

Univerza v Ljubljani  
Fakulteta *za računalništvo  
in informatiko*



# Elektronsko in mobilno poslovanje

## **VAJE 2**

Jan Meznarič  
Simon Vrhovc

Ljubljana  
2. 11. 2016



# Pred začetkom razvoja

- Na računalnikih v učilnicah
  - Poženite Android Studio
  - Ob prvem zagonu se prikaže čarovnik za konfiguracijo
  - Izberite *custom* nastavitve
  - Za lokacijo SDK izberite mapo *F:\androidsdk*
  - Izberite prenos SDKja in emulatorja
  - dokončajte namestitev





# Pred začetkom razvoja

- Programiranje v Java 7
- Android Studio
  - Android Software Development Kit (SDK)
    - Vsako različico moramo prenesti posebej
  - Emulator
- Nastavitev virtualne naprave (emulator)
  - Tools -> Android -> AVD Manager
    - Android Virtual Device – AVD
  - Z uporabo Intel x86 Emulator Accelerator je mogoče delovanje emulatorja močno pohitriti
    - Emulator Accelerator morate posebej namestiti in ponovno zagnati računalnik. Nato kot ABI izberemo Intel x86 ali Intel x86\_64
    - <http://software.intel.com/en-us/articles/installation-instructions-for-intel-hardware-accelerated-execution-manager-microsoft>
- Pripravite Virtualno napravo za Android 6.0





# Bistveni gradniki projekta

- **java** – mapa v kateri se nahajajo datoteke z izvorno kodo
- **res** – viri (npr. slike, grafični vmesnik v xml, znakovni nizi itn)
- **manifests/AndroidManifest.xml** – definicije aktivnosti, politike dostopa: do podatkov, do storitev mobilne naprave (npr. splet, GPS,...) itn.
- **[vidno v pogledu *Project*] app/build** – generirane Java datoteke (R.java), ki se avtomatično ustvarijo za referenciranje na resurse (npr. slike, tekstovni nizi, ...) projekta.





# Vaja 1

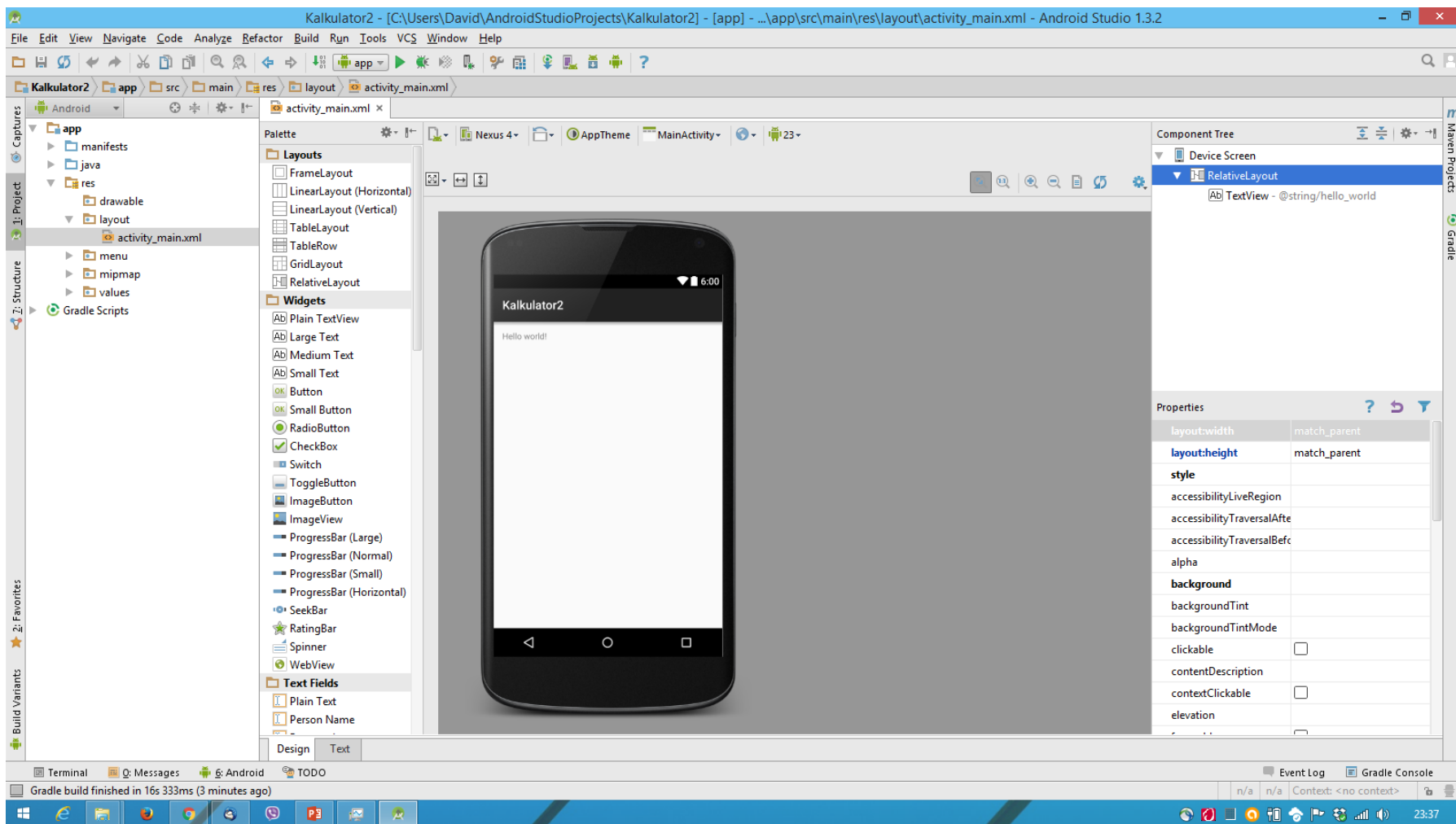
- Kreirajte nov Android projekt:
  - Ime: Kalkulator1
  - Package: si.fri.emp.vaje2
  - Minimum SDK: API 23 Android 6.0
- Preglejte generirano kodo
- Poženite projekt v emulatorju





# Grafični vmesnik

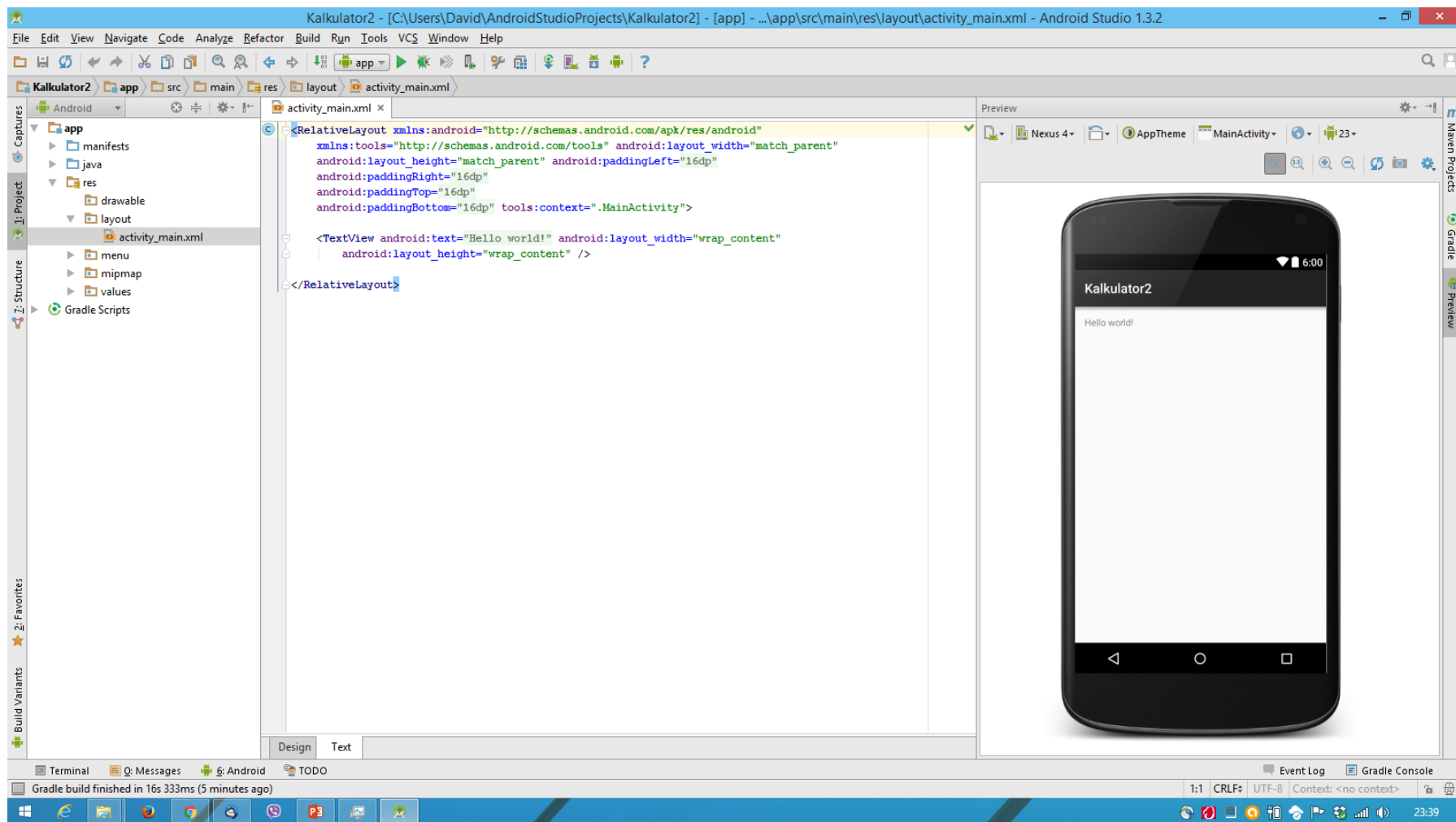
- Zapisan je v **./res**; različni elementi so porazdeljeni po podimenikih





# Grafični vmesnik

- Napredno urejanje poteka neposredno v XML datotekah, ki so v imeniku **./res**





# Grafični vmesnik

- Po potrebi lahko ustvarjamo prilagojene datoteke GUI v XML za različne gostote in velikosti zaslona ter za orientacijo. Možno je tudi dodatno prilagajanje za posamezne različice Androida.
- Primeri:
  - res/layout-xlarge-land
  - res/drawable-v12
  - res/layout-sw600dp (od Androida 3.2 dalje)







# Razhroščevanje

- `System.out.print()`
- LogCat
  - <http://developer.android.com/tools/help/logcat.html>
- Debug
- Napredno razhroščevanje z DDMS
  - Tools > Android Device Monitor
  - <http://developer.android.com/tools/debugging/ddms.html>





# Povezava med razvojnim okoljem in emulatorjem/napravo

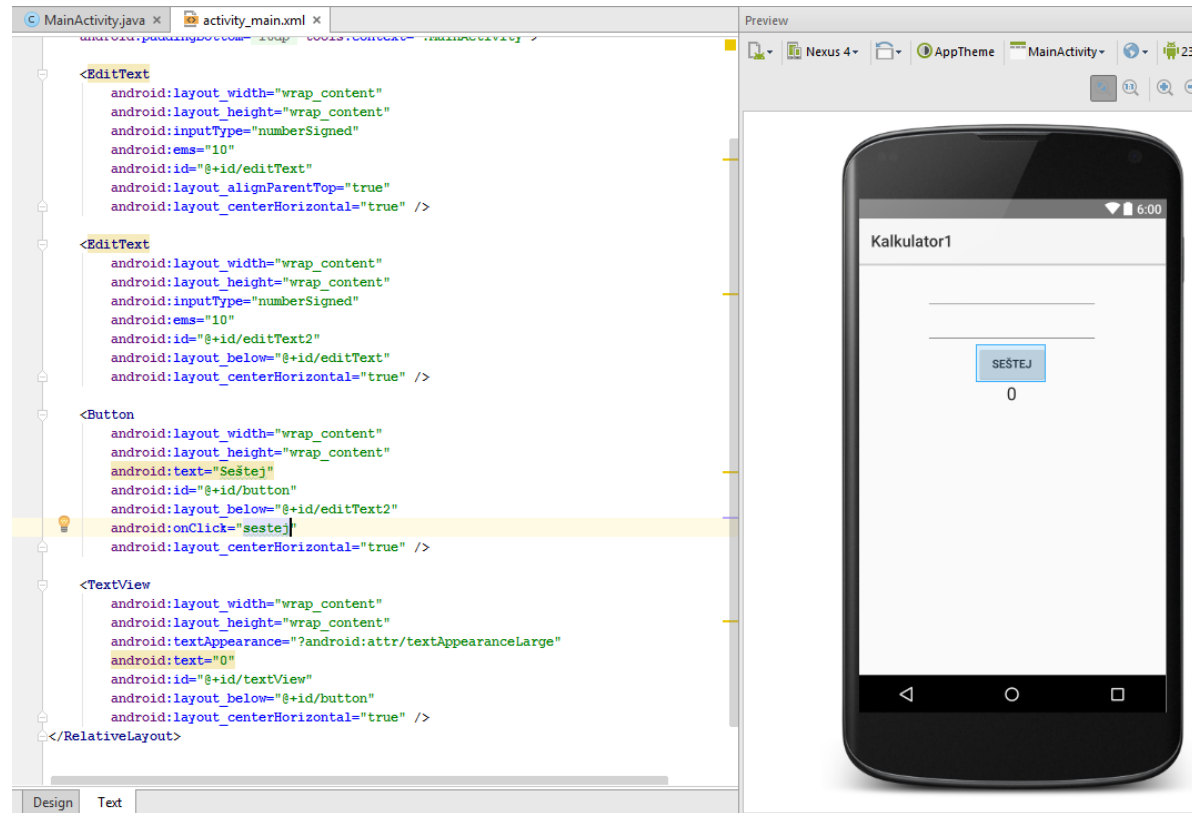
- Orodje ADB (Android Debug Bridge)
  - C:\<android-sdk>\platform-tools
  - Omogoča celovit nadzor nad napravo - velik del funkcij je dostopen preko GUI razvojnega okolja
  - Del funkcij je dostopen le preko ukazne vrstice
- Primer: Preprosto „naključno“ preizkušanje:
  - adb shell monkey -p mojevaje.vaje1\_calc -v 500
  - <http://developer.android.com/tools/help/monkey.html>

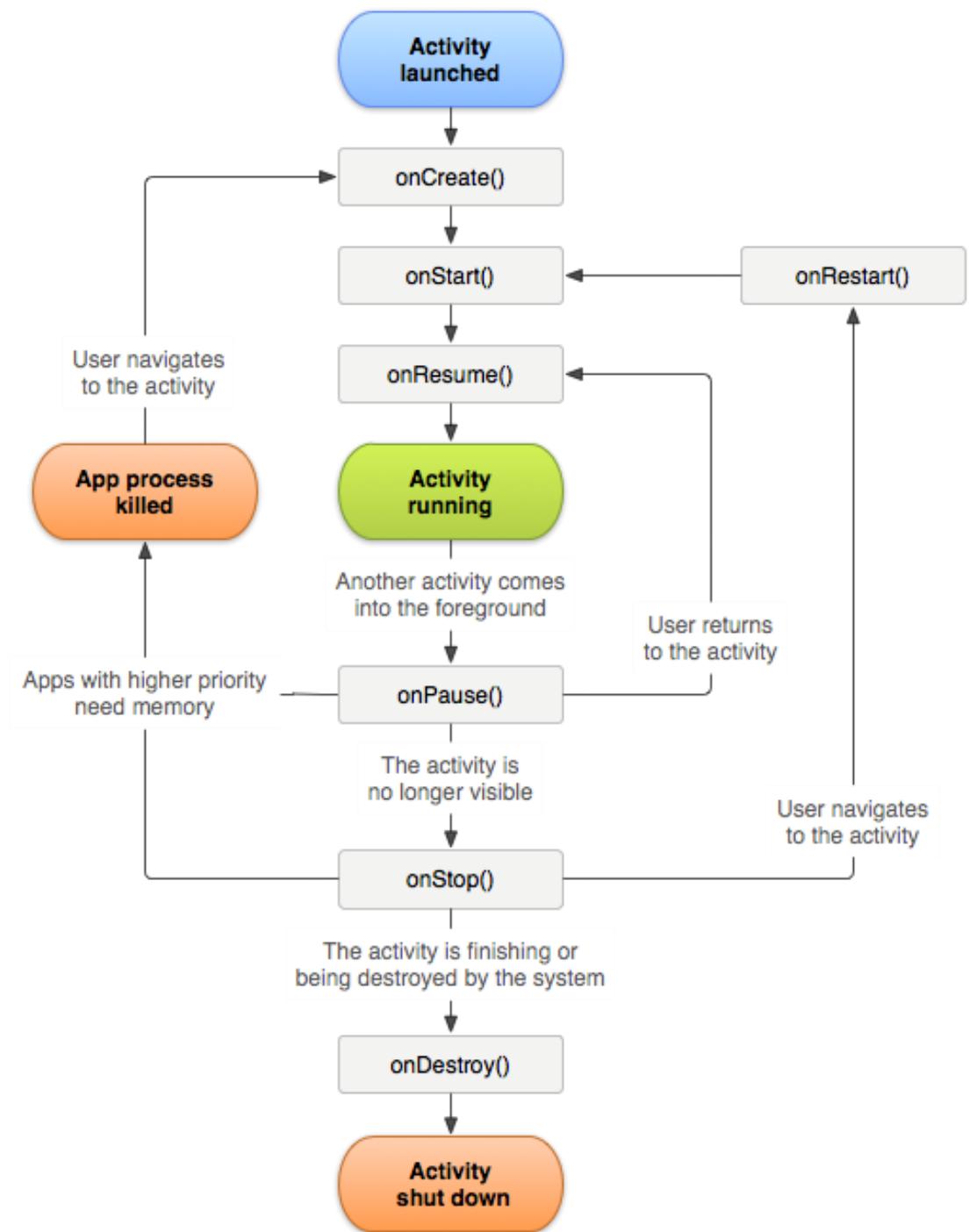




# Vaja 2

- Izdelajmo **seštevalnik**
- Odprite nov android projekt
- Na UI dodajte:
  - 2x Text Field  
(Number signed)
  - 1x Button
  - 1x Text View







## Vaja 2

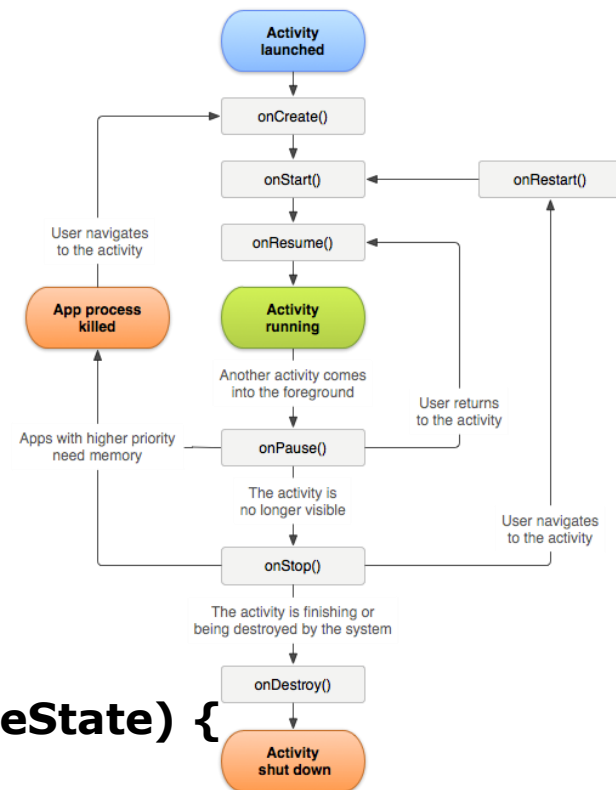
- Odprite *.java* datoteko, ki vsebuje kodo osnovne aktivnosti

```
public class MainActivity extends Activity {  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
    }  
}
```

GUI, ki se bo prikazal je definiran v v main.xml

onCreate call-back

Naša nova aktivnost





## Vaja 2

- V onCreate za klic setContentView dodajte naslednjo kodo:

```
Button bt=(Button)findViewById(R.id.button1);
bt.setOnClickListener(new OnClickListener()
{
    public void onClick(View v)
    {
        EditText ed1=(EditText)findViewById(R.id.editText1);
        EditText ed2=(EditText)findViewById(R.id.editText2);
        TextView tv=(TextView)findViewById(R.id.textView1);

        tv.setText(String.valueOf(Integer.parseInt(
            ed1.getText().toString()+Integer.parseInt(ed2.getText().toString()))));
    }
});
```

Kreiramo listener za pritisk na tipko button1

Iz GUI „poberemo“ posamezne kontrole

Konvertiramo in aritmetično seštejemo vsebino EditText-ov

Ne pozabite na import! (alt+enter)