

INSTITUTO FEDERAL DE
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
MARANHÃO

Prof. MSc. Flávio Barros

flavioifma@gmail.com

www.flaviobarros.com.br

Programação para Dispositivos Móveis

Aula - Layout (parte I)

Roteiro

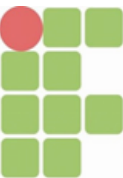
Layout



Layout

Fundamentos

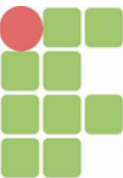
- O layout define a estrutura visual para uma interface do usuário, como a IU de uma atividade ou de um widget de aplicativo;
- É possível declarar um layout de dois modos:
 - **Declarar elementos da IU em XML**; O Android fornece um vocabulário XML direto que corresponde às classes e subclasses de View, como as de widgets e layouts;
 - **Instanciar elementos do layout em tempo de execução**; O aplicativo pode criar objetos View e ViewGroup (e processar suas propriedades) programaticamente.



Layout

Vantagens

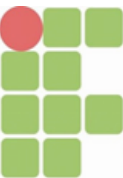
- A vantagem de declarar a IU em XML é separar melhor a apresentação do aplicativo do código que controla seu comportamento;
- As descrições da IU são externas ao código do aplicativo, ou seja, é possível modificá-las ou adaptá-las sem modificar o código-fonte e recompilar;
- Ex. É possível criar layouts XML para diferentes orientações de tela, diferentes tamanhos de tela de dispositivos e diferentes idiomas.



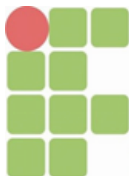
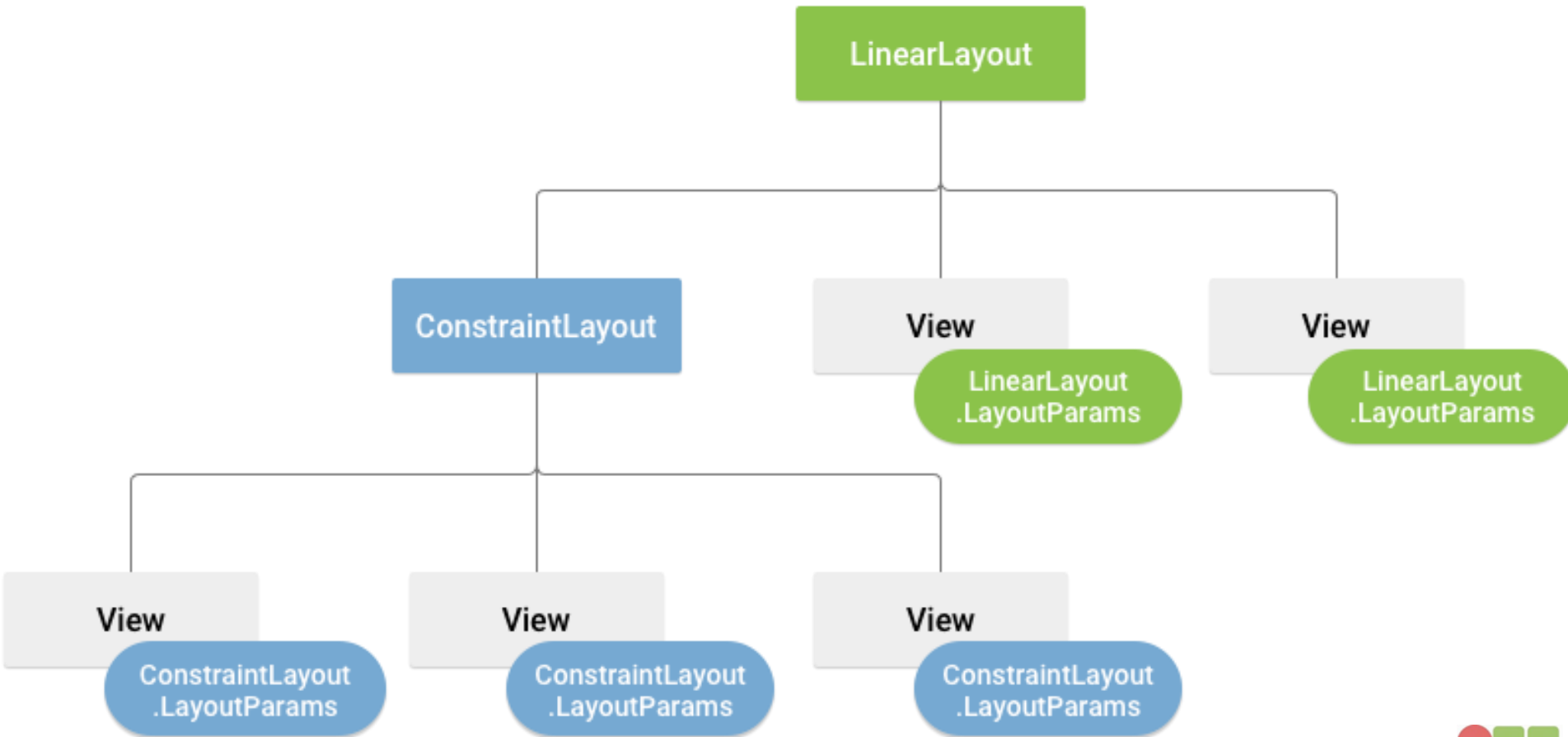
Layout

Parâmetros

- Os atributos do layout XML chamados `layout_something` definem parâmetros para a `View` apropriados para a `ViewGroup` em que reside;
- Cada classe `ViewGroup` implementa uma classe aninhada que estende `ViewGroup.LayoutParams`;
- Um grupo de exibições pais define parâmetros de layout para cada exibição filha (incluindo o grupo de exibições filhas).



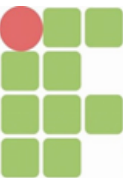
Parâmetros do layout



Layout

Parâmetros

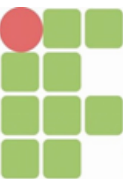
- Todos os grupos de exibições contêm largura e altura (`layout_width` e `layout_height`) e cada exibição é obrigatória para defini-las;
- Muitos `LayoutParams` também contêm margens e bordas opcionais;
- É possível especificar largura e altura com medidas exatas, embora não seja recomendável na maioria dos casos;



Layout

Parâmetros

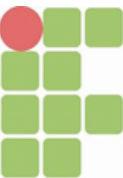
- Em geral, usa-se uma destas constantes para definir a **largura** e a **altura**:
 - **wrap_content** instrui a exibição a se redimensionar de acordo com as medidas exigidas pelo conteúdo;
 - **match_parent** instrui a exibição a assumir o maior tamanho permitido pelo grupo de exibições pais;



Posição do Layout

Fundamentos

- A geometria de uma exibição de um retângulo;
- As exibições têm uma localização, expressa como um par de coordenadas esquerda e topo e duas dimensões, expressas como largura e altura;
- A unidade de localização e de dimensões é o pixel;
- É possível recuperar a localização de uma exibição chamando os métodos:
 - `getLeft()`, `getTop()`, `getRight()` e `getBottom()`;



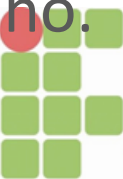
Roteiro

Layout linear



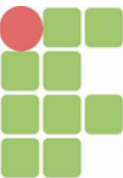
Layout linear

- É um grupo de exibições que **alinha todos os filhos** em uma **única direção vertical** ou **horizontal**; Você pode especificar a direção do layout com o atributo `android:orientation`;
- Todos os filhos de um **LinearLayout** são empilhados um após o outro;
- Uma lista vertical terá apenas um filho por linha, independentemente de sua largura, e uma lista horizontal terá altura de apenas uma linha (a altura do filho mais alto, mais preenchimento);
- Um **LinearLayout** respeita margens entre filhos e a gravidade (alinhamento à direita, no centro ou à esquerda) de cada filho.



Layout linear

```
<LinearLayout  
  xmlns:android="http://schemas.android.com/apk/res/android"  
  
  android:layout_width="match_parent"  
  android:layout_height="match_parent"  
  android:paddingLeft="20dp"  
  android:paddingRight="20dp"  
  android:orientation="vertical" >  
  ...  
</LinearLayout>
```



Roteiro

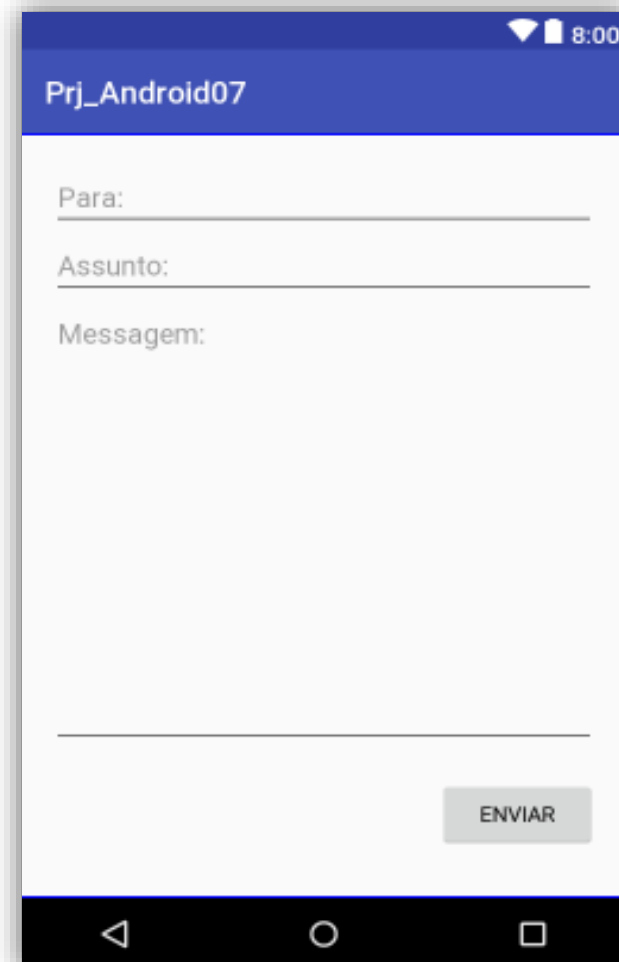


Prj_Android07

Prj_Android07

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="20dp"
    android:paddingRight="20dp"
    android:orientation="vertical" >
    <EditText
        android:id="@+id/edt_para"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="20dp"
        android:hint="Para: " />
    <EditText
        android:id="@+id/edt_assunto"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Assunto: " />
```

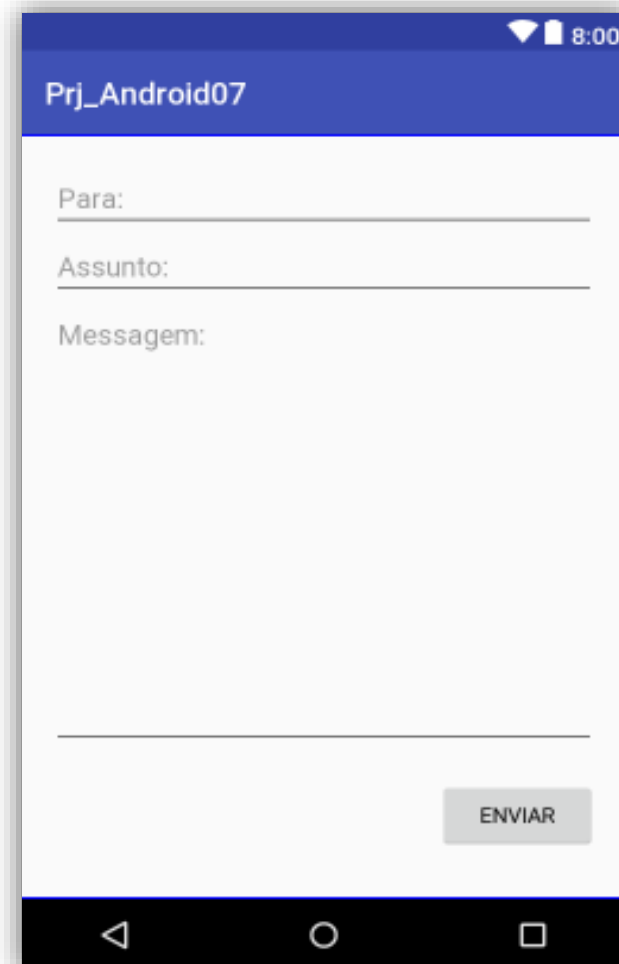
activity_main.xml



Prj_Android07

```
<EditText
    android:id="@+id/edt_msg"
    android:layout_width="match_parent"
    android:layout_height="300dp"
    android:gravity="top"
    android:hint="Mensagem: " />
<Button
    android:id="@+id/btn_enviar"
    android:layout_width="100dp"
    android:layout_height="wrap_content"
    android:layout_gravity="right"
    android:layout_marginTop="20dp"
    android:text="Enviar" />
</LinearLayout>
```

activity_main.xml



Roteiro

Table Layout

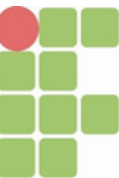


Table Layout

- É uma especialização do Linear Layout, muito utilizado para criar formulários; Seu layout que organiza seus filhos em linhas e colunas;
- Consiste em um número de objetos `TableRow`, cada um definindo uma linha;
- `TableLayout` não exibem linhas de borda para suas linhas, colunas ou células;
- Cada linha tem zero ou mais células; Cada célula pode conter um objeto View.

<https://developer.android.com/reference/android/widget/TableLayout.html>



Roteiro



Prj_Android08

Prj_Android08

```
<TableLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:shrinkColumns="*"
    android:background="#ffffff">
    <!-- Linha 1 com uma coluna -->
    <TableRow
        android:layout_height="wrap_content"
        android:layout_width="fill_parent"
        android:gravity="center_horizontal">
        <TextView
            android:id="@+id/TextView00"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textSize="20dp"
            android:text="Linha 1"
            android:layout_span="3"
            android:padding="18dip"
            android:background="#FF0000"
            android:textColor="#000"/>
        </TableRow>
```



Prj_Android08

```

<!-- Linha 2 com 2 colunas -->
<TableRow>
    android:layout_height="wrap_content"
    android:layout_width="fill_parent"
    android:gravity="center_horizontal">

    <TextView
        android:id="@+id/TextView01"
        android:text="Linha 2 Coluna 1"
        android:layout_weight="1"
        android:background="#00AF00"
        android:textColor="#000000"
        android:padding="18dip"
        android:gravity="center"/>

    <TextView
        android:id="@+id/TextView02"
        android:text="Linha 2 Coluna 2"
        android:layout_weight="1"
        android:background="#00FF00"
        android:textColor="#000000"
        android:padding="18dip"
        android:gravity="center"/>
</TableRow>

```



Prj_Android08

```

<TableRow
    android:id="@+id/tableRow1"
    android:layout_height="wrap_content"
    android:layout_width="match_parent">
    <TextView
        android:id="@+id/TextView03"
        android:text="Linha 3 Coluna 1"
        android:layout_weight="1"
        android:background="#0000FF"
        android:textColor="#000000"
        android:padding="20dip"
        android:gravity="center"/>
    <TextView
        android:id="@+id/TextView04"
        android:text="Linha 3 Coluna 2"
        android:layout_weight="1"
        android:background="#0000CF"
        android:textColor="#000000"
        android:padding="20dip"
        android:gravity="center"/>
    <TextView
        android:id="@+id/TextView05"
        android:text="Linha 3 Coluna 3"
        android:layout_weight="1"
        android:background="#0000AF"
        android:textColor="#000000"
        android:padding="20dip"
        android:gravity="center"/>
</TableRow>

```



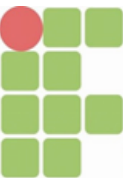
Roteiro

Relative Layout



Relative Layout

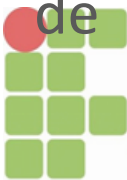
- É um grupo de visualizações que exhibe seus filhos em posições relativas;
- Permite especificar a localização de objetos filhos relativos entre si (filho A à esquerda do filho B) ou relativos aos pais (alinhados no topo do pai);
- `RelativeLayout` é um utilitário muito poderoso para projetar uma interface de usuário porque pode eliminar grupos de exibição aninhados e manter a hierarquia de layout plana, o que melhora o desempenho.



Relative Layout

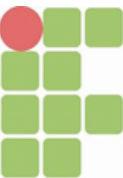
Visualizações de Posicionamento

- Algumas propriedades de layout disponíveis para visualizações em um `RelativeLayout` incluem:
 - `android:layout_alignParentTop` (Se "**true**", faz com que a borda superior dessa exibição corresponda à borda superior do pai);
 - `android:layout_centerVertical` (Se "**true**", centraliza esse filho verticalmente dentro de seu pai);
 - `android:layout_below` (Posiciona a borda superior dessa vista abaixo da visualização especificada com um ID de recurso);
 - `android:layout_toRightOf` (Posiciona a borda esquerda desta exibição à direita da visualização especificada com um ID de recurso).



Relative Layout

```
<RelativeLayout  
  xmlns:android="http://schemas.android.com/apk/res/android"  
  
  android:layout_width="match_parent"  
  android:layout_height="match_parent"  
  android:paddingLeft="20dp"  
  android:paddingRight="20dp" >  
  ...  
</RelativeLayout>
```



Roteiro

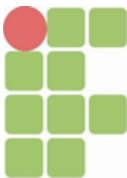


Praticando...

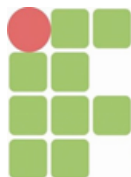
Praticando

Prj_Android_Extra06

- Construir um aplicativo com 3 notícias, sobre tecnologia.
 - Imagem no topo da página principal (*logo do portal*);
 - Imagem no topo de cada página, referente as notícias;
 - Botão “ANTERIOR” e “PRÓXIMA” em cada página de notícia, e botão “HOME” para a página principal;
 - Botão SAIR em todas páginas.

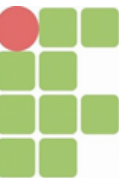


Praticando



Roteiro

Referências



Referências

- DEITEL, P.; DEITEL, H.; DEITEL, A. **Android 6 - Para Programadores - Uma Abordagem Baseada em Aplicativos**. 2.ed. Bookman, 2015.
- Android Studio “**Layout**”. Disponível em:
<<https://developer.android.com/guide/topics/ui/declaring-layout.html?hl=pt-br>>.

