



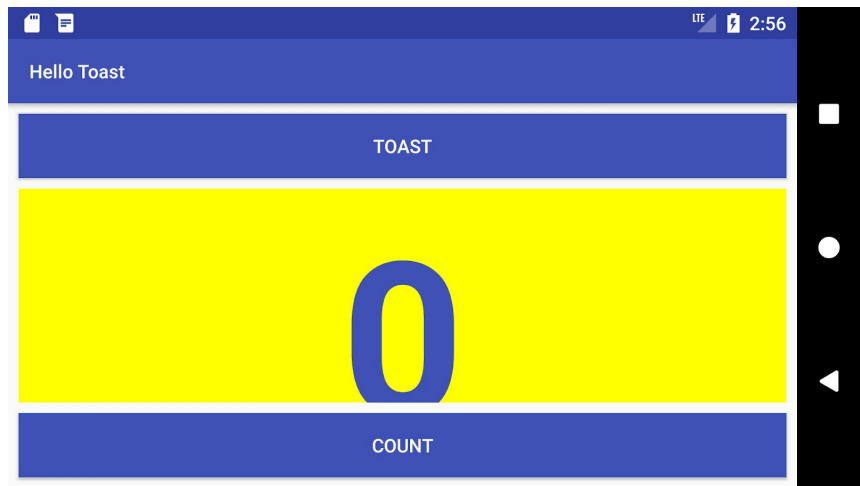
android



Coding challenge?

■ Tudo certo?

■ Como resolver?

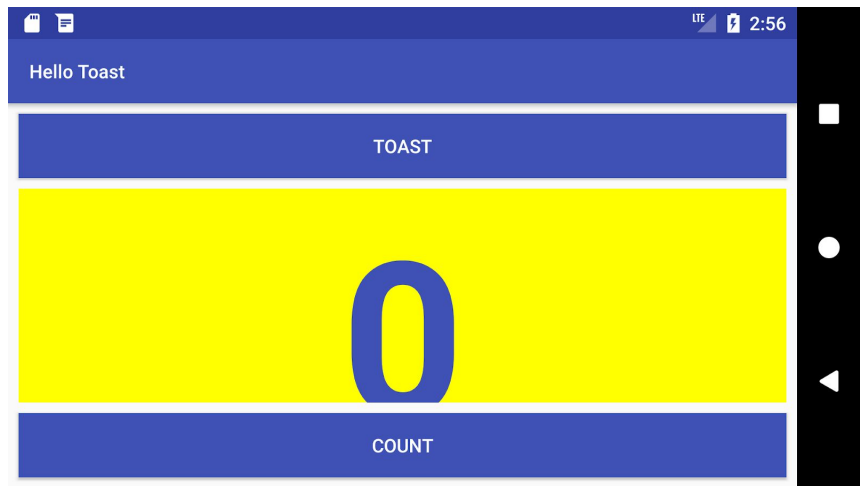


Coding challenge?

- Tudo certo?

- Como resolver?

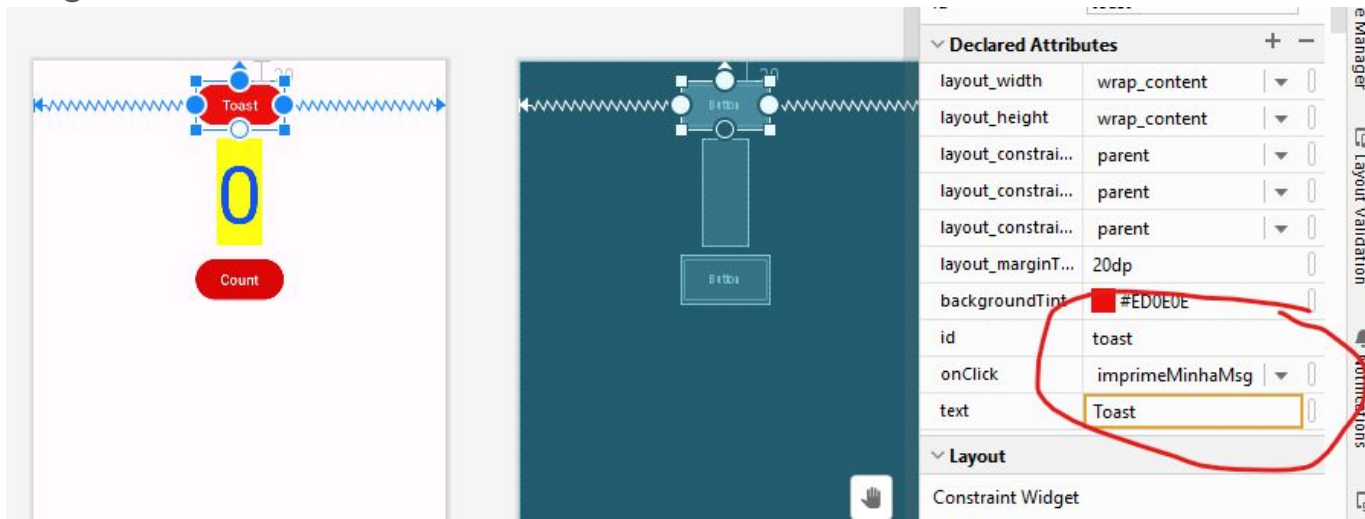
- Quantos componentes?
- Quais os constraints de layout?
- E as modificações no controller?



Parte 1 - View

- Vamos primeiro mexer nos componentes
 - Adicionem 2 buttons e um text
 - Ajustem as configurações para algo mais bonitinho

- Segredo do click -> onClick
 - Deve chamar uma função definida no controller daquela tela



Parte 2 - Controllers

- Defina as funções necessárias e variáveis de apoio para os “valores”, como o número dentro do textView, que será o contador.
 - Dica: pode ser qualquer tipo de variável, desde as simples como int até outras classes
- Como resolver?
 - Atribui o campo de texto a uma variável para manipular quando for necessário

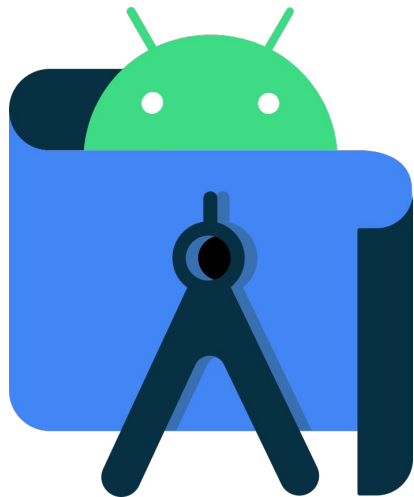
```
private TextView meuNumero;  
  
@SuppressWarnings("MissingInflatedId")  
@Override  
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
  
    meuNumero = (TextView) findViewById(R.id.meuNumero);  
    Log.i("debug", "Meu numero eh: " +  
        meuNumero.getText().toString());  
}
```

Parte 2 - Controllers

- Função de Toast no onClick no botão de Toast.
 - Dica: pode ser qualquer tipo de variável, desde as simples como int até outras classes
- Função de soma no onClick do botão de Count.

```
public void somaTexto(View view){  
    contador++;  
  
    Log.i("debug", "Novo valor do contador: "  
        + contador);  
  
    meuNumero.setText(Integer.toString(contador));  
}
```

```
public void imprimeMinhaMsg(View view){  
    Toast meuToast = null;  
    meuToast = Toast.makeText(this, "valor do contador é "  
        + contador,  
        Toast.LENGTH_LONG);  
    meuToast.show();  
}
```

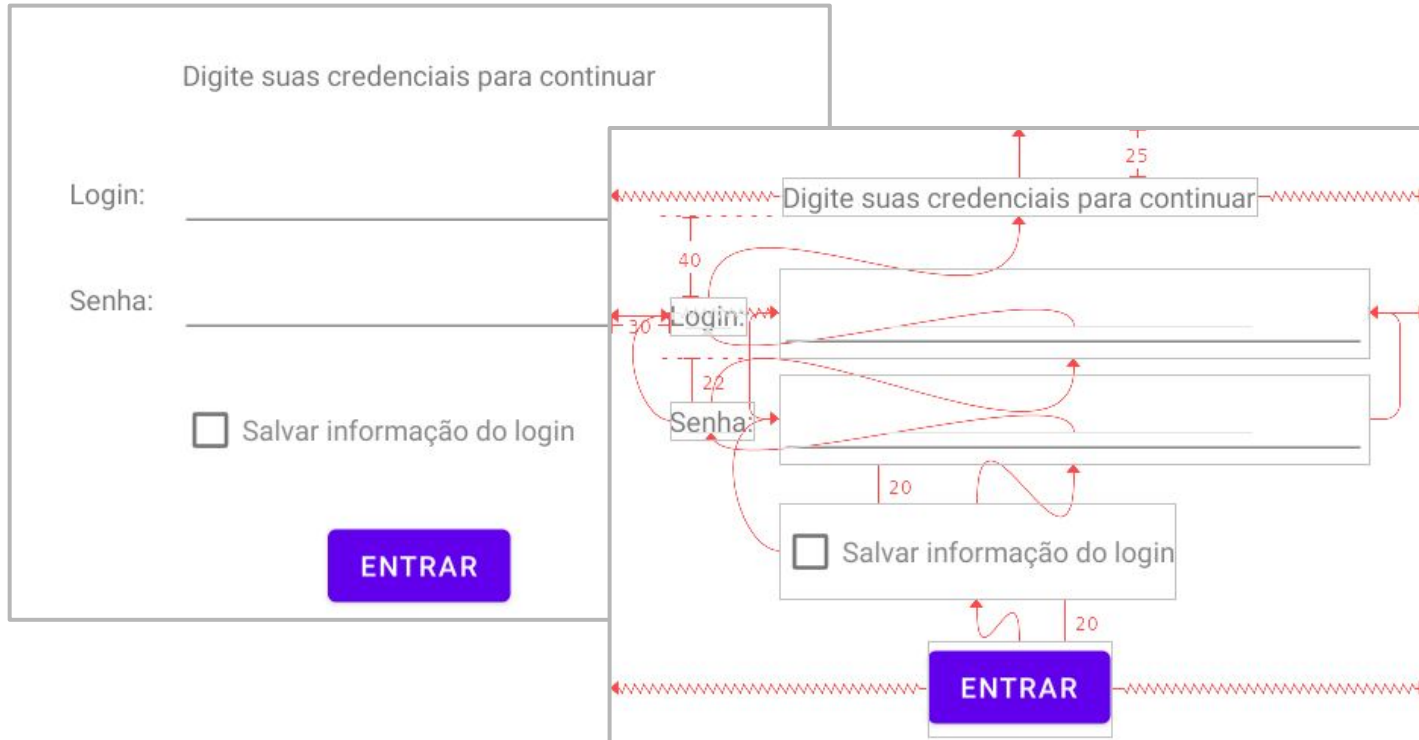


Tela de login



ConstraintLayout

- Posiciona elementos com base em outros, de forma relativa



Componentes



- Quantos componentes há na tela anterior?
 - Resposta: 1 (Constraint Layout) + 7 (EditText, TextView, Checkbox e Button)
- Tente adicionar as restrições de layout
- Teste seu app em diferentes telas e diferentes orientações para testar

TextView

- Representa um texto (label)

The image shows the Android Studio interface with several annotations and dialog boxes:

- Palette:** The 'Common' tab is selected, showing various widgets. A blue box points to the 'TextView' widget with the text: "Selecione o TextView".
- Design View:** A 'TextView' widget is being dragged onto the design surface. A blue box points to it with the text: "Arraste para a tela de design".
- Properties Panel:** On the right, the 'Declared Attributes' and 'Common Attributes' sections are visible. The 'text' attribute is highlighted with a blue box and the text: "Clique no botão para selecionar um recurso (String)".
- New String Value Dialog:** A dialog box is open with the following fields:
 - Resource name: message
 - Resource value: Digite suas credenciais para
 - Source set: main src/main/res
 - File name: strings.xml
 - Create the resource in directories: ☒ values, ☐ values-nightThe 'OK' button is highlighted with a blue box and the text: "Novo valor".
- Pick a Resource Dialog:** A dialog box is open showing the 'String' resource. The 'PlainText.app (1)' resource is selected. A blue box points to it with the text: "Escolha String value".

TextView

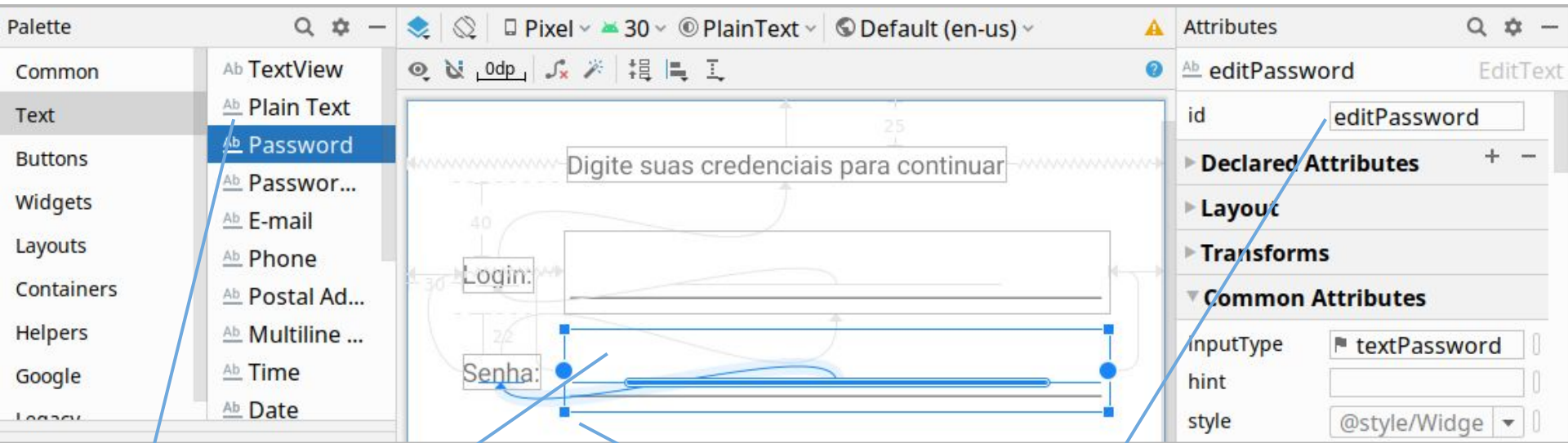
- Representa um texto (label)

- XML resultante do slide anterior (campo de senha apenas)

```
<TextView
    android:id="@+id/textMessage"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="25dp"
    android:text="@string/message"
    android:textSize="14dp"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintTop_toTopOf="parent" />
```

EditText

- Texto de entrada do usuário
- Vários tipos: `text`, `textPassword`, `number`, etc



Tipos de entrada

Arraste para a tela de design

Ajuste os alinhamentos

Diga o ID do componente. Ele será usado para acessar o valor digitado a partir do Java

EditText

- Texto de entrada do usuário
- Vários tipos: `text`, `textPassword`, `number`, etc

- XML resultante do slide anterior (campo de senha apenas)

```
<EditText
    android:id="@+id/editPassword"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:inputType="textPassword"
    app:layout_constraintStart_toStartOf="@id/editLogin"
    app:layout_constraintEnd_toEndOf="@id/editLogin"
    app:layout_constraintBaseline_toBaselineOf="@id/textPassword" />
```

CheckBox

Caixa de seleção

The screenshot shows the Android Studio IDE with a login form design. The **Palette** on the left has the **CheckBox** widget selected under the **Buttons** category. The **Component Tree** at the bottom left lists the form's components: `ConstraintLayout`, `Ab textMessage "@string/..."`, `Ab textLogin "Login:"`, `Ab editLogin (E-mail)"`, and `Ab textPassword "Senha:"`. The **Design View** in the center shows a login form with a title "Digite suas credenciais para continuar", two text input fields labeled "Login:" and "Senha:", and a checkbox labeled "Salvar informação de login". Blue arrows point from the bottom text boxes to these elements: from "Componente" to the CheckBox in the Palette; from "Arraste para a tela de design" to the "Salvar informação de login" checkbox; from "Ajuste os alinhamentos" to the alignment handles of the checkbox; from "Mude o texto" to the text "Salvar informação de login"; and from "Ajuste o ID" to the `id` attribute in the **Attributes** panel, which is currently set to `checkSaveLogin`. The **Attributes** panel also shows **Declared Attributes**, **Layout** constraints (e.g., `Start → StartOf editPassword`), and **Constraints**.

Componente

Arraste para a tela de design

Ajuste os alinhamentos

Mude o texto

Ajuste o ID

CheckBox

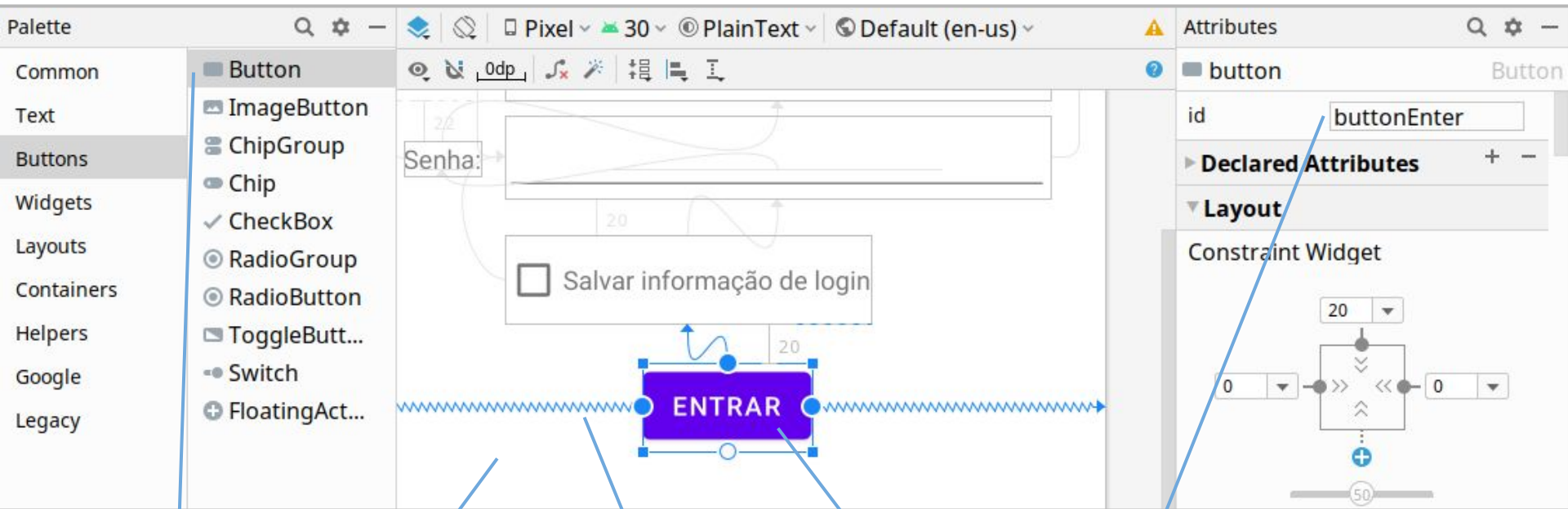
Caixa de seleção

XML resultante do slide anterior

```
<CheckBox
    android:id="@+id/checkSaveLogin"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:text="@string/saveLogin"
    android:textColor="#777777"
    app:layout_constraintStart_toStartOf="@+id/editPassword"
    app:layout_constraintTop_toBottomOf="@+id/editPassword" />
```

Button

■ Botão clicável



Componente

Arraste para a
tela de design

Ajuste os
alinhamentos

Mude o texto

Ajuste o ID

Button

- Botão clicável

- XML resultante do slide anterior

```
<Button
    android:id="@+id/buttonEnter"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:text="@string/enter"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/checkSaveLogin" />
```

Resultado Final

- Dica: experimentem diferentes parâmetros dos componentes
- Atributos dos componentes e seus significados podem ser encontrados na documentação oficial (afinal, são classes!)
- Link do TextView:
[TextView | Android Developers](#)

Digite suas credenciais

Login

Senha

Digite seu login

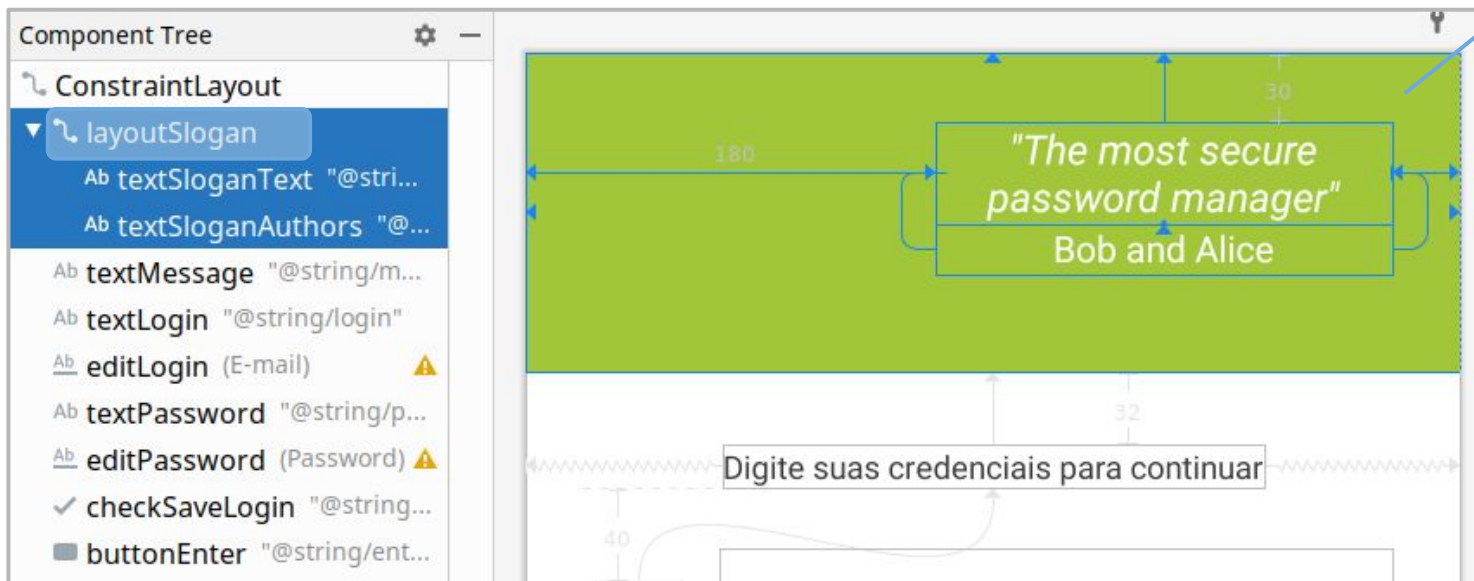
Digite sua senha

☐ Salvar informação do login

Button

Layout dentro de Layout (exemplo 1)

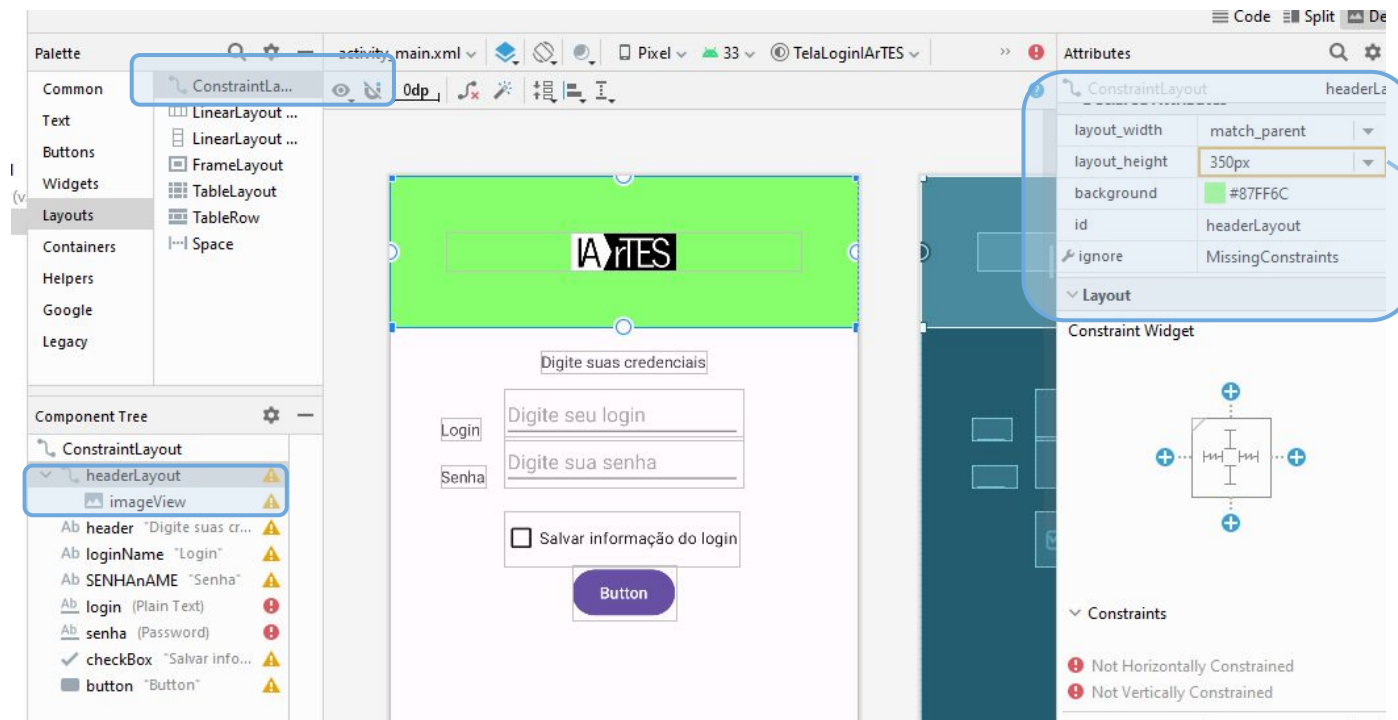
- Um layout pode ser um sub-componente de um layout
- Combine layouts para obter os efeitos desejados



layoutSlogan
é um layout com
componentes
dentro

Layout dentro de Layout (exemplo 2)

- Para fazer isto, basta arrastar um layout da paleta.



headerLayout (do tipo constraintLayout) é um layout com uma imagem.

Reparem no atributo height e width.

Além disso, o TextView que tem a mensagem Digite suas credenciais está com um constraintLayout de top para o bottom de headerLayout

Xml resultante

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout 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">

    <androidx.constraintlayout.widget.ConstraintLayout
        android:id="@+id/headerLayout"
        android:layout_width="match_parent"
        android:layout_height="350px"
        android:background="#87FF6C"
        tools:ignore="MissingConstraints">

        <ImageView
            android:id="@+id/imageView"
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:layout_marginStart="50dp"
            android:layout_marginTop="50dp"
            android:layout_marginEnd="50dp"
            android:layout_marginBottom="50dp"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:srcCompat="@drawable/img" />

    </androidx.constraintlayout.widget.ConstraintLayout>
```

Xml resultante

```
<TextView
    android:id="@+id/header"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:text="Digite suas credenciais"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/headerLayout" />
```

```
<TextView
    android:id="@+id/loginName"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="40dp"
    android:layout_marginEnd="20dp"
    android:text="Login"
    app:layout_constraintEnd_toStartOf="@+id/login"
    app:layout_constraintTop_toBottomOf="@+id/header" />
```

```
<TextView
    android:id="@+id/SENHAnAME"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="22dp"
    android:text="Senha"
    app:layout_constraintStart_toStartOf="@+id/loginName"
    app:layout_constraintTop_toBottomOf="@+id/login" />
```

Xml resultante

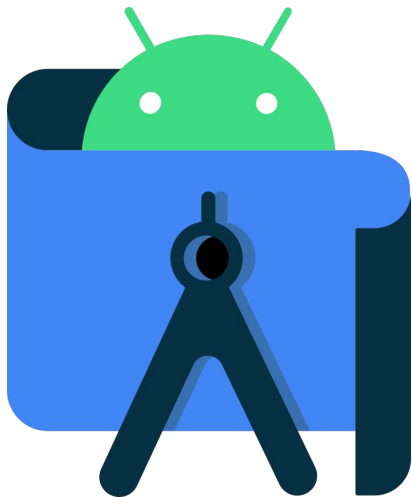
```
<EditText
    android:id="@+id/login"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="Digite seu login"
    android:inputType="text"
    app:layout_constraintBottom_toBottomOf="@+id/loginName"
    tools:layout_editor_absoluteX="100dp" />
```

```
<EditText
    android:id="@+id/senha"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:ems="10"
    android:hint="Digite sua senha"
    android:inputType="textPassword"
    app:layout_constraintBottom_toBottomOf="@+id/SENHAnAME"
    app:layout_constraintEnd_toEndOf="@+id/login"
    app:layout_constraintStart_toStartOf="@+id/login" />
```

```
<CheckBox
    android:id="@+id/checkBox"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginTop="20dp"
    android:text="Salvar informação do login"
    app:layout_constraintStart_toStartOf="@+id/senha"
    app:layout_constraintTop_toBottomOf="@+id/senha" />
```

Xml resultante

```
<Button
    android:id="@+id/button"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Button"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/checkbox" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Challenging

Github: github.com/diogosm/android_iartes_2025

Prof. Diogo Soares



diogosoaresm@ufam.edu.br

Tela de login



- Tentem pegar os valores de login e senha e atribuir para variáveis no controller
 - Desafio 2: imprima um Toast com “Olá, nomeColocadoNoLogin”