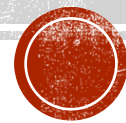


ANDROID

Menus



Menus

- Link: <https://developer.android.com/guide/topics/ui/menus.html?hl=pt-br>
- Dois tipos:
 - **Menu de Opções:** apresentam opções relevantes para a Activity atual ou para a aplicação
 - **Menu de Contexto:** apresentam opções relevantes para o elemento que está sendo apresentado



Menu de Opções

- Menus são definidos em arquivos XML separados
- Devem ficar no diretório **res/menu**
 - O nome do arquivo é usado para identificar o menu, exemplo: `meu_menu.xml`
- O menu é criado na própria Activity chamando: **onCreateOptionsMenu**
 - Aqui deve-se usar um **MenuInflater** para carregar o XML do menu para ser apresentado
- Quando um item do menu é pressionado
 - Chamado o método: **onOptionsItemSelected**



Menu de Opções: meu_menu.xml

- Itens de Menu são indicados dentro do arquivo XML

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">

    <item
        android:id="@+id/sair"
        android:icon="@android:drawable/ic_menu_close_clear_cancel"
        android:title="@string/sair"/>

</menu>
```



Menu de Opções: meu_menu.xml

- Itens de Menu são indicados dentro do arquivo XML

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">

    <item
        android:id="@+id/sair"
        android:icon="@android:drawable/ic_menu_close_clear_cancel"
        android:title="@string/sair"/>

</menu>
```



Menu de Opções: onCreateOptionsMenu

- Na Activity, implementa-se o método: `onCreateOptionsMenu`
- MenuInflater**: carrega a definição do menu que está no arquivo XML para dentro do objeto Menu recebido como parâmetro

```
public boolean onCreateOptionsMenu(Menu menu) {
    MenuInflater inflater = getMenuInflater();
    inflater.inflate(R.menu.meu_menu, menu);
    return true;
}
```



Menu de Opções: *listener*

- Na Activity, implementa-se o método: `onMenuItemSelected`
- Ou `onOptionsItemSelected`, quando usando `AppCompatActivity`
- Como só tem a opção SAIR, não é necessário qualquer teste, basta finalizar a activity (`finish()`)

```
@Override
public boolean onMenuItemSelected(int featureId, MenuItem item) {
    finish();
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem menu) {
    if (menu.getItemId() == R.id.item_menu) {
        finish();
        return true;
    }
    else {
        super.onOptionsItemSelected(menu);
    }
}
```



Menu com Mais Opções

- Itens de Menu são indicados dentro do arquivo XML

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">

    <item
        android:id="@+id/novo"
        android:icon="@android:drawable/ic_menu_add"
        android:title="@string/novo"/>

    <item
        android:id="@+id/sair"
        android:icon="@android:drawable/ic_menu_close_clear_cancel"
        android:title="@string/sair"/>

</menu>
```



Menu com Mais Opções

- `onCreateOptionsMenu` é o mesmo
- `onOptionsItemSelected` deve testar a opção selecionada

```
@Override
public boolean onOptionsItemSelected(MenuItem item) {
    switch (item.getItemId()) {
        case R.id.novo:
            startActivity(new Intent(this, SegundaActivity.class));
            return true;
        case R.id.sair:
            finish();
            return true;
        default:
            return super.onOptionsItemSelected(item);
    }
}
```



Grupos de Menus / Submenus

- Grupos de Menus
 - Um container invisível para agrupar elementos ITEM.
 - Bom para esconder vários itens ao mesmo tempo, por exemplo
- Submenus
 - Adicionando um MENU dentro de um elemento ITEM
 - Organização de opções



Grupo de Menus / Submenus

▪ Exemplo de Grupo de Menus

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:id="@+id/menu_save"
          android:icon="@drawable/menu_save"
          android:title="@string/menu_save" />

    <!-- grupo de menus -->
    <group android:id="@+id/group_delete">
        <item android:id="@+id/menu_archive"
              android:title="@string/menu_archive" />
        <item android:id="@+id/menu_delete"
              android:title="@string/menu_delete" />
    </group>
</menu>
```



Grupo de Menus / Submenus

▪ Exemplo de Submenus

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:id="@+id/file"
          android:title="@string/file" >
        <!-- submenu da opção FILE -->
        <menu>
            <item android:id="@+id/create_new"
                  android:title="@string/create_new" />
            <item android:id="@+id/open"
                  android:title="@string/open" />
        </menu>
    </item>
</menu>
```



Menus de Contexto

- Menu específico que se abre para um elemento selecionado
- Ex.: Segurando pressionado sobre uma mensagem aparece um menu que permite o reenvio, em caso de falha
- Necessário:
 - Ter um XML com as opções do menu
 - Para criar o menu, sobrescrever o método `onCreateContextMenu`
 - Para tratar a opção pressionada, criar o método `onContextItemSelected`
 - Registrar a View para ter um menu de contexto, usando `registerForContextMenu (View view)`
- Para abrir o Menu de Contexto: Selecionar o elemento e manter pressionado



Exemplo: cmenu.xml

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item
        android:id="@+id/adicionar"
        android:icon="@android:drawable/ic_menu_add"
        android:title="@string/adicionar"/>
    <item
        android:id="@+id/remove"
        android:icon="@android:drawable/ic_menu_delete"
        android:title="@string/remove"/>
</menu>
```



Exemplo: Activity

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    registerForContextMenu((TextView) findViewById(R.id.texto));
}

@Override
public void onCreateContextMenu(ContextMenu menu, View v,
                               ContextMenu.ContextMenuInfo menuInfo) {
    super.onCreateContextMenu(menu, v, menuInfo);
    MenuInflater inflater = getMenuInflater();
    inflater.inflate(R.menu.cmenu, menu);
}
```



Exemplo: Activity

```
@Override
public boolean onContextItemSelected(MenuItem item) {
    AdapterView.AdapterContextMenuInfo info =
        (AdapterView.AdapterContextMenuInfo) item.getContextMenuInfo();

    if (item.getItemId() == R.id.remover) {
        Toast.makeText(this, "ADICIONAR", Toast.LENGTH_SHORT).show();
        return true;
    }
    if (item.getItemId() == R.id.adicionar) {
        Toast.makeText(this, "REMOVER", Toast.LENGTH_SHORT).show();
        return true;
    }
    return super.onContextItemSelected(item);
}
```



Menus de Contexto - ListView

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    String[] de = { "data", "descricao" };
    int[] para = { R.id.data, R.id.descricao };
    SimpleAdapter adapter = new SimpleAdapter(this, listarDados(),
                                             R.layout.listagem, de, para);

    setListAdapter(adapter);
    getListView().setOnItemClickListener(this);
    // Aqui registra-se o menu de contexto
    registerForContextMenu(getListView());
}
```



Menus de Contexto: cmenu.xml

- XML com os itens do menu

```
<?xml version="1.0" encoding="utf-8"?>
<menu xmlns:android="http://schemas.android.com/apk/res/android" >
    <item
        android:id="@+id/remover"
        android:icon="@android:drawable/ic_menu_delete"
        android:title="@string/remover"/>
</menu>
```



Menus de Contexto: onCreateContextMenu

- Para criar o menu, na Activity implementa-se `onCreateContextMenu()`

```
@Override
public void onCreateContextMenu(ContextMenu menu, View v,
                               ContextMenuInfo menuInfo) {

    super.onCreateContextMenu(menu, v, menuInfo);

    MenuInflater inflater = getMenuInflater();
    inflater.inflate(R.menu.cmenu, menu);
}
```



Menus de Contexto: onContextItemSelected

- Para tratar um item de menu de contexto pressionado, na Activity implementa-se `onContextItemSelected()`

```
@Override
public boolean onContextItemSelected(MenuItem item) {

    AdapterContextMenuInfo info = (AdapterContextMenuInfo) item.getMenuInfo();
    if (item.getItemId() == R.id.remover) {
        lista.remove(info.position);
        getListView().invalidateViews();
        dataAnterior = "";
        // Fazer o que tem que ser feito, ex: remover do banco de dados
        return true;
    }
    return super.onContextItemSelected(item);
}
```

