Python scripting sto Android xrhsimopoiontas SL4A kai embedded python interpreter

References:

https://code.google.com/p/android-scripting/wiki/ApiReference

Examples

Just a hi

Toasts

```
import android

droid = android.Android()
droid.makeToast('hello happy android')
```

or Notifications

```
import android
droid = android.Android()

droid.notify('hi notification', "hi from python")
```

Hi over loop

```
import android, time

droid = android.Android()

droid.makeToast('hello happy android...')
time.sleep(5)
droid.makeToast('...and hello again')
```

Counting to 10

```
import android

droid = android.Android()

droid.makeToast('happy android count for me...')

for i in range(10):
    droid.makeToast('counting: ' + str(i))
```

Check if file exists

```
import android
import os

droid = android.Android()

filename = '/mnt'

if not os.path.exists(filename):
    droid.makeToast('file: ' + filename +' does not exists')
else:
    droid.makeToast('file ' + filename + ' exists')
```

Get Input

```
import android
droid = android.Android()
name = droid.getInput('Hello!', 'What is your name?')
print name # name is a named tuple
droid.makeToast('Hello, %s' % name.result)
```

Get Location (Gps or network)

```
import android, datetime, time

droid = android.Android()

droid.startLocating()
time.sleep(15)

loc = droid.readLocation()

droid.stopLocating()

if 'gps' in loc.result:
        lat = str(loc.result['gps']['latitude'])
        lon = str(loc.result['gps']['longitude'])
else:
        lat = str(loc.result['network']['latitude'])
        lon = str(loc.result['network']['longitude'])
now = str(datetime.datetime.now())
```

```
droid.makeToast('I am here: ' + now + ' ' + lat + ' ' + lon)
```

Communication with other activities

```
makeIntent(
 String action,
 String uri[optional],
 String type[optional]: MIME type/subtype of the URI,
 JSONObject extras[optional]: a Map of extras to add to the Intent,
 JSONArray categories[optional]: a List of categories to add to the Intent,
 String packagename[optional]: name of package. If used, requires classname to
be useful,
 String classname[optional]: name of class. If used, requires packagename to be
useful,
 Integer flags[optional]: Intent flags)
Create an Intent.
Returns:
  An object representing an Intent
```

Read Sensors (accelerometer, magnetometer, light)

```
import android, time
droid = android.Android()

droid.startSensingThreshold(1, 0, 7)

time.sleep(5)

res = droid.readSensors()

droid.stopSensing()

droid.makeToast(str(res))
```

Read Magnetometer

```
import android, time
droid = android.Android()

droid.startSensingThreshold(1, 0, 7)

time.sleep(5)
```

```
res = droid.sensorsReadMagnetometer().result

droid.stopSensing()

droid.makeToast(str(res))
```

Read Accelearometer

```
import android, time
droid = android.Android()

droid.startSensingThreshold(1, 0, 7)

time.sleep(5)

res = droid.sensorsReadAccelerometer().result

droid.stopSensing()

droid.makeToast(str(res))
```

Battery Level

```
import android, time
droid = android.Android()

droid.batteryStartMonitoring()

time.sleep(5)

res = droid.batteryGetLevel()

droid.batteryStopMonitoring()

droid.makeToast(str(res.result))
```

Dynamix Framework

dynamix plugin

dynamix demo