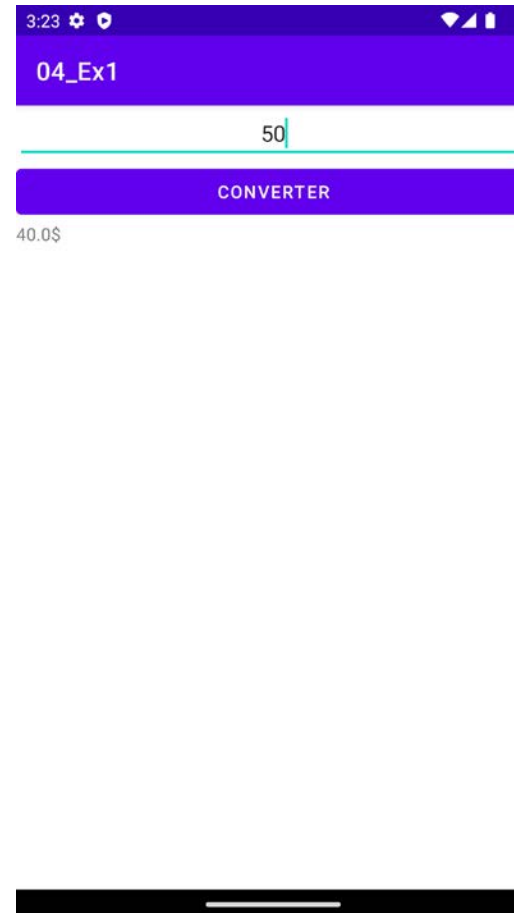
# Exercício 1



# Exercício 1

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

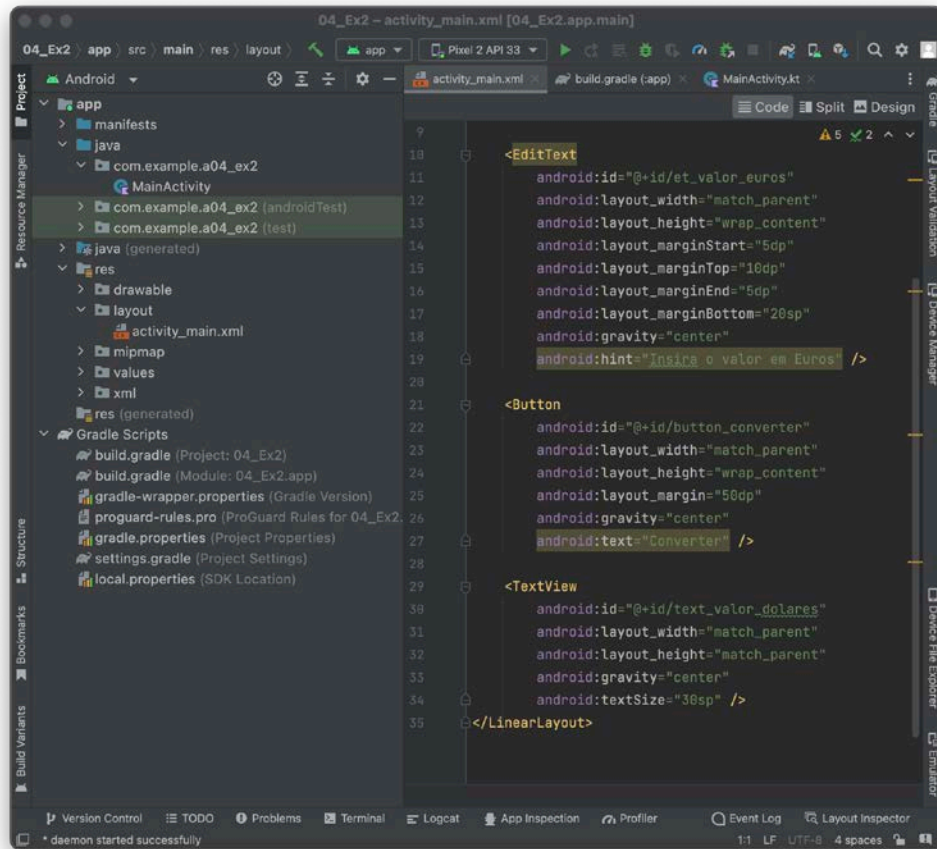
# Exercício 1



# Exercício 2

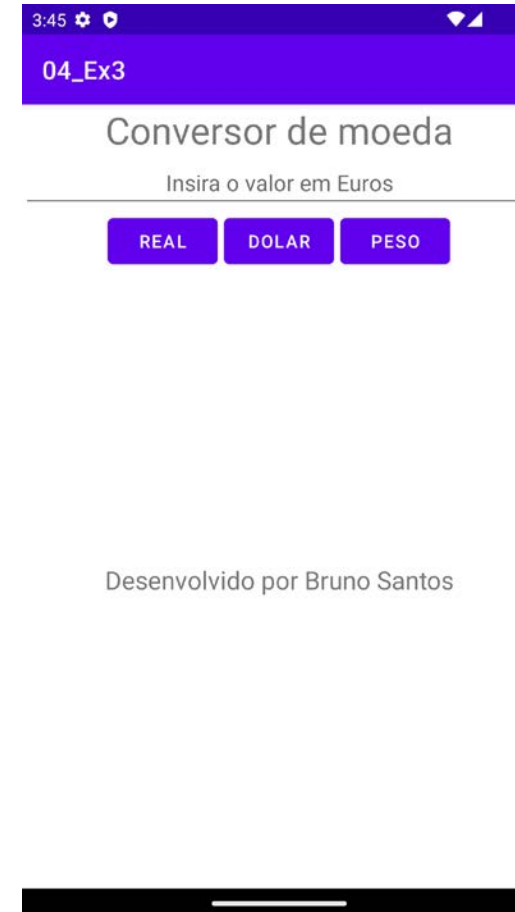
- Altere ou crie uma nova aplicação semelhante à anterior e realize as seguintes operações:
  1. Acrescente as seguintes margens ao elemento EditText:
    1. Top: 10dp
    2. Start: 5dp
    3. End: 5dp
    4. Bottom: 20dp
  2. Acrescente uma margem a toda a volta de 50dp ao elemento Button
  3. Altere 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

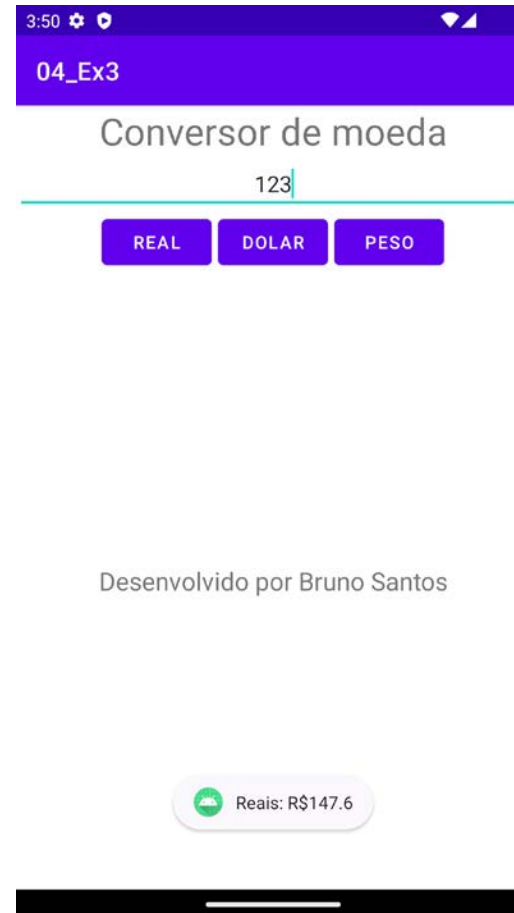
# LinearLayout

- 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



# 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

