

# Desarrollo de aplicaciones android con python

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



**PyConES**  
2013

# Pequeña biografía - @SankoSK

Autodidácta, amante del software libre, constante aprendiz sobre programación y seguridad informática y miembro de @GrampusTeam.

Echale un ojo a nuestros proyectos en los repositorios de bitbucket::  
<https://bitbucket.org/grampusteam/>



# ÍNDICE :

- ¿Qué es Android? (Por si vives en una cueva)
- ¿Desarrollar aplicaciones en Python? ¿En serio?
  - SL4A (Scripting Layer for Android)
    - Cómo funciona y qué nos permite
    - Primer vistazo a la API
  - Entornos para un desarrollo cómodo:
    - ADB y emulador android
  - Distribuir nuestros scripts al público:
    - Generar Barcode
    - Empaquetando en .apk
    - Otras formas de desarrollo, kivy framework

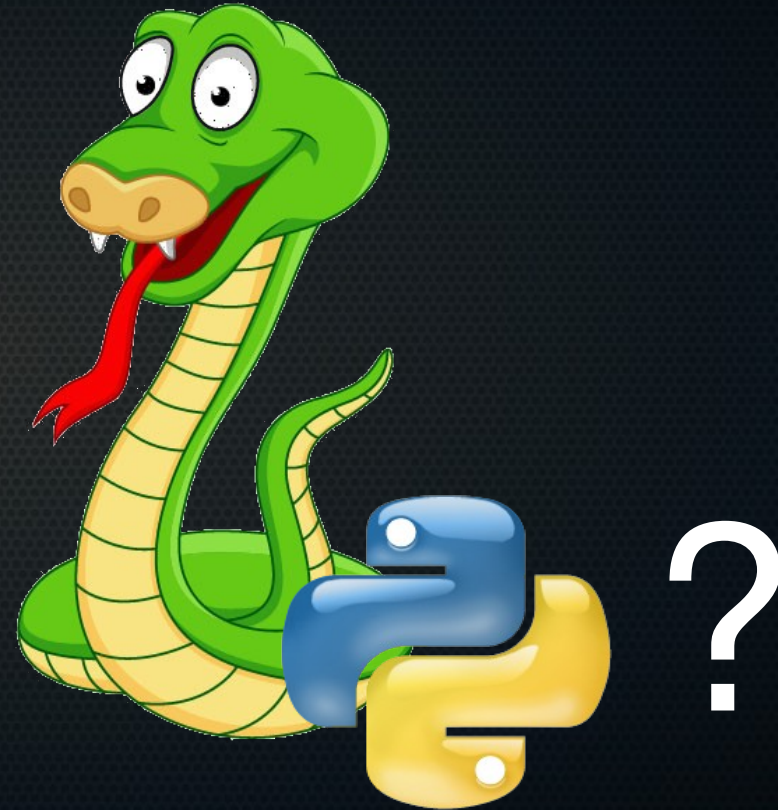
# ¿Qué es Android?

- Android es un sistema operativo basado en GNU/Linux el cual esta presente en la mayoría de terminales del mercado.
- En su interior esta escrito en Java, XML y C/C++ (el núcleo)
- Entre sus características principales cuenta con:
  - Navegador integrado basado en la tecnologia webkit
  - Almacena los datos en bases de datos SQLite
  - Sus gráficos estan basados en OpenGL
  - Soporta multimedia en general (imágenes, videos y audio)

# Desarrollar aplicaciones en Python, ¿En serio?



ó



# SL4A (Scripting Layer for Android)

- Permite a los desarrolladores editar, ejecutar scripts e interactuar con intérpretes en los dispositivos móviles.
- Es posible acceder a muchas de las API's del sistema operativo de forma cómoda.
- SL4A es open-source, se distribuye bajo licencia Apache 2.0 y soporta otros lenguajes aparte de python: Beanshell, JRuby, Lua, Perl y Rhino.

API Reference:

<http://code.google.com/p/android-scripting/wiki/ApiReference>

# SL4A y su intérprete

- Barcode y enlace para SL4A



[http://android-scripting.googlecode.com/files/sl4a\\_r4.apk](http://android-scripting.googlecode.com/files/sl4a_r4.apk)

- Barcode y enlace para el intérprete



[http://android-scripting.googlecode.com/files/PythonForAndroid\\_r4.apk](http://android-scripting.googlecode.com/files/PythonForAndroid_r4.apk)

# Entornos para un desarrollo cómodo

- ADB :
- Herramienta que viene junto con el SDK de android y que nos permite acceder y, por lo tanto, controlar un dispositivo android desde una PC.
- Emulador Android :
- Con este emulador se cargan y crean los AVD (Android Virtual Device), en los que se especifica el hardware y la versión del sistema operativo

<http://developer.android.com/sdk/index.html> ,

Enlace para descargar el Android SDK que nos proporciona ambas herramientas



# Distribuir nuestros scripts al público

- Generando un barcode :

Este método sirve tanto para desarrollar más cómodamente como para distribuir tu script a un público limitado (Desarrolladores)

Generador QR online: <http://zxing.appspot.com/generator/>

- Este es un buen método siempre y cuando tu público se dedique también a la programación.

# Generador de barcode (QR)

QR Code Generator from the ZXing Project

Contents

Text content 

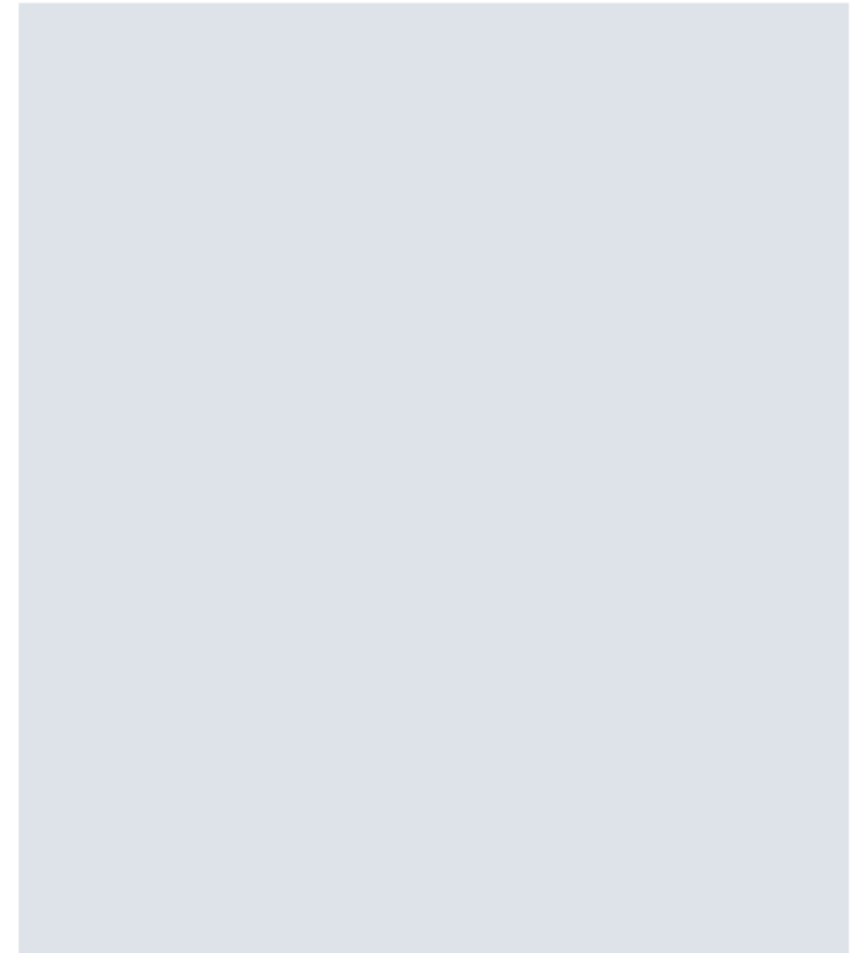
```
import android
droid = android.Android()
# y seguís escribiendo code
```

Barcode size

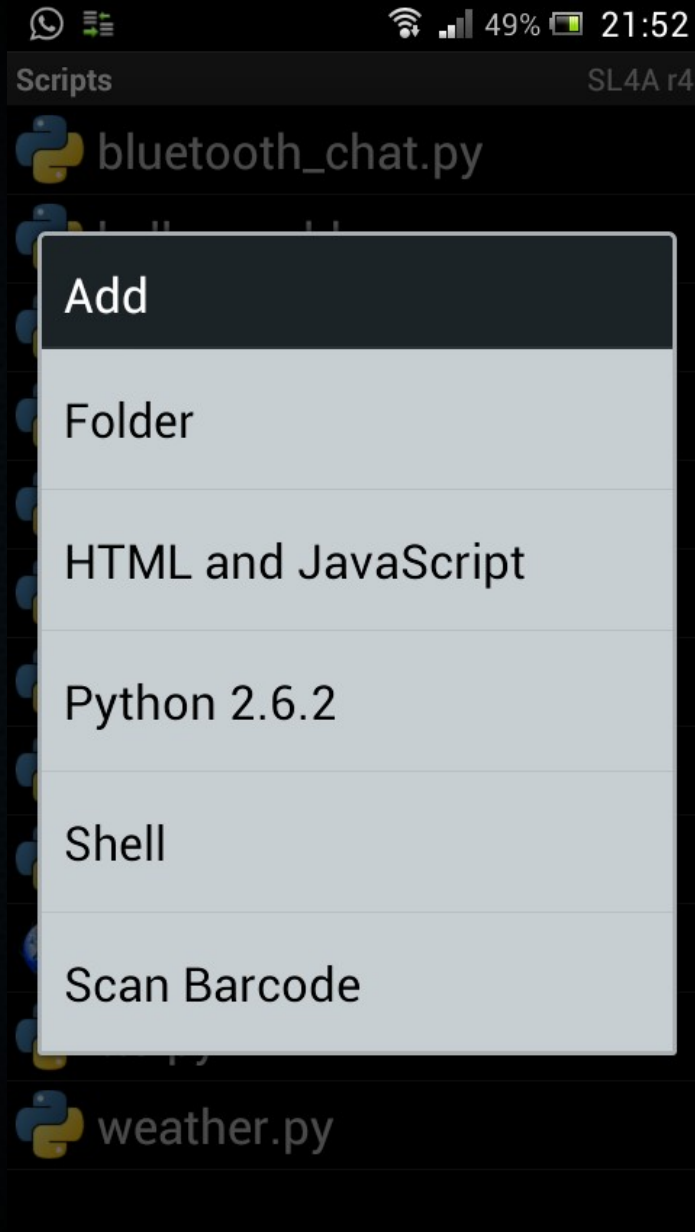
Error correction

Character encoding

Generate →



# Escanear barcode



Abrir SL4A >>

Opciones >>

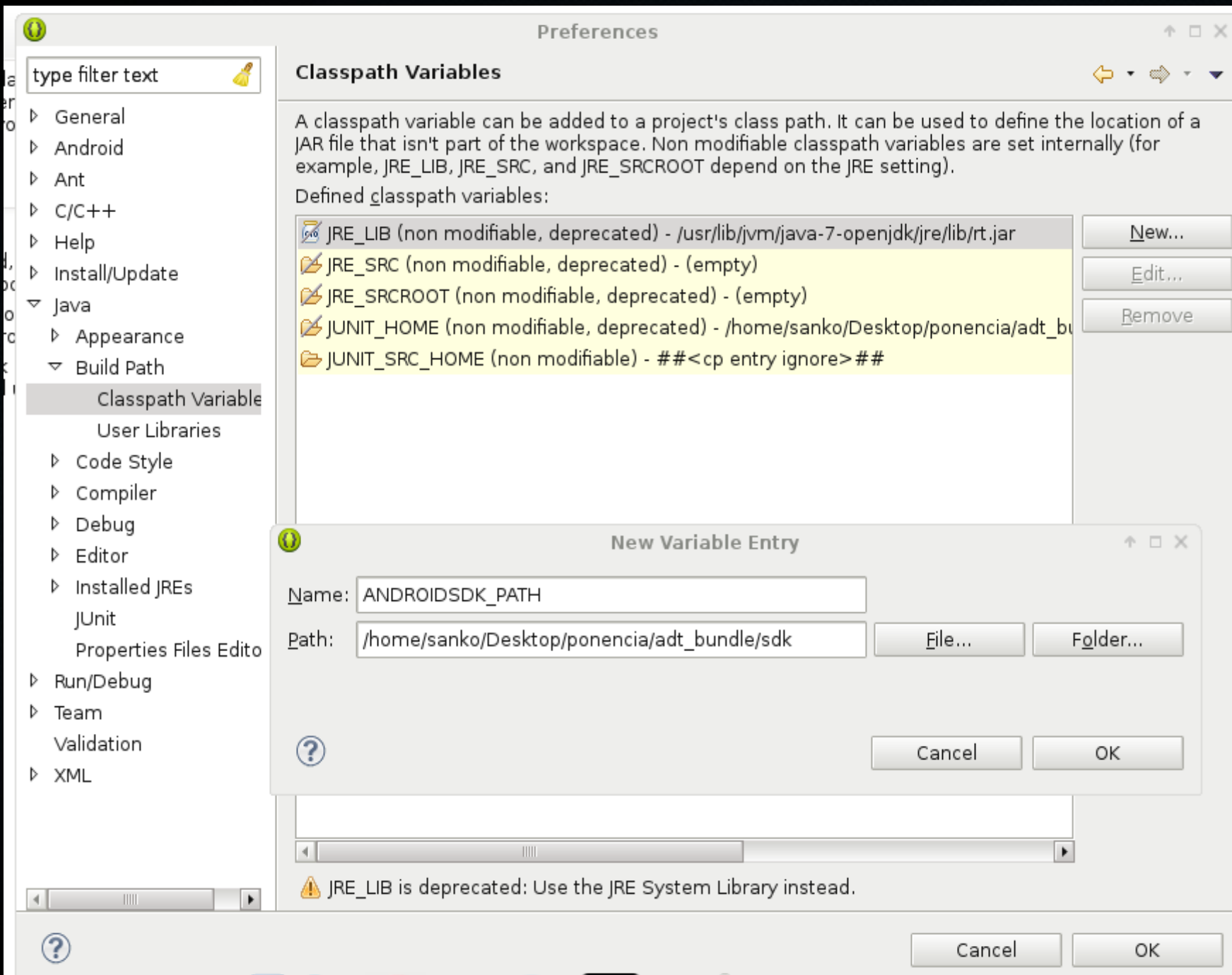
Add >>

ScanBarcode >>

El script se añadira al  
directorio <Scripts>

# Empaquetando en .apk

- Este proceso esta aun en una fase experimental, por lo que la distribución de aplicaciones a un mercado es muy limitada por el momento.
- Requiere eclipse para el empaquetamiento.
- Descargar el template:  
[https://github.com/damonkohler/sl4a/raw/master/android/script\\_for\\_android\\_template.zip](https://github.com/damonkohler/sl4a/raw/master/android/script_for_android_template.zip)



Window >>

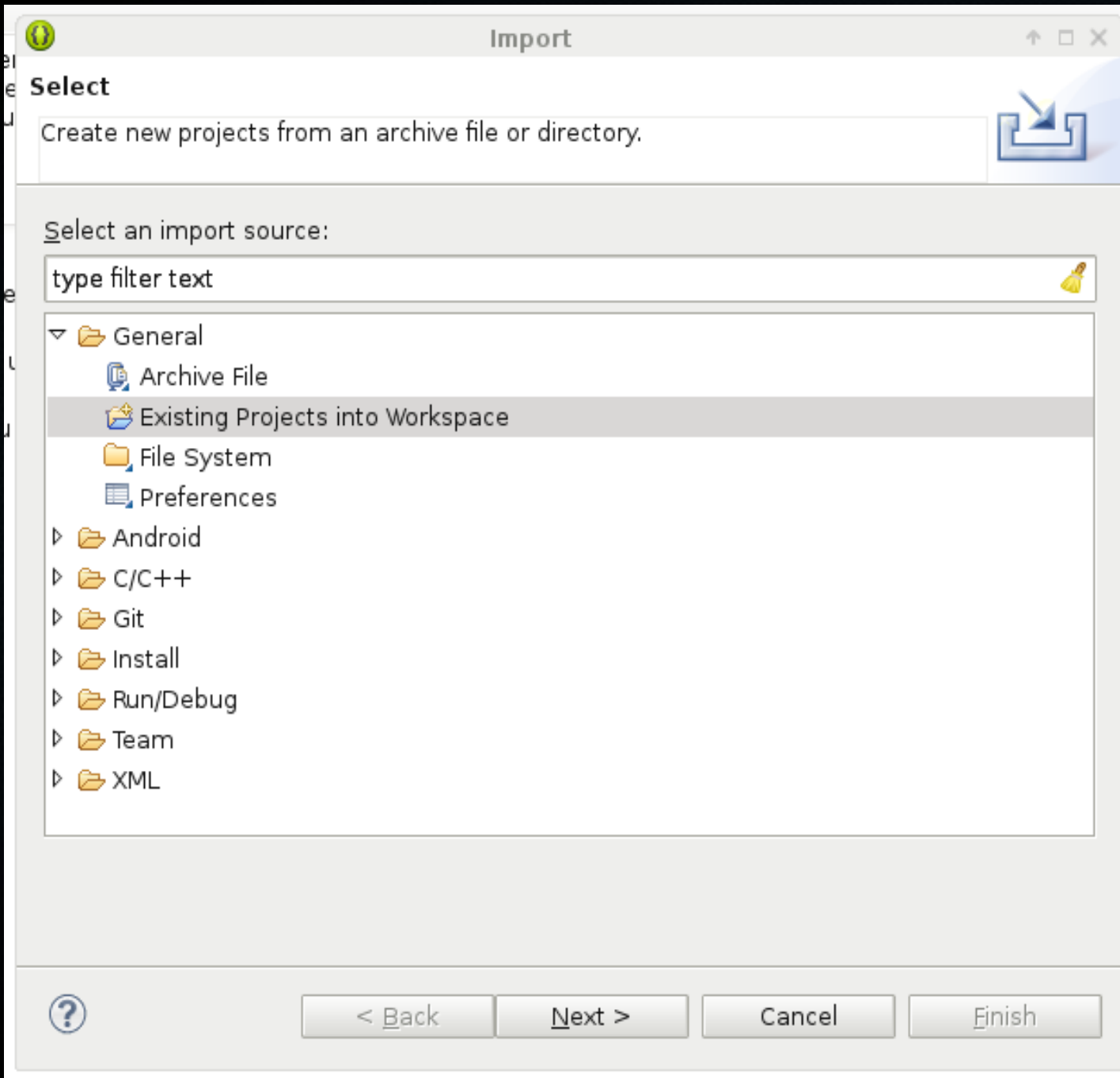
Preferences >>

Build Path >>

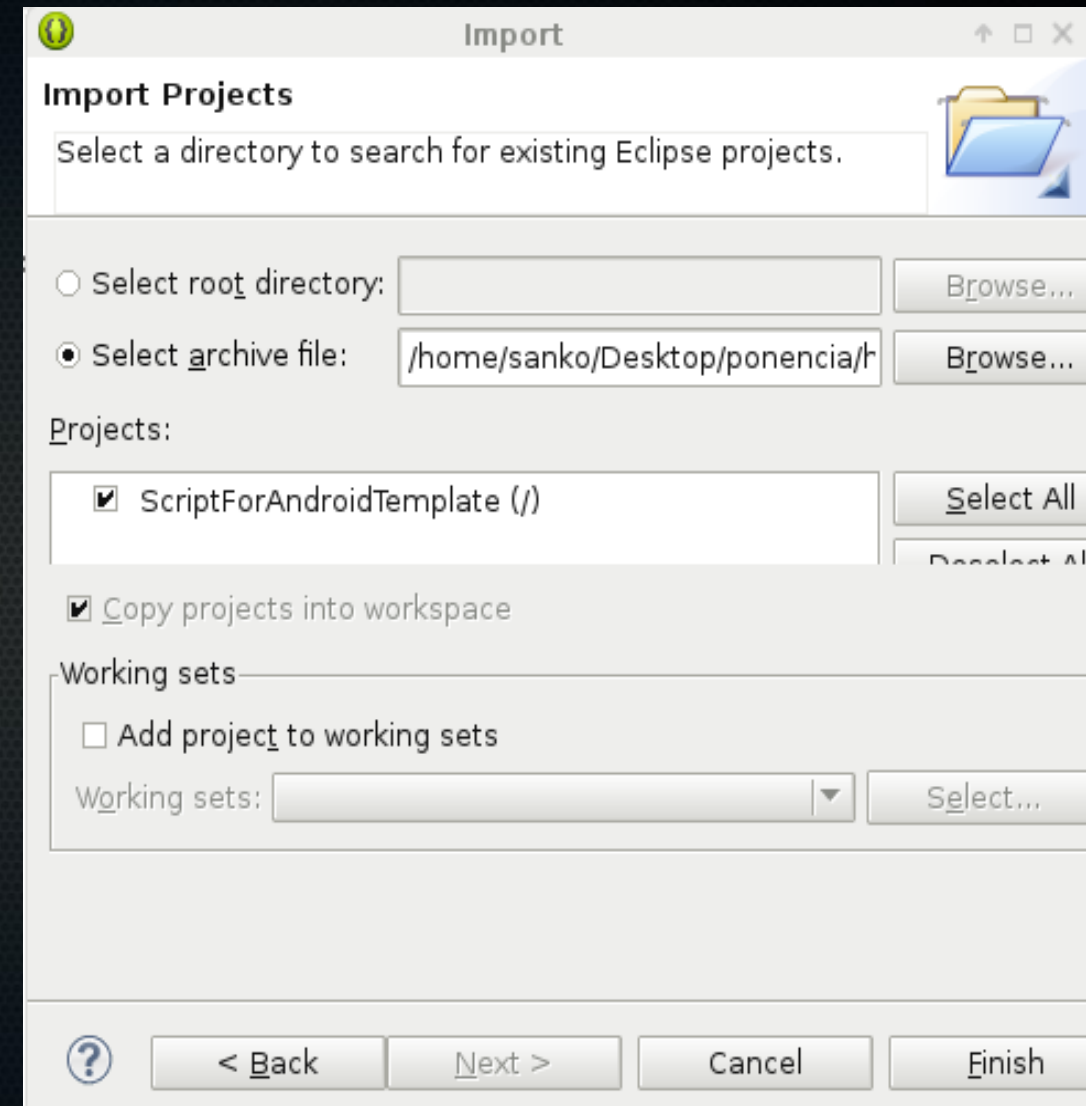
Classpath Variable  
>>

Click en New >>  
Indicas la ruta del  
SDK

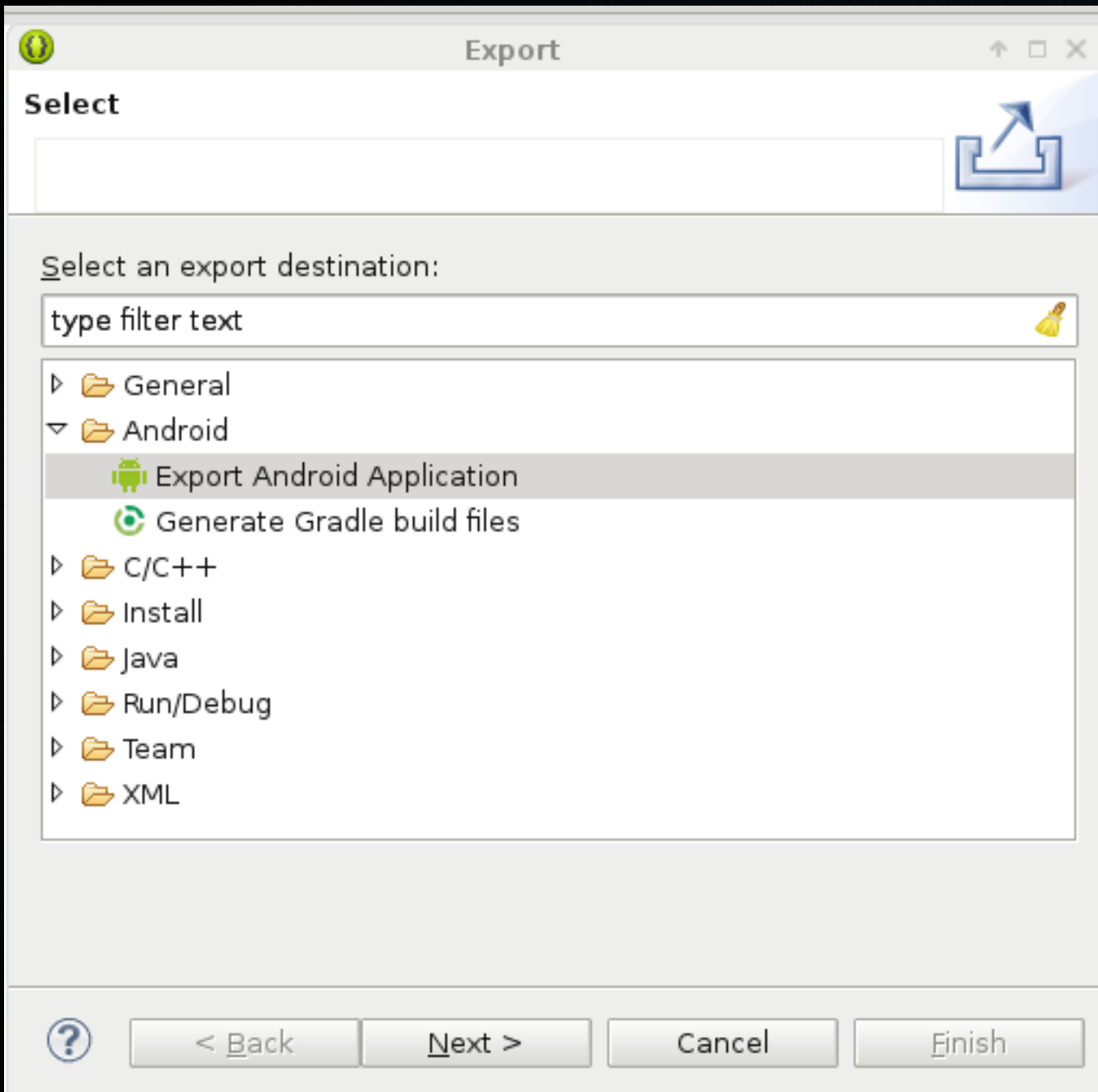
Click en OK



Click en Existing Projects...



Click en finish...



Por último :

File >> Export >>

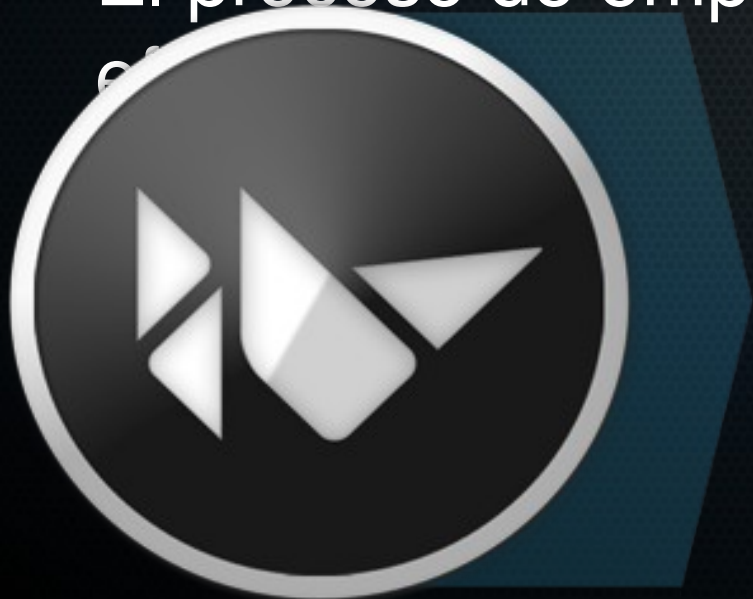
Export Android App >>

Click en Next >>

Seleccionas el proyecto y si eres afortunado y no tienes errores, tendrás tu .apk listo para correr

# Otras formas de desarrollo, kivy

- Kivy es un framework/toolkit con núcleo python que nos permite desarrollar aplicaciones multi-touch para distintas plataformas (iOS, Windows, Android...)
- <http://kivy.org/docs/gettingstarted/intro.html>
- El proceso de empaquetamiento parece ser más cómodo y





# Referencias

- <http://revista.python.org.ar/5/es/html/desarrollo-de-aplicaciones-moviles-para-android-con-python.html>
- <http://code.google.com/p/android-scripting/>
- <http://code.google.com/p/android-python27/wiki/TutorialHowToInstallAndroidSDKAndEclipse>  
(MUY ÚTIL PARA EL PROCESO DE PACKAGING)
- <http://kivy.org/docs/gettingstarted/intro.html>

# FIN DE LA PRESENTACIÓN

- FIN DE LA PRESENTACIÓN, AUNQUE NO DE LA PYCON, DISFRUTAD QUE LA CON ES CORTA!