

KIV/MKZ, CVIČENÍ 2

verze 2016, v2

L. Pešička

CO VYZKOUŠÍME

- ◉ Volání aktivity s dialpadem
 - Jak z aplikace někomu zatelefonovat
 - Vyvolání intentu
- ◉ Modifikace UI pro landscape otočení
 - Jak vytvořit jiný layout při otočení telefonu
- ◉ Action bar
 - Obsluha položek menu
 - Přidání ikony
 - Navigace na parent aktivitu
- ◉ Aplikace z více aktivit
 - Kontrola vstupu uživatele

TELEFONOVÁNÍ

- VYVOLÁNÍ AKTIVITY DIALPADU

```
public void call (View v) {
```

```
    Intent callIntent = new  
    Intent(Intent.ACTION_DIAL);
```

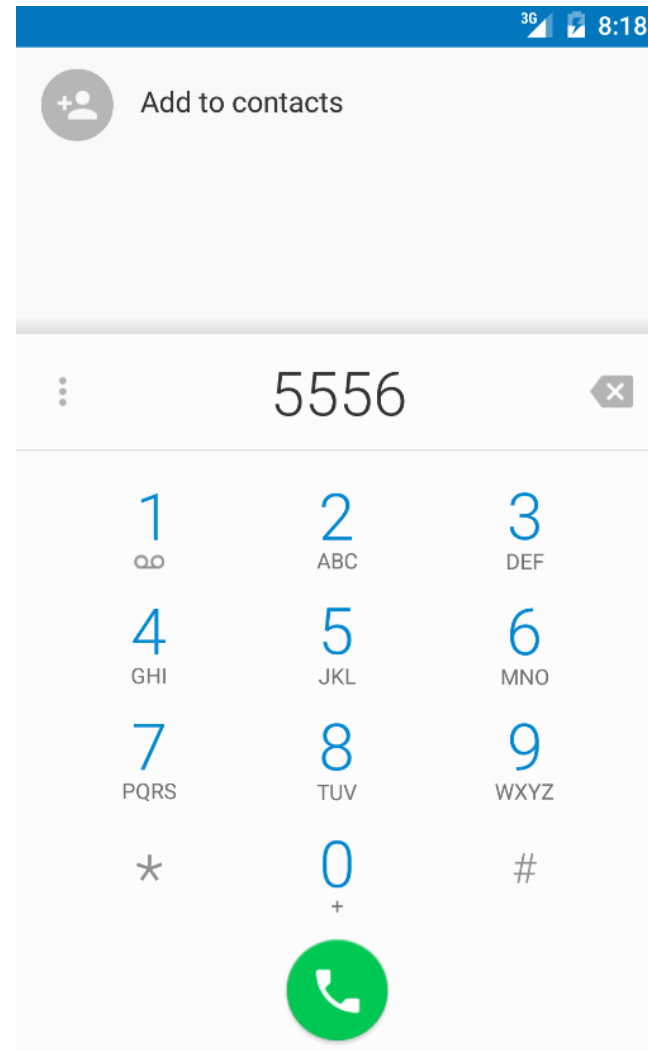
```
        callIntent.setData(Uri.parse("tel:5556"));  
        startActivity(callIntent);
```

```
}
```

DIALPAD

Po stisknutí tlačítka se zobrazí Dialpad

Dovolí uživateli volání zrušit, změnit akci
-> uživatel má pod kontrolou



CO KDYBYCHOM NECHTĚLI DIALPAD?

- ◉ Změnit akci na ACTION_CALL
 - Rovnou volá
- ◉ Vyžaduje přidat právo do manifestu:
 - `<uses-permission
android:name="android.permission.CALL_PHONE" />`
 - (přidáme před Application)
- ◉ Vhodnější je používat ACTION_DIAL

ZOBRAZENÍ LANDSCAPE

- ◉ Emulátor otočíme **CTRL+F11**
- ◉ Layout - pravá myš - **new layout resource file**
stejný název layout (content_main.xml), ale
přidat modifikátor
orientation landscape
- ◉ Můžeme navrhnout jiné UI
- ◉ Můžeme volat stejné obsluhy tlačítek
- ◉ Přidejte editační políčko s hintem

ACTION BAR - ÚPRAVA XML

- Chceme přidat ikonu do action baru pro vyhledávání
- Vložit obrázek do drawable (např. z ikon packu) s názvem action_search.png nebo .jpg
- Přidat řetězec action_search do strings.xml

```
<item android:id="@+id/action_search"  
    android:icon="@drawable/action_search"  
    android:title="@string/action_search"  
    app:showAsAction="ifRoom" />
```

POZNÁMKA

<http://stackoverflow.com/questions/28715765/what-is-orderincategory-in-actionbar-menu-item-why-it-is-use-for>

<item

```
    android:id="@+id/action_settings"  
    android:orderInCategory="100"  
    android:title="@string/action_settings"  
    app:showAsAction="never" />
```


OBSLUHA ACTION BARU

- Upravíme / přidáme obsluhu tlačítek zobrazením Toastu ve funkci `onOptionsItemSelected`:

```
if (id == R.id.action_settings) {  
    Toast.makeText(this, "Zvoleno Setting",  
    Toast.LENGTH_LONG).show();  
    return true;  
}  
if (id == R.id.action_search) {  
    Toast.makeText(this, "Zvoleno Search",  
    Toast.LENGTH_LONG).show();  
    return true;  
}
```



POZNÁMKA K TOASTU

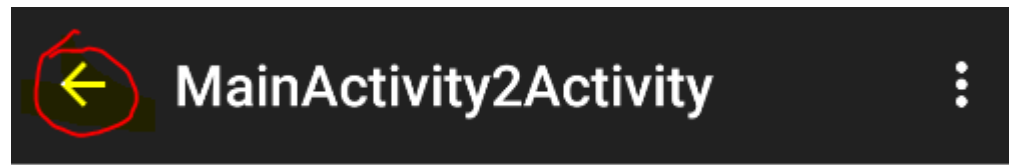
modifikace Toastu, aby používal řetězce ze strings.xml:

```
Toast.makeText(this, R.string.napis,  
Toast.LENGTH_LONG).show();
```

PŘIDÁNÍ DRUHÉ AKTIVITY

- ◉ Aktivita bude mít jako rodiče hlavní aktivitu
- ◉ New - Activity - Blank Activity
- ◉ Nastavíme Hierarchical Parent - MainActivity
- ◉ Jak se projeví v manifestu?

```
public void dalsiAktivita(View v) {  
    Intent i = new Intent(this, Main2Activity.class);  
    startActivity(i);  
}
```



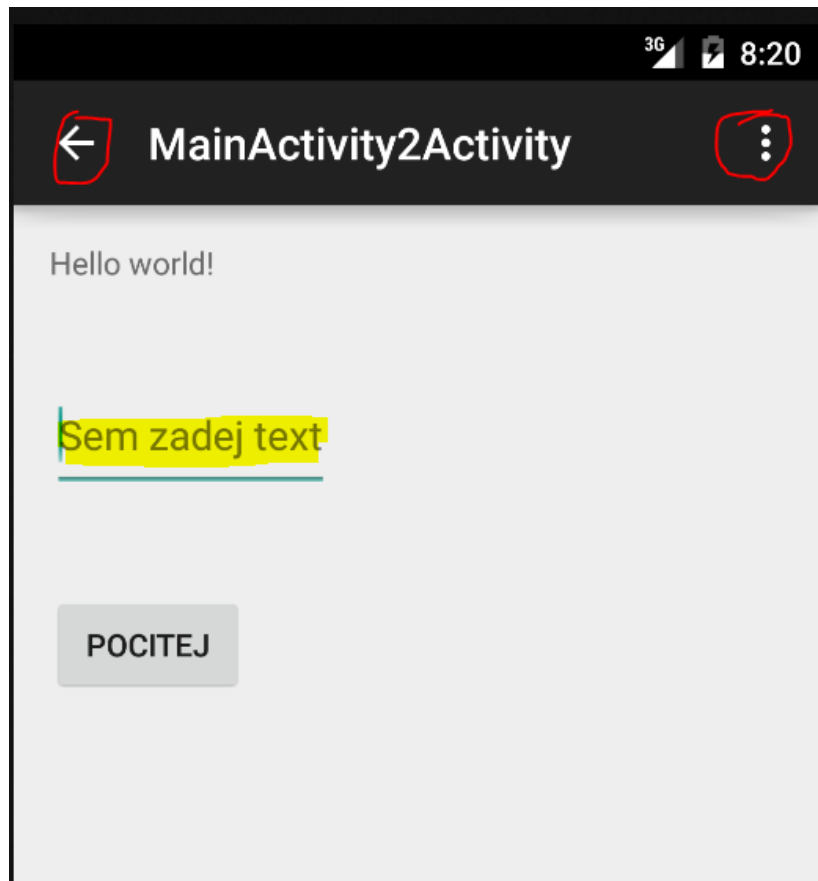
DRUHÁ AKTIVITA - POČÍTÁNÍ

- ◉ Editační pole pro vkládání textu
- ◉ (Text Field - Plain text, hint **Sem zadej text**)
- ◉ Tlačítko **pocitej**
- ◉ Lze omezit vstup uživatele např. volbou input type: numbers - přizpůsobí se tomu i klávesnice
- ◉ Výhoda hintu - nezabírá místo

OBSLUHA TLAČÍTKA DRUHÉ AKTIVITY

```
public void pocitej (View v) {  
    final EditText edit1 = (EditText)  
    findViewById(R.id.editText);  
    int c1, vysledek;  
    try {  
        c1 = Integer.parseInt(edit1.getText().toString());  
    } catch (NumberFormatException e) {  
        c1 = 0; edit1.setText(String.valueOf(0));  
    }  
    vysledek = c1 + 1;  
    edit1.setText(String.valueOf(vysledek));  
}
```

VZHLED DRUHÉ AKTIVITY



VYZKOUŠET SAMPLE APLIKACI

- ◉ Uzavřít projekt
- ◉ Import an Android code sample
 - [Action Bar Compat - List Popu Menu](#)
- ◉ Je zde hodně ukázkových aplikací, z kterých je možné se poučit, co a jak v Androidu udělat