

Android Student Club @ University of Bucharest

04.05.2015



Org.

- prezență
- temă proiect
- întrebări pe grup?!

hackathon: 16 mai, ora 9:00

Lecția 2

Logs

Logcat (tab Android, Android Device Monitor)

- verbose, debug, error, warning, info

<http://developer.android.com/reference/android/util/Log.html>

Lecția 2

NetworkOnMainException

Threads - Main vs Background

Main (UI Thread): user inputs & user outputs

sincronizarea thread-urilor: *AsyncTask*

Lecția 2

AsyncTask: <http://developer.android.com/reference/android/os/AsyncTask.html>

- onPreExecute()
- doInBackground()
- onProgressUpdate()
- onPostExecute()

} main / background?



AsyncTask

- **onPreExecute()**
 - main, înainte de task- folosit pentru setup
- **doInBackground()**
 - background, thread separat
- **onProgressUpdate()**
 - main, arată progresul unui task, poate fi folosit pentru un loading bar
- **onPostExecute()**
 - main, ce se execută după finalizarea taskului

!!! se distruge odată cu activitatea de care este legat task-ul

Meniul

- Folder: res/menu
- Pentru fragmente: `setHasOptionsMenu(true)`
- Inflate: `onCreateOptionsMenu`
 - `inflater.inflate(R.menu.forecastfragment, menu);`

values/strings.xml

- aplicație tradusă
- string folosit în mai multe locuri - un singur loc de modificat

Permissions

- se declară în ..
- Exemple?
 - ..
 - ..
 - ..



JSON

[sursa](#)

The following JSON example defines an employees object, with an array of 3 employee records:

JSON Example

```
{ "employees": [
  { "firstName": "John", "lastName": "Doe" },
  { "firstName": "Anna", "lastName": "Smith" },
  { "firstName": "Peter", "lastName": "Jones" }
]}
```

The following XML example also defines an employees object with 3 employee records:

XML Example

```
<employees>
  <employee>
    <firstName>John</firstName> <lastName>Doe</lastName>
  </employee>
  <employee>
    <firstName>Anna</firstName> <lastName>Smith</lastName>
  </employee>
  <employee>
    <firstName>Peter</firstName> <lastName>Jones</lastName>
  </employee>
</employees>
```

What is JSON?

- JSON stands for **J**ava**S**cript **O**bject **N**otation
- JSON is a lightweight data-interchange format
- JSON is language independent *
- JSON is "self-describing" and easy to understand

Lecția 3

- Activity - unde trebuie declarate toate activitățile?
- Explicit Intents vs Implicit Intents
 - <http://developer.android.com/reference/android/content/Intent.html>
 - <https://developer.android.com/guide/components/intents-common.html>



Lecția următoare

- *Activity lifecycle and background activity termination*
- *SQLite databases and JUnit tests*
- *Creating and using a Content Provider as an abstraction layer*
- *Using Loaders to asynchronously load data*
- *Creating Adapters to bind UI components to Content Providers*

Până săptămâna viitoare:

- lecția 3
- lecția 4a
- mock-ups pentru aplicația voastră: xml sau desene
 - [Lucid chart](#)
 - [FLUID UI](#)

Linkuri utile

- <http://developer.android.com/guide/topics/providers/content-providers.html>
- <http://www.vogella.com/tutorials/AndroidSQLite/article.html>
- <https://www.udacity.com/course/viewer#!/c-ud258/l-3372188753>