

Kliensalkalmazások

Android 3 – View, Menu, Style, Animation

2024. 04. 22.

Gazdi László

gazdi.laszlo@aut.bme.hu



Automatizálási és
Alkalmazott
Informatikai Tanszék

Miről volt szó az előző órán?

- Activity Back Stack
- Navigálás Activity-k között
- Komponensek közti kommunikáció, Intent
- Erőforrás típusok
- ViewBinding
- Felhasználói felület alapok



Tartalom

- View/ViewGroup-ok
- Menü kezelés, Toolbar
- Felugró ablakok
- Stílusok&Témák
- Grafikai erőforrások
- Animációk

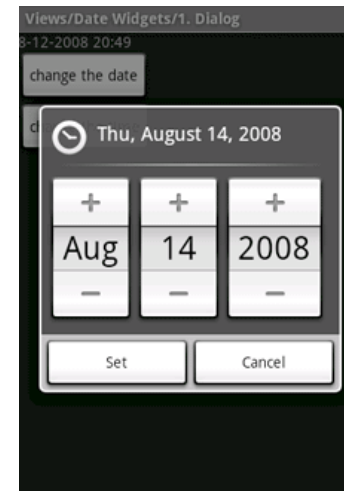
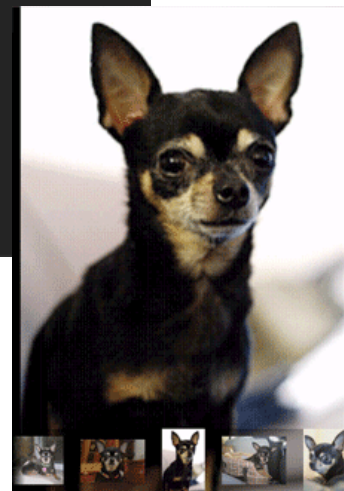
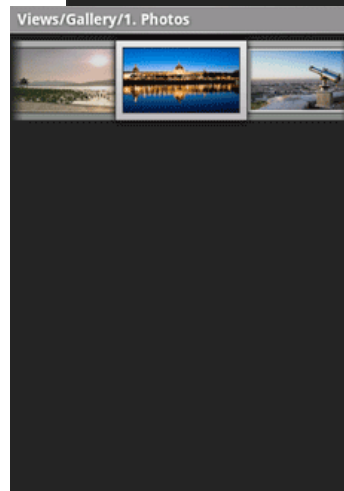
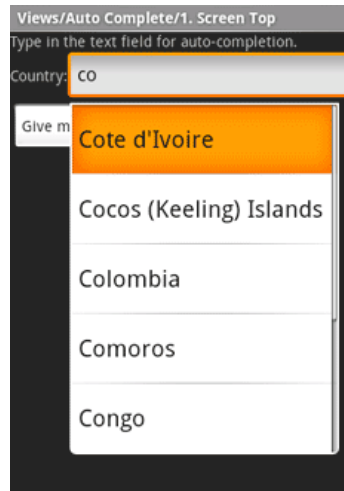
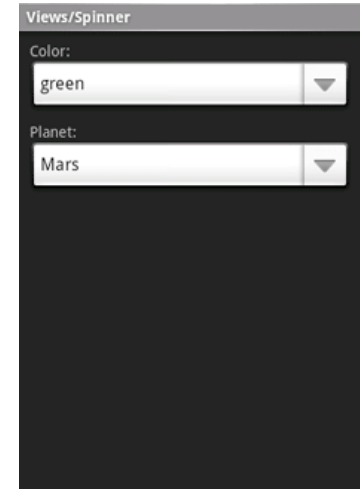
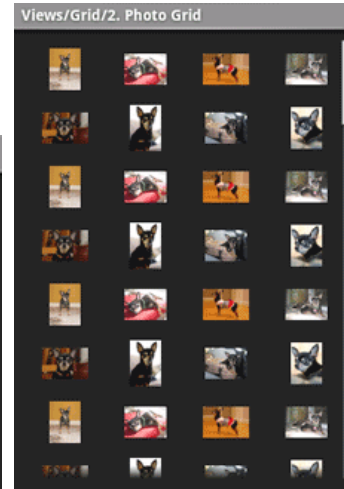
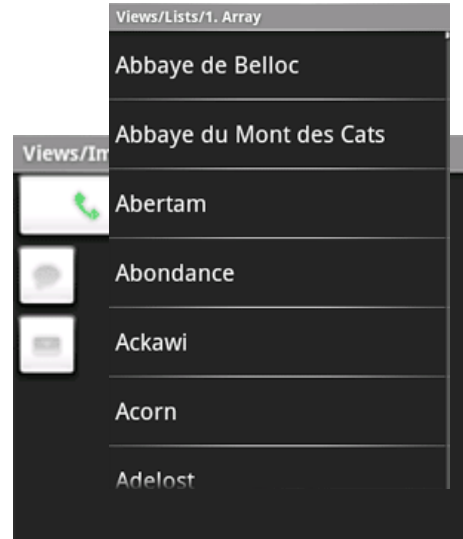
Nézetek (Widgetek/"View"-k)

View-k 1/2



- *Button, EditText, CheckBox, RadioButton, ToggleButton*
- *ImageButton*
- *ListView*
- *GridView*
- *Spinner*
- *AutoCompleteTextView*
- *Gallery*
- *ImageSwitcher*
- *DatePicker, TimePicker*

View-k 2/2



Dinamikus UI kezelés - LayoutInflater

- LayoutInflater feladata:
 - > XML-ben összeállított felületi elemek példányosítása
- Használati mód:

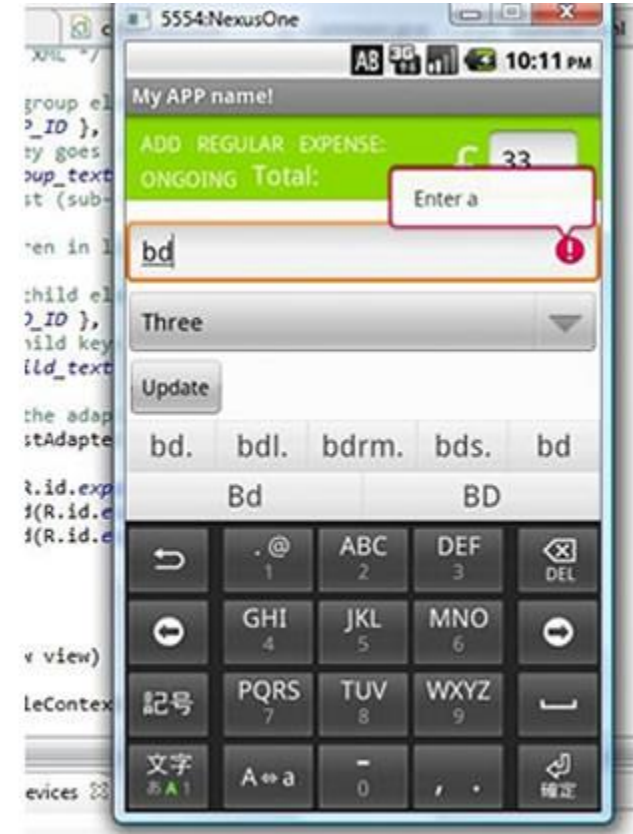
```
// érdemes megadni a layout-ot ahova kerülni  
// fog, hogy a méretezés optimális legyen  
val myView = getLayoutInflater()  
    .inflate(R.layout.activity_main, null)
```

Legfontosabb tényezők

- Használjuk a `wrap_content`, `match_parent`, vagy `dp` egységeket, amikor egy felületet készítünk!
- Súlyozás
- Ne használjunk beégetett pixel értékeket!
- Ne használjuk az `AbsoluteLayout`-ot (elavult)!
- Mindenképp készítsünk különböző kép erőforrásokat az eltérő képernyősűrűségekhez
- Szövegek méretezéséhez érdemes használni az `sp` (scale-independent pixel) mértéket, `dp`-hez hasonlóan működik

Validáció támogatása

- `setError(errorText)` (pl. EditText-nél)



Menük

- *ActionBar*->*Toolbar* része
- Menü definiálása kódból
- Menü definiálása erőforrásból
- Dinamikus menük
 - > Láthatóság beállítása
 - > Manipuláció Java kódból
- Almenük támogatása

Menü erőforrás

```
<menu xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:id="@+id/item1"
        android:title="@string/item1"/>
    <item android:id="@+id/item2"
        android:title="@string/item2"/>
    <item android:id="@+id/submenu"
        android:title="@string/submenu_title">
        <menu>
            <item android:id="@+id/submenu_item1"
                android:title="@string/submenu_item1" />
            <item android:id="@+id/submenu_item2"
                android:title="@string/submenu_item2" />
        </menu>
    </item>
</menu>
```

Menü kezelése

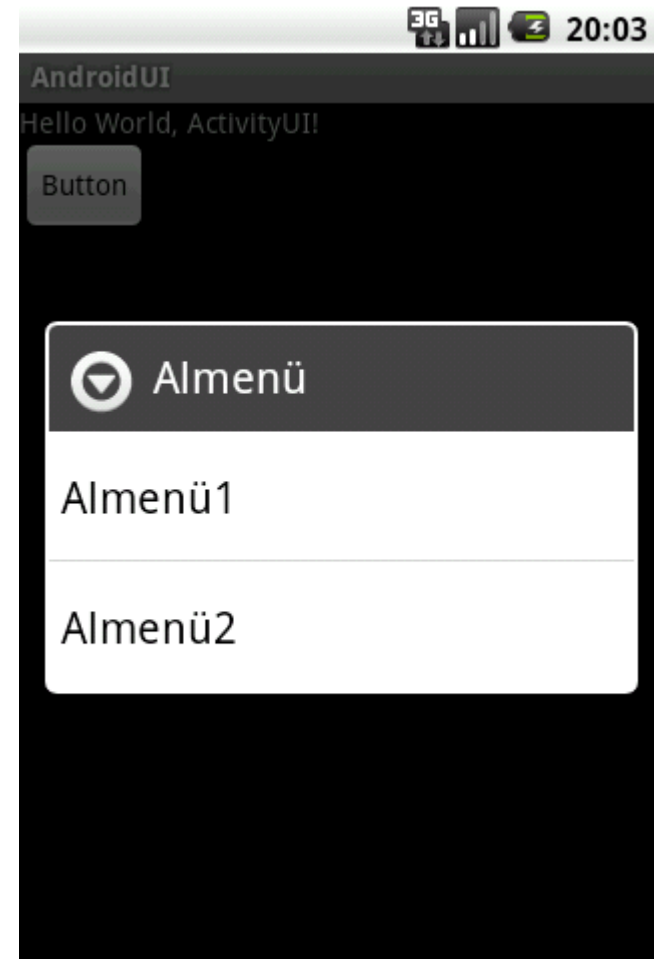
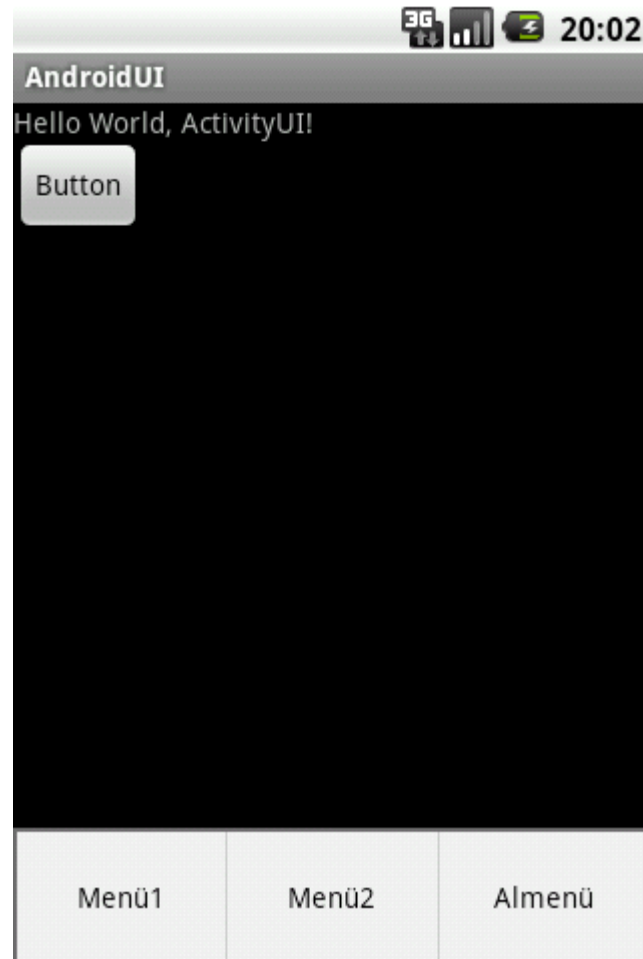
- Menü felhelyezése:

```
override fun onCreateOptionsMenu(menu: Menu): Boolean {  
    menuInflater.inflate(R.menu.mymenu, menu)  
    return true  
}
```

- Választás kezelése:

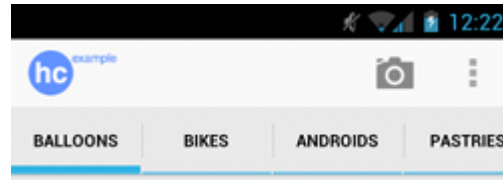
```
override fun onOptionsItemSelected(item: MenuItem): Boolean {  
    Toast.makeText(this@ActivityUI,  
        item.getTitle(), Toast.LENGTH_LONG).show()  
    return super.onOptionsItemSelected(item)  
}
```

Menü és almenü



ActionBar és Menük

- Dedikált alkalmazás menü, logo és cím
- Tipikus felhasználás:
 - > Menü
 - > Branding (logo/background) és alkalmazás ikon
 - > Konzisztens alkalmazás navigáció
 - > Fő funkciók bemutatása



ActionBar specifikus XML menü paraméterek

```
<item android:id="@+id/action_time"  
    android:title="@string/action_show_time"  
    android:orderInCategory="5"  
    android:icon="@drawable/clock_icon"  
    android:showAsAction="always|withText" />
```

ActionBar -> Toolbar

- ActionBar helyett Toolbar
- Sokkal dinamikusabb viselkedés
- Menü erőforrások támogatása
- Custom elemek támogatása
- Manuális pozicionálás
- Toolbar tutorialok:
 - > <http://javatechig.com/android/android-lollipop-toolbar-example>
 - > <http://www.101apps.co.za/index.php/articles/using-toolbars-in-your-apps.html>

Toolbar használat

- ActionBar nélküli téma(styles.xml):
 - > Theme.AppCompat.NoActionBar

- Layout erőforrás:

```
<androidx.appcompat.widget.Toolbar
    android:id="@+id/toolbar"
    android:minHeight="?attr/actionBarSize"
    android:background="#2196F3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">
</androidx.appcompat.widget.Toolbar>
```

- Activity onCreate(...):

```
lateinit var binding: ActivityMainBinding
```

```
override fun onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState)
    binding = ActivityMainBinding.inflate(layoutInflater)
    setContentView(binding.root)
```

```
// Set a toolbar to replace the action bar.
```

```
setSupportActionBar(binding.toolbar);
```

```
}
```

Mi igaz a felületi elemekre?

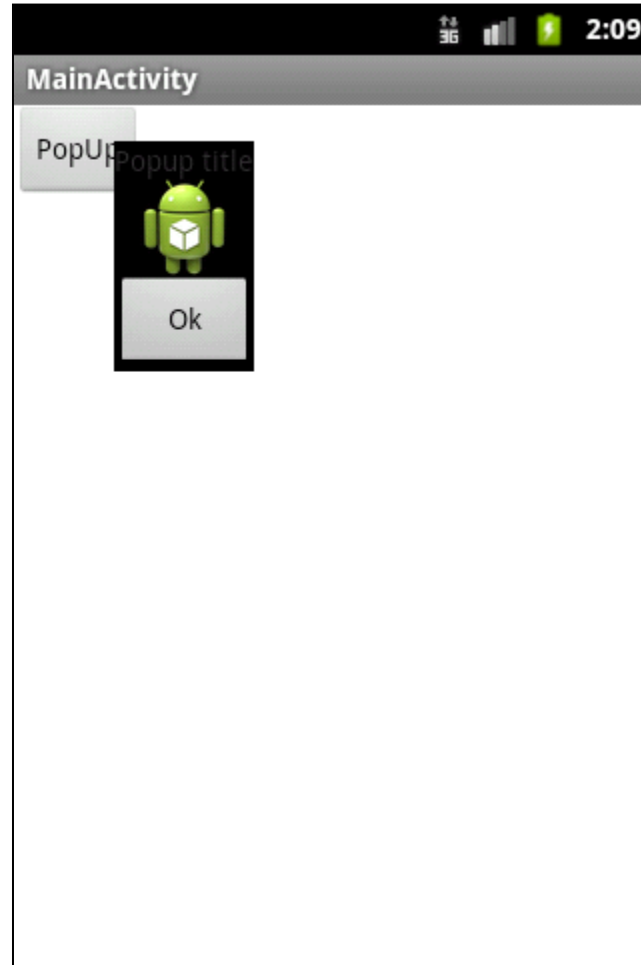
- A. A menü elemeket kódból kell felvennünk.
- B. A view-k méretét mindig pixelben adjuk meg.
- C. A szövegek méretezéséhez dp helyett sp-t használunk.
- D. A Toolbar nem támogatja a menü erőforrásokat.

Felugró ablakok

Felugró ablakok

- Activity felugró ablakban
 - > Manifest állományba:
`android:theme="@android:style/Theme.Dialog"`
- *PopupWindow*
 - > PopupWindow objektum létrehozása és layout beállítása
 - > Eseménykezelő a gombokra
- *AlertDialog*
 - > *AlertDialog.Builder*
- *Toast*
 - > Szöveg megjelenítése egy kis ablakban
- *SnackBar*

PopUpWindow példa



PopUp megjelenítése

```
fun showPopUp() {  
    val inflater = baseContext  
        .getSystemService(LAYOUT_INFLATER_SERVICE) as LayoutInflater  
    val popupView: View = inflater.inflate(  
        R.layout.popup, null  
    )  
    val popupWindow = PopupWindow(  
        popupView,  
        LayoutParams.WRAP_CONTENT, LayoutParams.WRAP_CONTENT  
    )  
  
    val btnDismiss: Button = popupView.findViewById<View>(R.id.btnPopUpOk) as Button  
    btnDismiss.setOnClickListener(object : OnClickListener() {  
        fun onClick(v: View?) {  
            popupWindow.dismiss()  
        }  
    })  
  
    popupWindow.showAsDropDown(btnShow, 50, -30)  
}
```

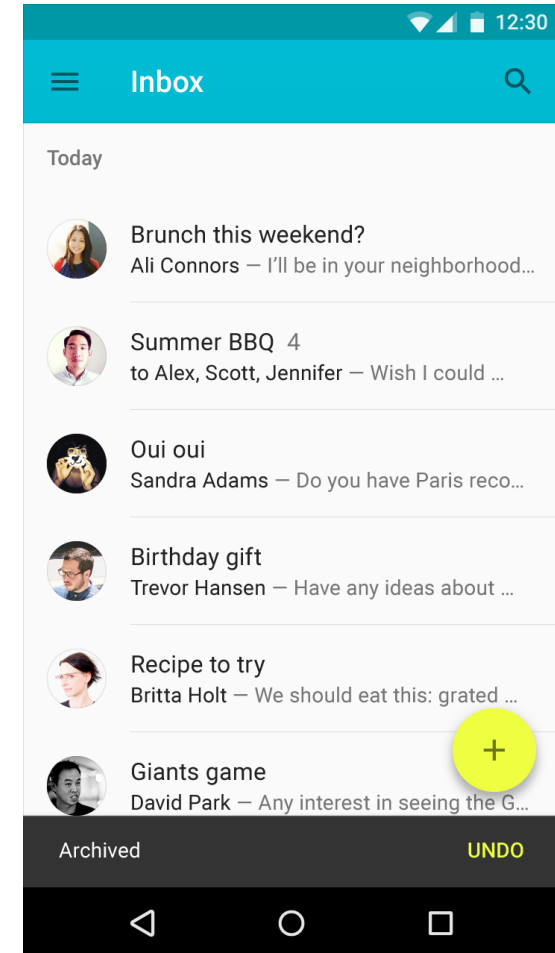
AlertDialog példa

```
private fun showAlertMessage(aMessage: String) {  
    val alertbox: AlertDialog.Builder = Builder(this)  
    alertbox.setMessage(aMessage)  
    alertbox.setNeutralButton("Ok",  
        DialogInterface.OnClickListener { arg0, arg1 ->  
            // Ok kiválasztva  
        })  
    alertbox.show()  
}
```

SnackBar

- Képernyő alján megjelenő információs sáv
 - > Toast helyett gyakran hasznos
 - > Egyedi akció

```
SnackBar.make (android.R.id.content,  
    "Demo message", SnackBar.LENGTH_LONG)  
    .setAction ("Undo", mOnClickListener)  
    .setActionTextColor (Color.RED) .show ()
```



Stílusok, témák

Stílusok készítése

- Stílus file: res/values/styles.xml

```
<style name="ExampleStyle">  
    <item name="android:textSize">22sp</item>  
    <item name="android:textColor">#0000EE</item>  
</style>
```

- Stílus használata:

```
<TextView  
    android:id="@+id/tvHello"  
    android:text="@string/hello_world"  
    style="@style/ExampleStyle" />
```

Saját témák használata

- Stílusokhoz hasonlóan definiálhatók:

```
<style name="CustomTheme" parent="android:Theme">  
  <item name="android:windowTitleSize">50dip</item>  
  <item name="android:textColor">#000000</item>  
  <item name=  
    "android:windowBackground">@color/white_color</item>  
</style>
```

- A témák öröklődhetnek egymásból
- Manifestben állítható be:

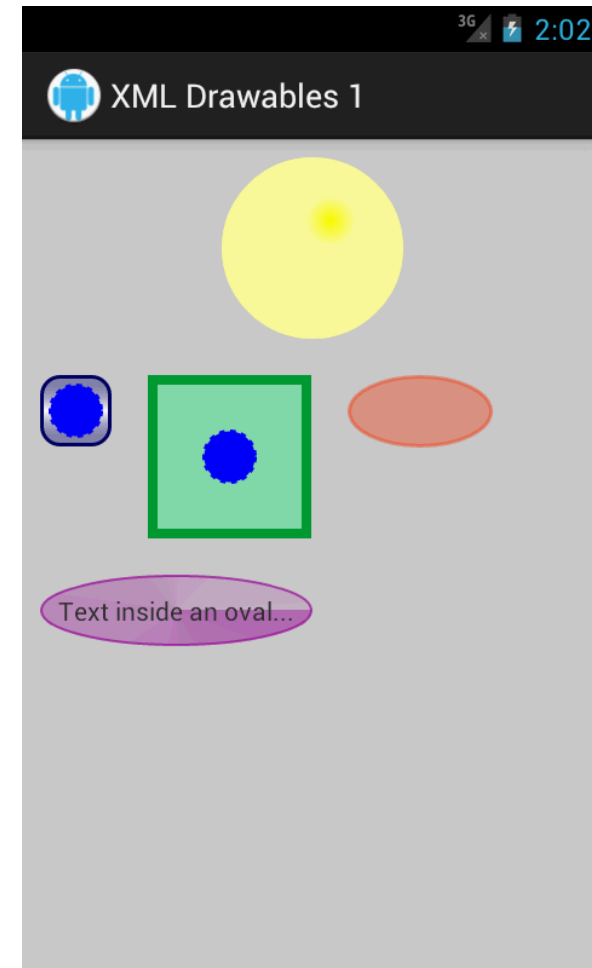
```
<activity android:label="" android:name=".MainActivity"  
  android:screenOrientation="portrait"  
  android:theme="@style/CustomTheme"/>
```

Grafikai erőforrások XML-ben

Grafikai erőforrások 1/2

- Grafikai elemek, hátterek is leírhatók XML-ben

```
<?xml version="1.0" encoding="utf-8" ?>
<shape
  xmlns:android="http://schemas.android.com/apk/res/android"
  android:shape="oval">
  <gradient
    android:type="radial"
    android:gradientRadius="20"
    android:centerX=".6"
    android:centerY=".35"
    android:startColor="#FFFF00"
    android:endColor="#FFFF99" />
  <size
    android:width="100dp"
    android:height="100dp"/>
</shape>
```

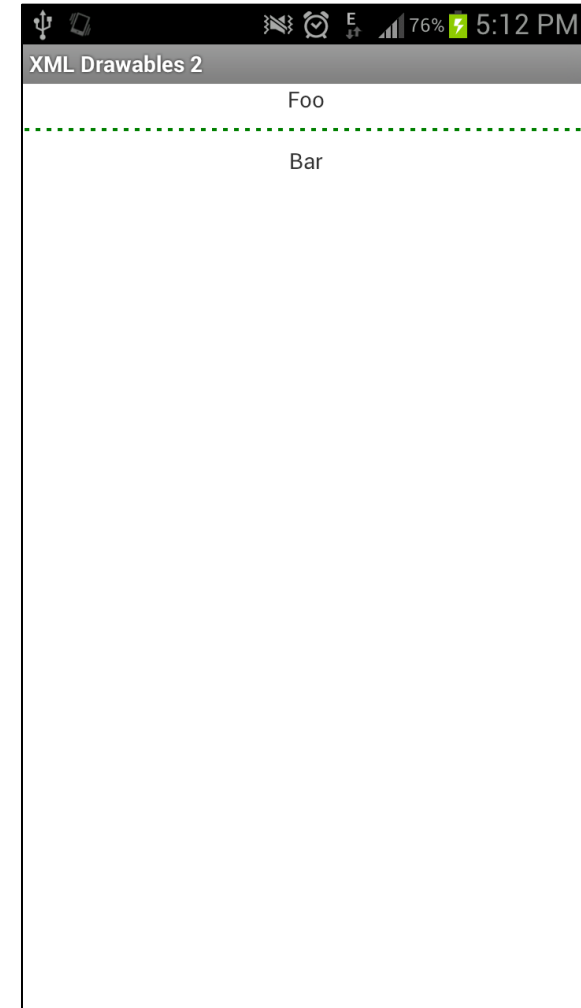


Grafikai erőforrások 2/2

- Zöld szaggatott vonal:

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="line">
    <stroke
        android:width="2dp"
        android:color="#008000"
        android:dashWidth="3dp"
        android:dashGap="4dp"/>

    <size android:height="20dp" />
</shape>
```



Állapotok leírása grafikai erőforrásokban

- Például gombfelirat színe:

- > res/color/button_text.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<selector xmlns:android="http://schemas.android.com/apk/res/android">
  <item android:state_pressed="true"
        android:color="#ffff0000"/> <!-- pressed -->
  <item android:state_focused="true"
        android:color="#ff0000ff"/> <!-- focused -->
  <item android:color="#ff000000"/> <!-- default -->
</selector>
```

- Használat:

```
<Button
  android:layout_width="match_parent"
  android:layout_height="wrap_content"
  android:text="@string/button_text"
  android:textColor="@color/button_text" />
```

Animációk

Animációk

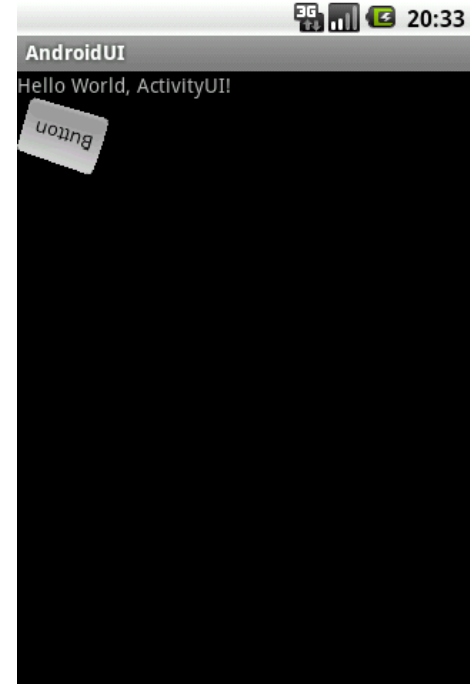
- Animációk támogatása
 - > XML erőforrás (*res/anim*)
 - > Programkód
- Layout animáció
 - > *Scale*
 - > *Rotate*
 - > *Translate*
 - > *Alpha*
- Három fő típus:
 - > Tween animáció
 - > Frame animáció
 - > Property animator

Tween animáció erőforrás

```
<?xml version="1.0" encoding="utf-8"?>
<set xmlns:android="http://schemas.android.com/apk/res/android"
    android:shareInterpolator="false">
    <scale
        android:interpolator=
            "@android:anim/accelerate_interpolator"
        android:fromXScale="0.0"
        android:toXScale="1.0"
        android:fromYScale="0.0"
        android:toYScale="1.0"
        android:pivotX="50%"
        android:pivotY="50%"
        android:duration="1000" />

    <alpha android:fromAlpha="0.0"
        android:toAlpha="1.0"
        android:duration="5000"/>

    <rotate
        android:interpolator="@android:anim/accelerate_interpolator"
        android:fromDegrees="0.0"
        android:toDegrees="360"
        android:pivotX="50%"
        android:pivotY="50%"
        android:duration="5000" />
</set>
```



Tween animáció lejátszása

```
class MainActivity : AppCompatActivity() {  
    override fun onCreate(savedInstanceState: Bundle?) {  
        super.onCreate(savedInstanceState)  
        setContentView(R.layout.activity_main)  
        var demoAnim = AnimationUtils.loadAnimation(this,  
            R.anim.demo_anim)  
        btnAnim.setOnClickListener {  
            btnAnim.startAnimation(demoAnim)  
        }  
    }  
}
```

Melyik állítás igaz?

- A. Egyszerű alakzatok megjelenítése is csak képi erőforrással lehetséges.
- B. Nem készíthető saját stílus, csak a beépítettek használhatók.
- C. A gombok állapotától függő színét selectorokkal egyszerűen állíthatjuk.
- D. Activity nem jelenhet meg dialógus formájában.

Hogy is volt?

- Mi a LayoutInflater feladata?
- Hogyan lehet elkészíteni és kezelni egy menüt Android platformon?
- Hogyan adható meg, hogy egy stílus az egész alkalmazásra érvényes legyen?

Összefoglalás

- View/ViewGroup-ok
- Menü kezelés, Toolbar
- Felugró ablakok
- Stílusok&Témák
- Grafikai erőforrások
- Animációk

Köszönöm a figyelmet!

