# Android alkalmazásfejlesztés

Adattárolás

Szenzorok

OE-NIK

2011. október 5.

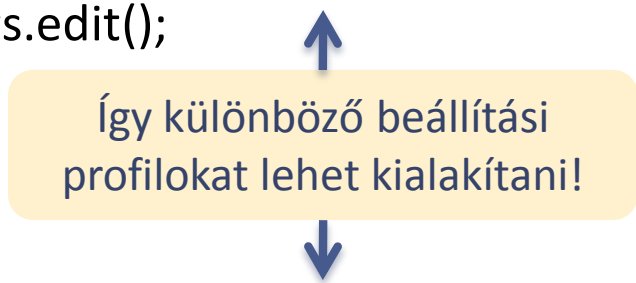
**Sicz-Mesziár János**

sicz-mesziar.janos@nik.uni-obuda.hu

# Shared Preference

- Egy egyszerű megoldás primitív értékek perzisztens tárolására.
- Az alábbiakat lehet: boolean, float, int, long, string.
- Felhasználói beállításoknál is erre építenek, de arra van egységes megoldás!
- Tárolás:

```
String PrefFileName = "MyPrefName";
SharedPreferences settings = getSharedPreferences(PrefFileName, 0);
SharedPreferences.Editor editor = settings.edit();
editor.putInt("test", 1027);
editor.commit();
```




- Visszaállítás:

```
SharedPreferences settings = getSharedPreferences(PrefFileName, 0);
int szam = settings.getInt("test", 0);
```

# Belső tároló

- Fájlmentés közvetlenül a belsőtárolóra.
- Ha az alkalmazást törlik, akkor ezek a fájlok is törlődnek!
- *Alapértelmezetten* más alkalmazás nem fér hozzá.
- Tárolás:

```
String FILENAME = "StorageFile„;
String saveThis = "Helló Storage!";
FileOutputStream fos = openFileOutput(FILENAME, MODE_PRIVATE);
fos.write(saveThis.getBytes());
fos.close();
```

- Betöltés:

/data/data/hu.uniobuda.nik.StorageTest/files/StorageFile

```
FileInputStream fis = openFileInput(FILENAME);
byte[] buffer = new byte[1024];   int len;
while((len = fis.read(buffer)) > 0)
        Log.d("NIK", new String(buffer, 0, len));
fis.close();
```

- Továbbá: fileList() , deleteFile(), getDir(), …

# Külső tároló (SD-kártya)

- Bármely másik alkalmazás (v. felhasználó) által olvasható és írható, osztott háttértár.
- Bármikor eltávolíthatja a felhasználó.

Ellenőrizni kell az elérhetőségét és állapotát:

```
String state = Environment.getExternalStorageState();
if(state.equals(Environment.MEDIA_MOUNTED)){
        // Elérhető, írható és olvasható
}else if(state.equals(Environment.MEDIA_MOUNTED_READ_ONLY)){
        // Elérhető, de csak olvasható
}else{   // Más állapotban van... DE se nem írható, se nem olvasható  }

String sdcard =
Environment.getExternalStorageDirectory().getAbsolutePath();
```

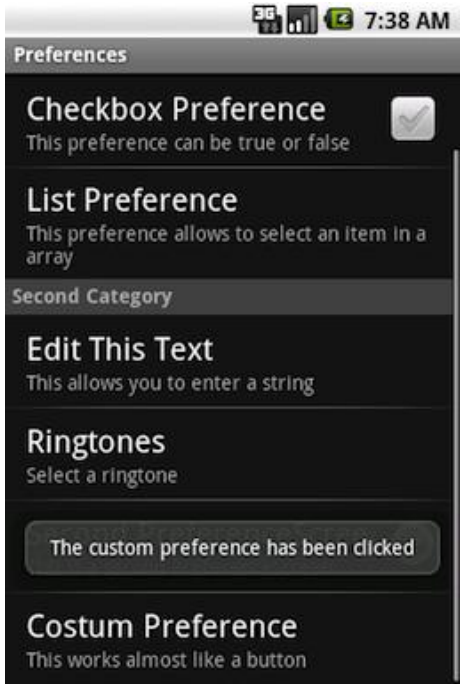
Jogosultság: **android.permission.WRITE_EXTERNAL_STORAGE**

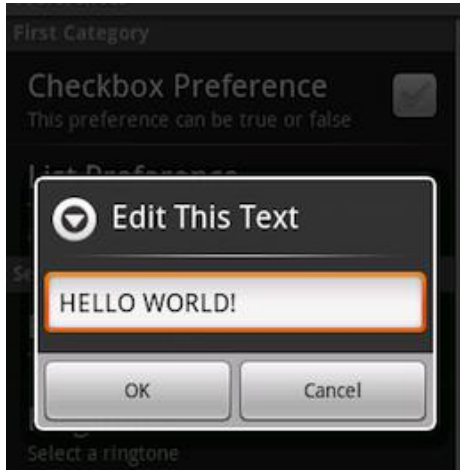
# Adatbázis használata (SQLite)

- Teljes SQLite támogatás.
- Alkalmazáson belül bárhonnan elérhető, de azon kívül nem!
- De ContentProvider-ek segítségével lehetséges az alkalmazások közötti adatcsere is.
  Pl.: Böngésző könyvjelzők, előzmények, kontaktok, …
- Megszokott SQL kódok használata: INSERT, SELECT, UPDATE, DELETE, CREATE, …

- Bővebben:
  - Alapok, egyszerű SQLite megoldás
  - SQLiteOpenHelper használata
  - Content Provider

# PreferenceActivity




- Kimondottan felhasználói beállítások tárolása.
- Néhány beépített form:
  CheckBoxPreference, EditTextPreference, ListPreference, RingtonePreference + **Egyéni!**
- Témák szerint kategorizálhatunk.
- Preference XML létrehozása:

1. File > New > Other… > Android XML File
2. Layout helyett **Preference** típus!
3. Add > ChechBoxPreference, …

- Java osztály:

1. extends PreferenceActivity
2. onCreate() implementálása
3. addPreferencesFromResource()

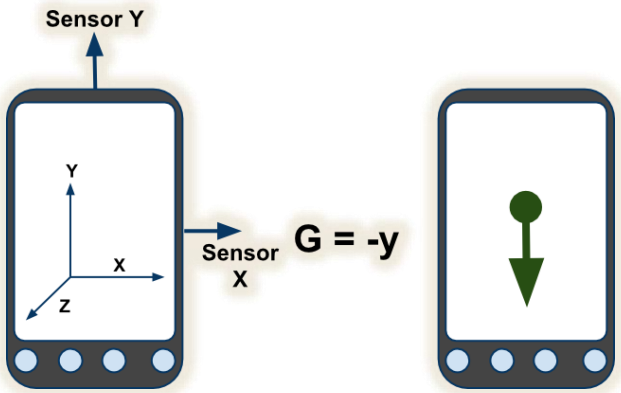
+ AndroidManifest.xml

# Szenzorok

# Accelerometer

Szenzor teszt videó ☺

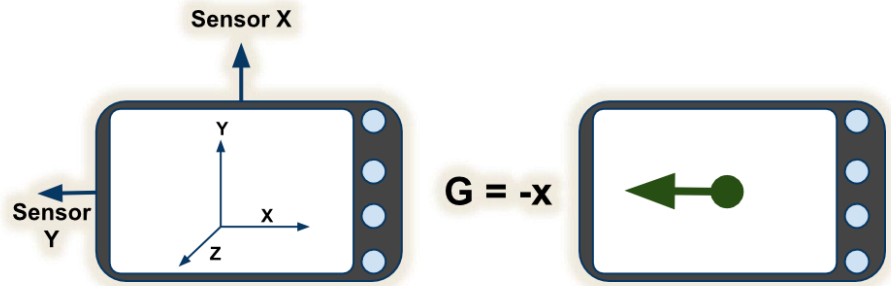
- Gyorsulásmérő, tipikusan egy BMA150-es szenzor
- Android készülékekben 3-tengelyű gyorsulásmérő




Mértékegysége: m/s^2
G = -9.81 m/s^2

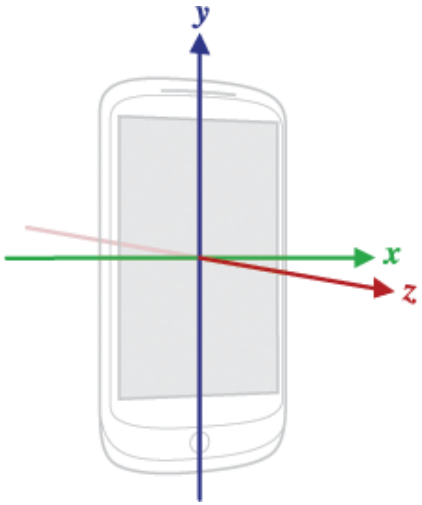
- SensorManager.remapCoordinateSystem()

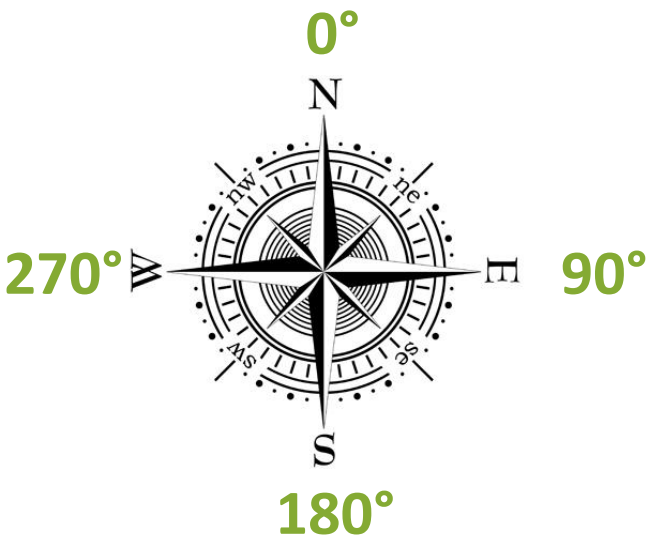
  Segítségével a koordináták transzformálhatóak.
  Pl.: telefon megdöntése miatt (landscape)



# Orientation sensor

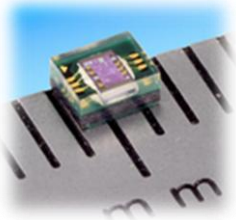
- Irány szenzor
- Mértékegység fokban értendő
- Digitális iránytű
- X, Y és Z tengely:
  - values[0]: Azimuth (0 - 359):
    - 0 = Észak
    - 90 = Kelet
    - 180 = Dél
    - 270 = Nyugat
  - values[1]: Pitch (-180 – +180)
  - values[2]: Roll (-90 – +90)
- Az értékek ugyanúgy remap…() segítségével transzformálhatóak



# További szenzorok Androidon

- Magneticfield
  - X, Y és Z tengelyen mért mágneses mező
- Proximity
  - Közelség érzékelő
- Temperature
  - Hőmérséklet érzékelése
- Light
  - Fényszenzor
- Pressure



- Kamera
- Mikrofon
- Touchscreen

Szenzorok működéséről részletesebben:
http://www.youtube.com/watch?v=C7JQ7Rpwn2k

# Szenzorok elérése gyakorlatban

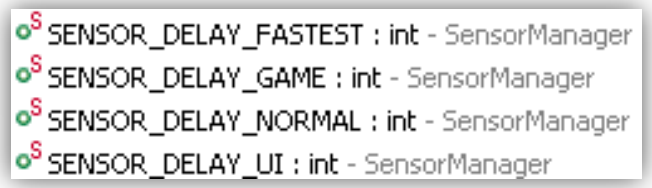
- Jogosultság kérése ebben az esetben nem szükséges.
- SensorManager példányosítása:

```
SensorManager manager =
  (SensorManager)getSystemService(SENSOR_SERVICE);
```

- SensorEventListener implementálása:

```
SensorEventListener listener = new
      SensorEventListener(){
  public void onSensorChanged(SensorEvent event){}
  public void onAccuracyChanged(Sensor s, int a){}
};
```

- Feliratkozás a szenzor adatokra




```
manager.registerListener(
      listener,
      manager.getDefaultSensor(Sensor.TYPE_PROXIMITY),
      SensorManager.SENSOR_DELAY_FASTEST
);
```