

Roteiro de Laboratório 06

José Cassiano Grassi Gunji

Criando Menu

Neste roteiro vamos criar um menu que pode ser usado nas *activities* do seu aplicativo.

Abra o Android Studio, crie um novo projeto usando o modelo *Empty Activity*. Nomeie seu projeto como “Exemplo Menu”. Defina o pacote como preferir, escolha a linguagem como Java e escolha o nível mínimo de API conforme sua preferência.

Quando o projeto estiver carregado, clique com o botão direito do mouse sobre a pasta **res** de seu projeto e escolha [New]->[Android Resource File], conforme a Figura 1.

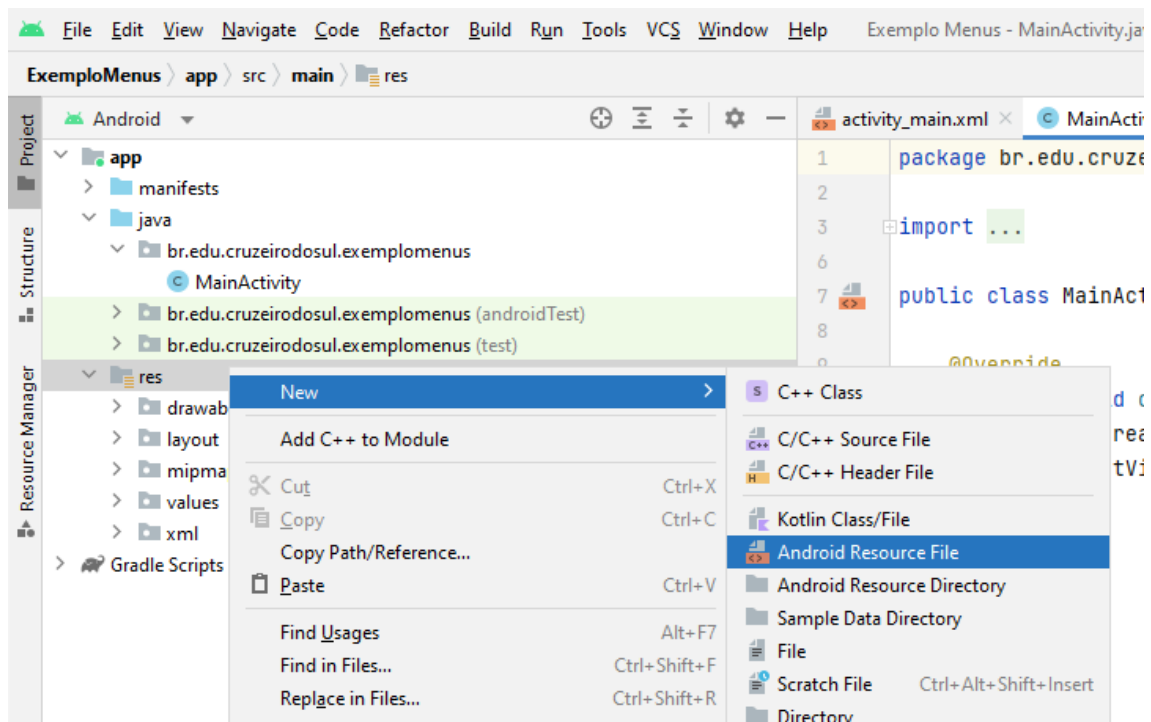


Figura 1: Criando um novo recurso.

A seguir, nomeie o arquivo de seu recurso como **menu_principal** e defina o tipo do recurso como **Menu** conforme a Figura 2.

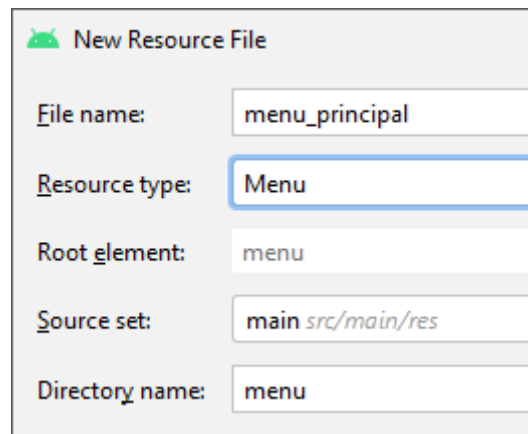


Figura 2: Definindo o nome e o tipo do recurso.

Neste momento, o Android Studio deve apresentar um editor de menus. Seu menu pode ser editado usando este editor, alterando o código XML ou mesmo por programação. Neste exemplo, vamos usar o editor gráfico.

Arraste para a visualização do menu um novo **Menu item**. Vide a Figura 3.

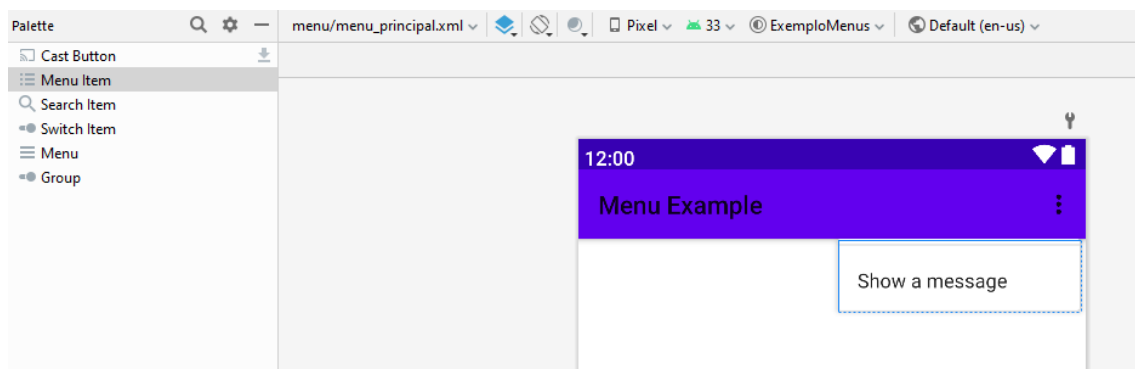


Figura 3: Inserindo um item de menu.

Defina o **Id** deste item de menu para **menuItemMensagem**. Crie um novo recurso de string chamado **exibirMensagem** com o conteúdo **"Show a message"**.

Aproveite este momento e crie os demais recursos de string apresentados na Figura 4 usando seu método preferido. Não se esqueça de definir o seu próprio nome como o autor do aplicativo!

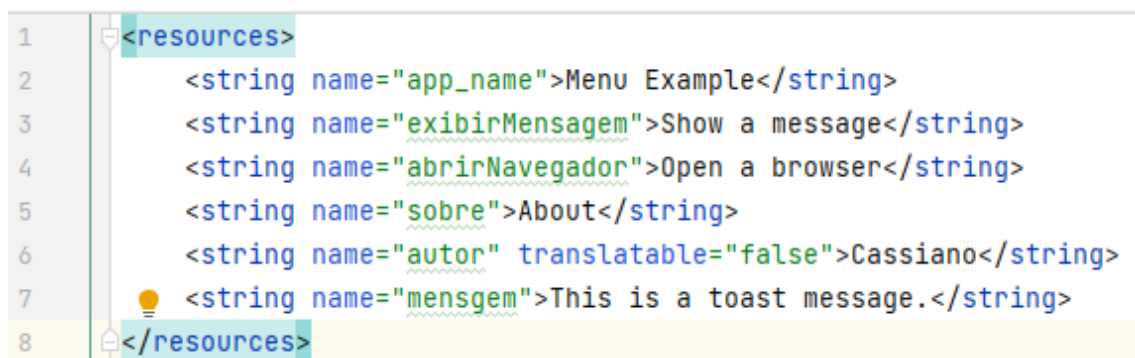


Figura 4: Criando os recursos de texto.

Abra o editor de traduções, crie um novo local para o idioma português brasileiro e faça as traduções dos textos. Marque o nome do autor (seu nome!) como não traduzível. Veja a Figura 5.

Key	Resource Folder	Untranslatable	Default Value	Portuguese (pt) in Brazil (BR)
app_name	app/src/main/res	<input type="checkbox"/>	Menu Example	Exemplo Menus
exibirMensagem	app/src/main/res	<input type="checkbox"/>	Show a message	Exibe uma mensagem
abrirNavegador	app/src/main/res	<input type="checkbox"/>	Open a browser	Abre um navegador
sobre	app/src/main/res	<input type="checkbox"/>	About	Sobre
autor	app/src/main/res	<input checked="" type="checkbox"/>	Cassiano	
mensagem	app/src/main/res	<input type="checkbox"/>	This is a toast mes	Esta é uma mensagem toast

Figura 5: Traduzindo o aplicativo.

Crie dois novos itens de menu. Defina seus **Ids** como **menuItemNavegador** e **menuItemSobre**. A seguir, altere seus títulos com os recursos de texto apropriados.

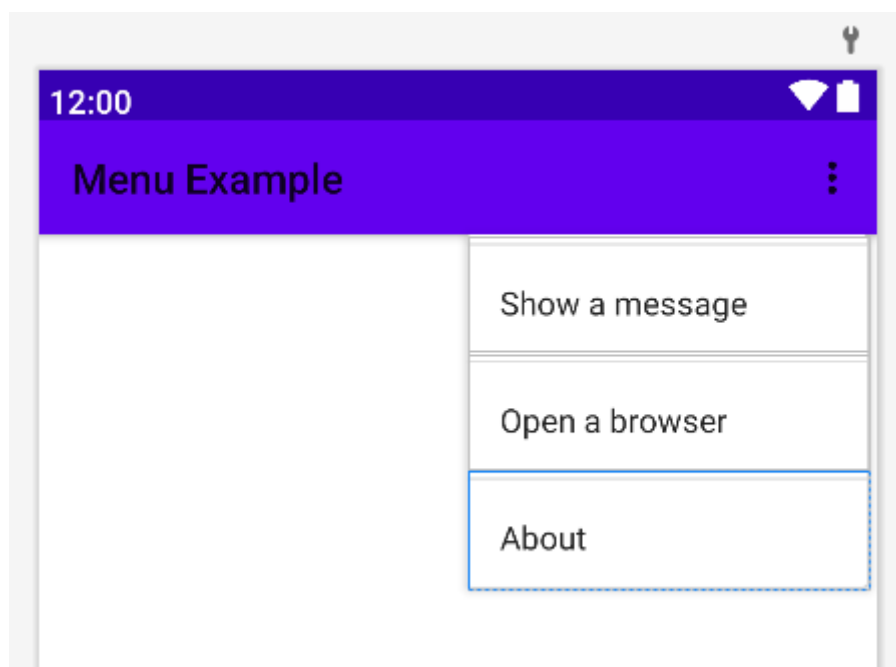


Figura 6: Menu completo.

Compare o código XML de seu menu com o da Figura 7.

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <menu xmlns:app="http://schemas.android.com/apk/res-auto"
3      xmlns:android="http://schemas.android.com/apk/res/android">
4
5      <item
6          android:id="@+id/menuItemMensagem"
7          android:title="@string/exibirMensagem" />
8
9      <item
10         android:id="@+id/menuItemNavegador"
11         android:title="@string/abrirNavegador" />
12
13     <item
14         android:id="@+id/menuItemSobre"
15         android:title="@string/sobre" />
16
17 </menu>

```

Figura 7: Código XML do menu.

Na **activity_main.xml**, remova o **TextView** que apresenta o texto “Hello world!”.

Neste momento podemos inflar o menu em nossa atividade principal. Inflar é o processo em que o Sistema Operacional transforma a especificação em XML de um layout em um componente gráfico em execução.

Abra a sua classe **MainActivity.java**, sobrescreva o método **onCreateOptionsMenu()** conforme a Figura 8. Dica: Se você digitar “@Override onCreate...”, o Android Studio vai autocompletar a sobrescrita inteira do método para você.

```

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.menu_principal, menu);
    return true;
}

```

Figura 8: Sobrescrita do método **onCreateOptionsMenu()**.

Antes de prosseguir com a codificação do menu, vamos criar uma nova atividade para servir como a tela de créditos do aplicativo. Crie uma nova atividade com o modelo **Empty Activity** chamada **SobreActivity**. No seu arquivo de layout **activity_sobre.xml**, coloque dois **TextViews** e defina o texto deles para os recursos de texto **@string/app_name** e **@string/autor**. Vide a Figura 9.

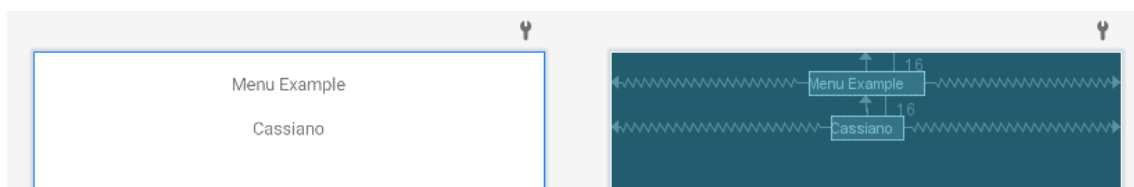


Figura 9: Layout da atividade **SobreActivity**.

Para tratar os eventos de clique nos menus, vamos sobrescrever o método **onOptionsItemSelected()** usando o mesmo truque apresentado acima. A seguir, altere seu

código conforme a Figura 10 e crie também os métodos **exibaMensagem()**, **exibaNavegador()** e **exibaSobre()**.

```
29      @Override
30      public boolean onOptionsItemSelected(@NonNull MenuItem item) {
31          boolean saida = true;
32          switch (item.getItemId()){
33              case R.id.menuItemMensagem:
34                  exibaMensagem();
35                  break;
36              case R.id.menuItemNavegador:
37                  exibaNavegador();
38                  break;
39              case R.id.menuItemSobre:
40                  exibaSobre();
41                  break;
42              default:
43                  saida = super.onOptionsItemSelected(item);
44          }
45          return saida;
46      }
47
48      private void exibaMensagem(){
49          Toast.makeText(context, this, "This is a toast message.", Toast.LENGTH_SHORT).show();
50      }
51
52      private void exibaNavegador(){
53          Uri uri = Uri.parse("https://www.cruzeirodosul.edu.br");
54          Intent intent = new Intent(Intent.ACTION_VIEW, uri);
55          startActivity(intent);
56      }
57
58      private void exibaSobre(){
59          Intent intent = new Intent(packageContext, this, SobreActivity.class);
60          startActivity(intent);
61      }
```

Figura 10: Sobrescrita do método `onOptionsItemSelected()`.

Para completar, defina no arquivo **AndroidManifest.xml** que a atividade pai de **SobreActivity** é a **MainActivity**. Defina também que o modo de lançamento da **MainActivity** é **singleTop**.

```
<activity
    android:name=".SobreActivity"
    android:parentActivityName=".MainActivity"
    android:exported="false">
    <meta-data
        android:name="android.app.lib_name"
        android:value="" />
</activity>
<activity
    android:name=".MainActivity"
    android:launchMode="singleTop"
    android:exported="true">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />

        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>

    <meta-data
        android:name="android.app.lib_name"
        android:value="" />
</activity>
```

Figura 11: Configuração das activities no arquivo AndroidManifest.xml.

Referências

<https://developer.android.com/develop/ui/views/components/menus>

DEITEL, Paul et al. Android para Programadores: uma abordagem baseada em aplicativos, 1ª Edição, 2013.