

TP1

Développement d'applications mobiles

Installation et configuration de Android Studio



Prérequis

- Architecture d'Android OS



Objectifs du cours

- Installer et configurer Android Studio
- Se familiariser avec l'IDE Android Studio
- Comprendre le cycle de développement d'une application
- Créer et exécuter une application Android

Outils utilisés dans ce module

- Android Studio v3.1.4



- JDK 8u181



- Android SDK - API 28



- NOX v6.2.2.0



Besoins en matériel

- **2 Go de mémoire RAM**
+ 8 Go de préférence
- **+ 10 Go d'espace disque** pour tout installer
- **Processeur puissant**
La puissance pure qui compte le plus rapide sera le mieux
- **1 cœur de votre processeur** est suffisant

Outil de développement

Android Studio v3.1.4 (Août 2018)

- Adapté à la programmation Android
- Gourmand en ressource
- Complétion de code avancée



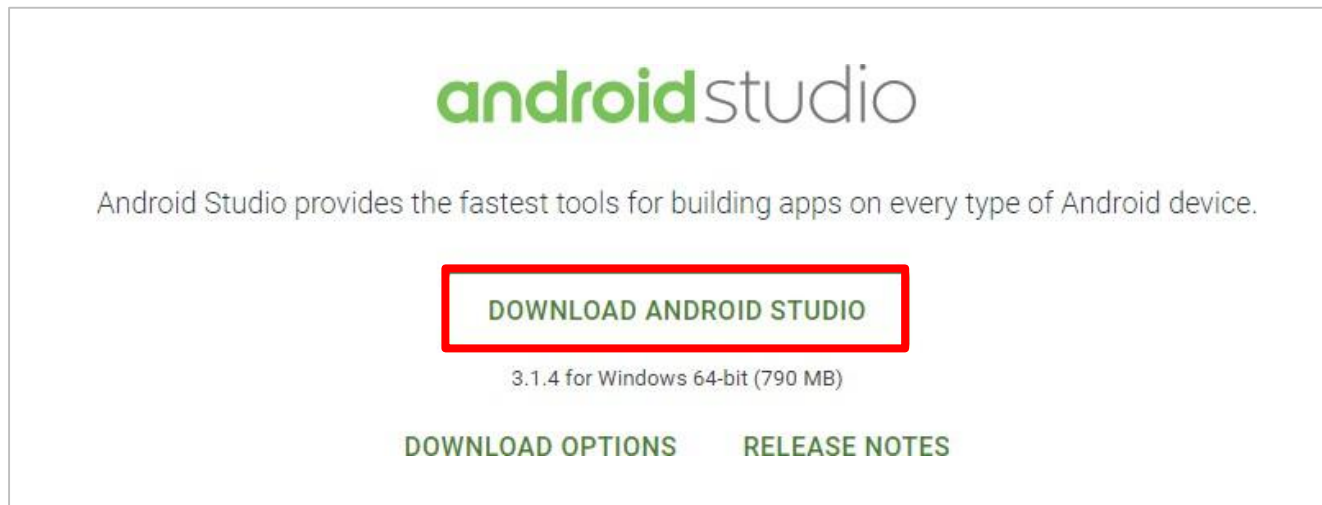
Installer Android Studio

1. Installer le JDK (Java SE 8u181)

- **Lien :** www.oracle.com/technetwork/java/javase/downloads/index.html

2. Installer Android Studio 3.1.4

- **Lien :** <https://developer.android.com/studio/>
- **Chemin :** D:\android\android-studio



Installer Android Studio

1. Récupérer le fichier sdk.zip

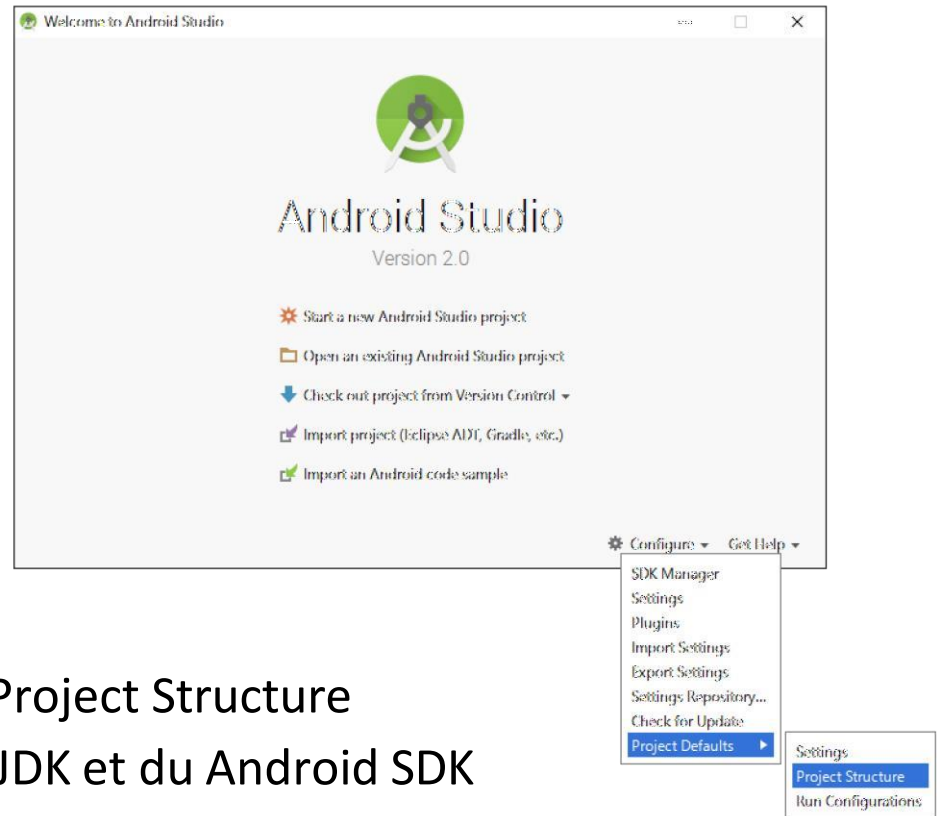
- fourni dans un flash disk

2. Décompressez sdk.zip dans

- D:\ android\android-sdk

3. Configurer Android studio

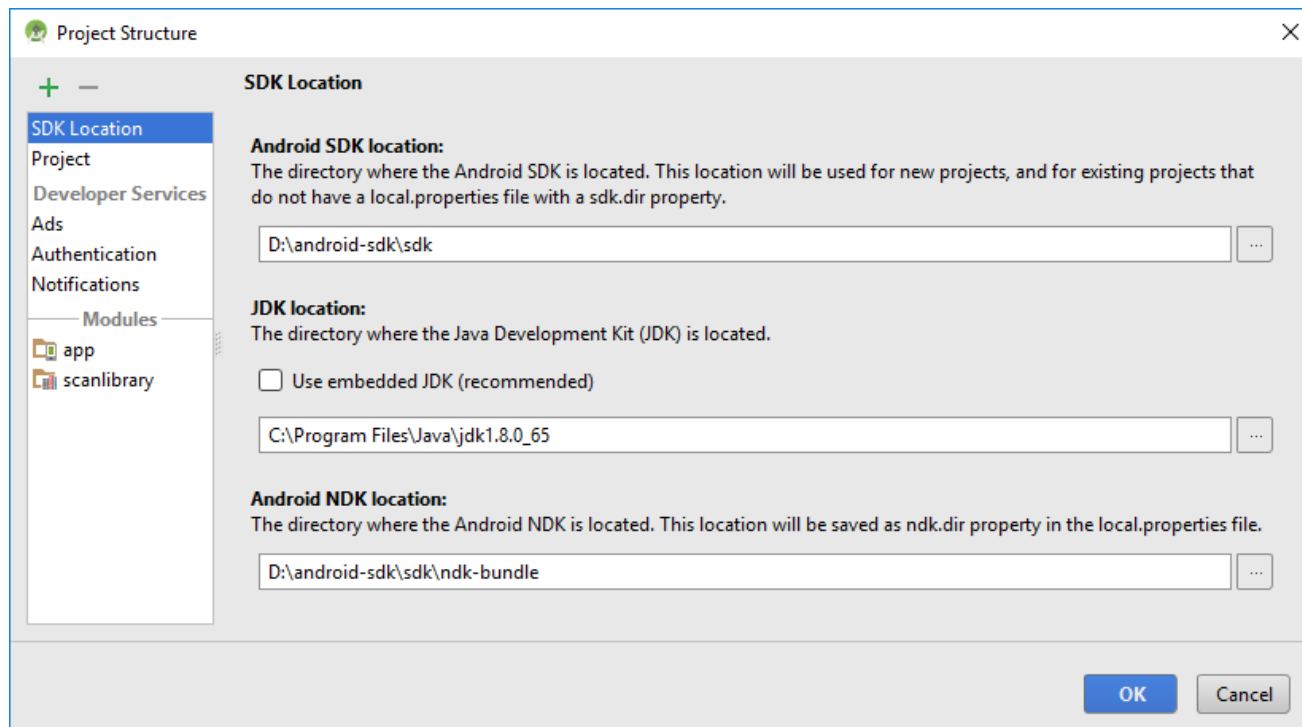
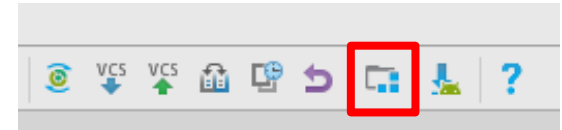
- Ouvrir Android Studio >
Configure > Project Defaults > Project Structure
- Indiquer le chemin d'accès du JDK et du Android SDK



Définir le chemin d'accès du JDK/SDK

Sous Android Studio :

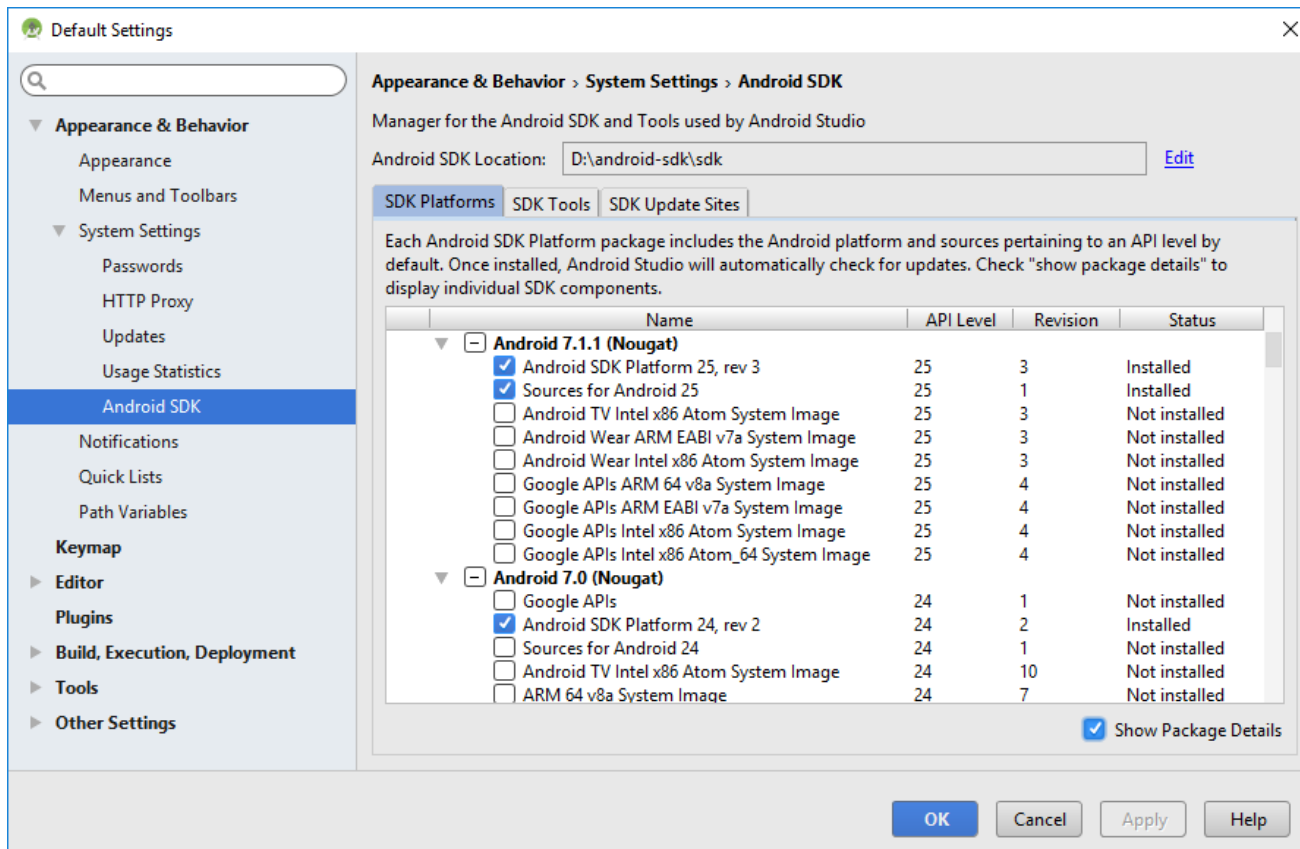
- **File > Project Structure... > SDK Location**



SDK Manager

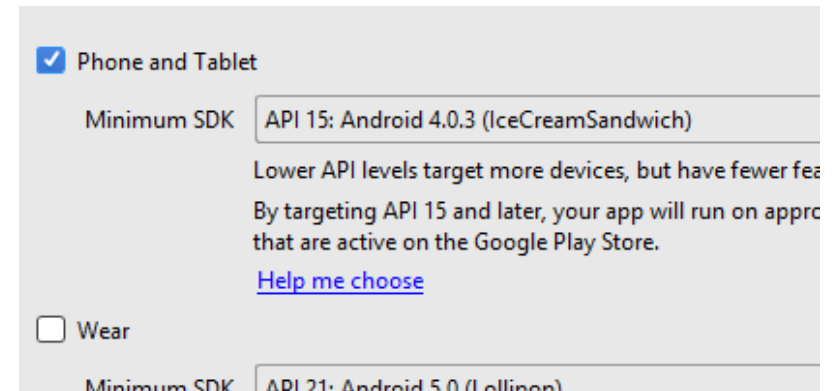
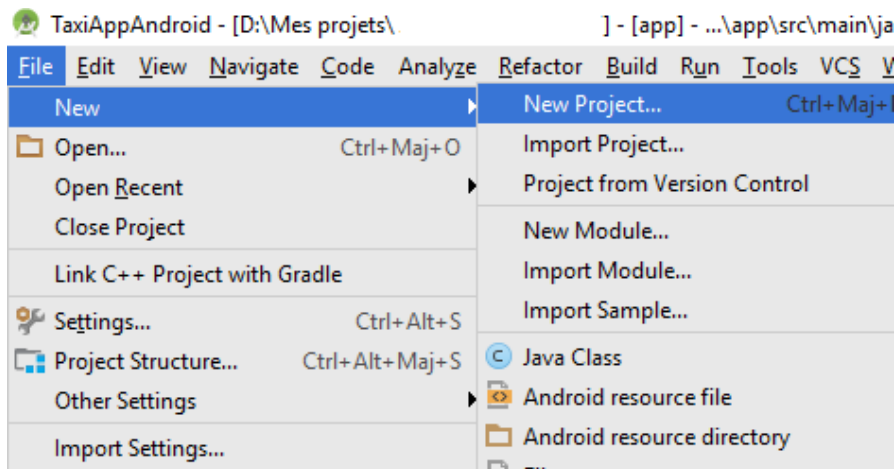
Sous Android Studio :

- Tools > Android > SDK Manager

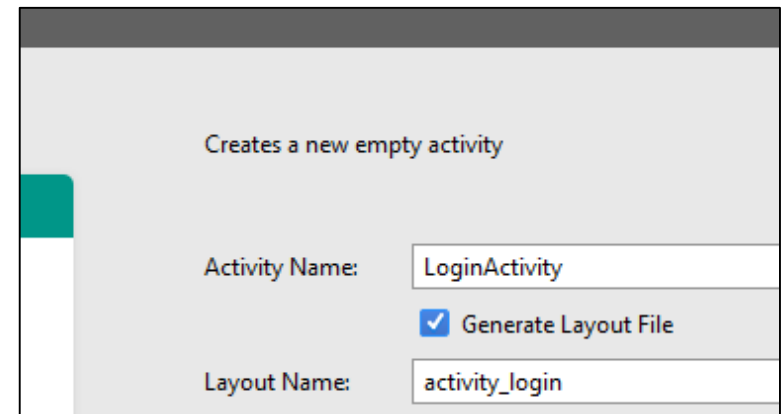
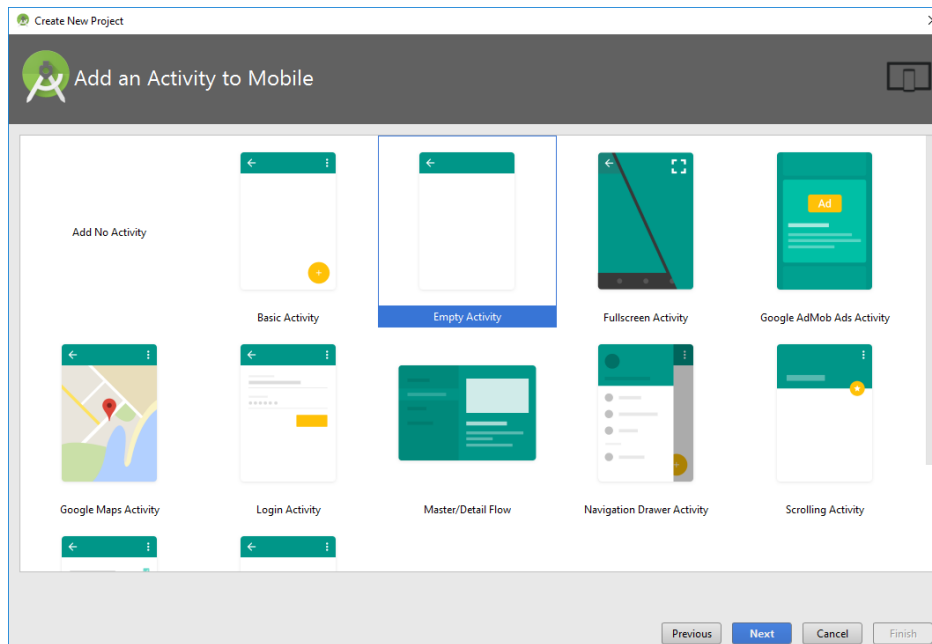


Création d'un nouveau projet (1/2)

- Application Name : **LearnApp**
- Compagny Domain : **iam.sci**
Package: **sci.iam.learnapp**
- Project location : **D:\android\projects**

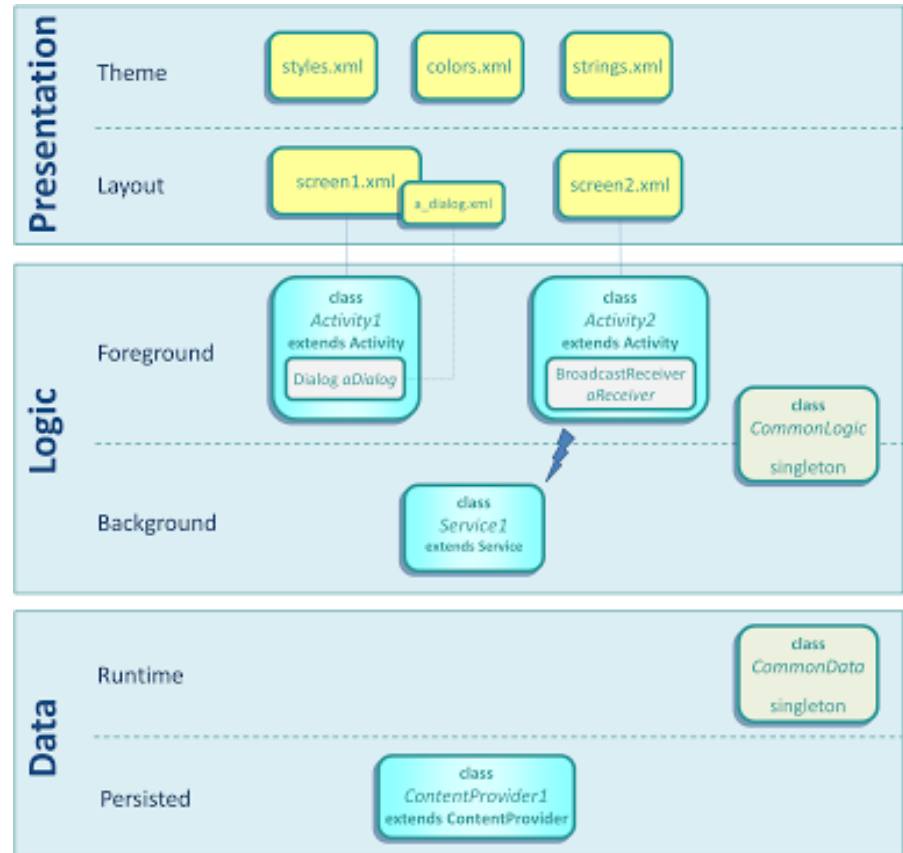


Création d'un nouveau projet (2/2)

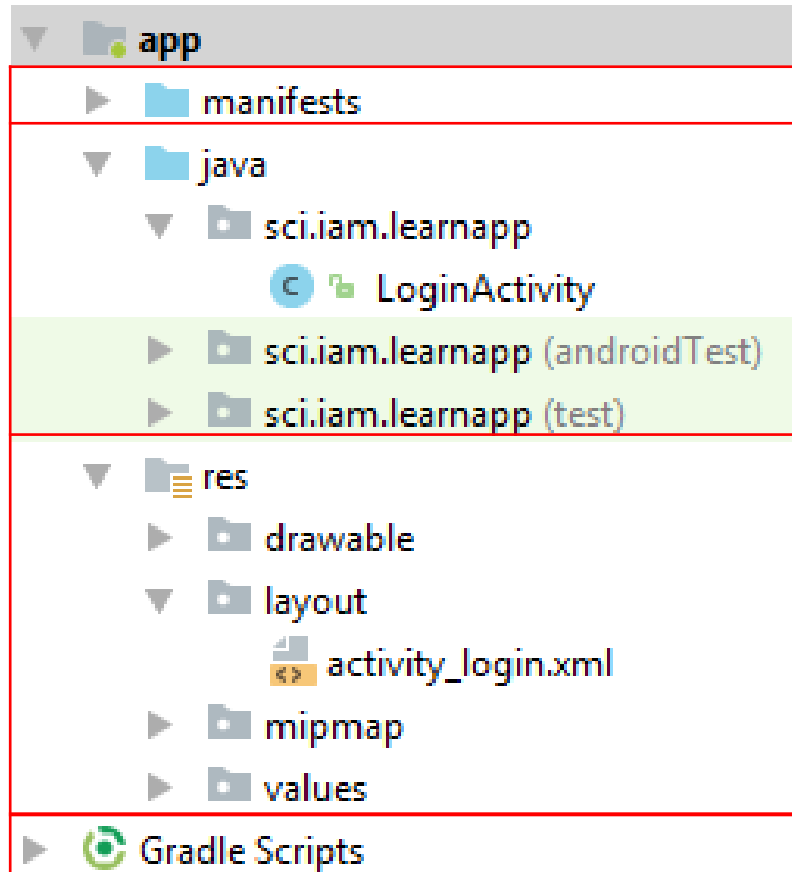


Structure d'un projet (1/2)

- Un projet Android possède une structure semblable à **MVC**, en séparant la vue du contrôle
- Cette structure permet d'organiser le projet en 3 couches :
 - **Présentation**
 - **Logique**
 - **Données**



Structure d'un projet (2/2)



Manifest

Fichiers java

Ressources organisées
(texte, ...)

Gradle

Processus de développement (1/2)

Compilation du code source

- Ecrit en Java et compilé en Dalvik bytecode (.dex)

Archivage (Packaging)

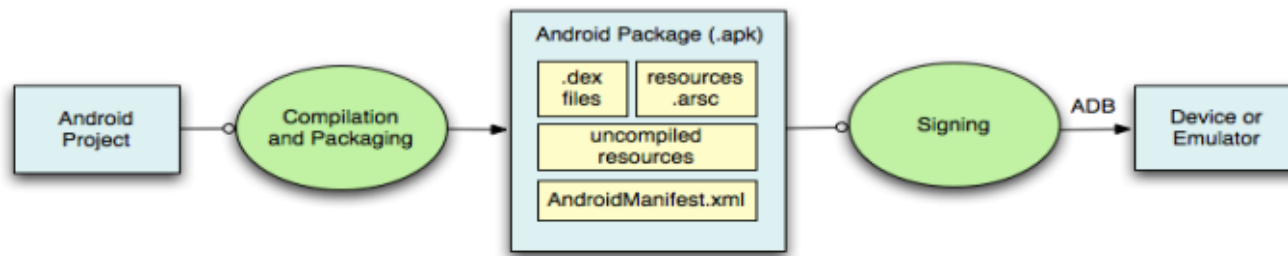
- Archive .apk

Signature

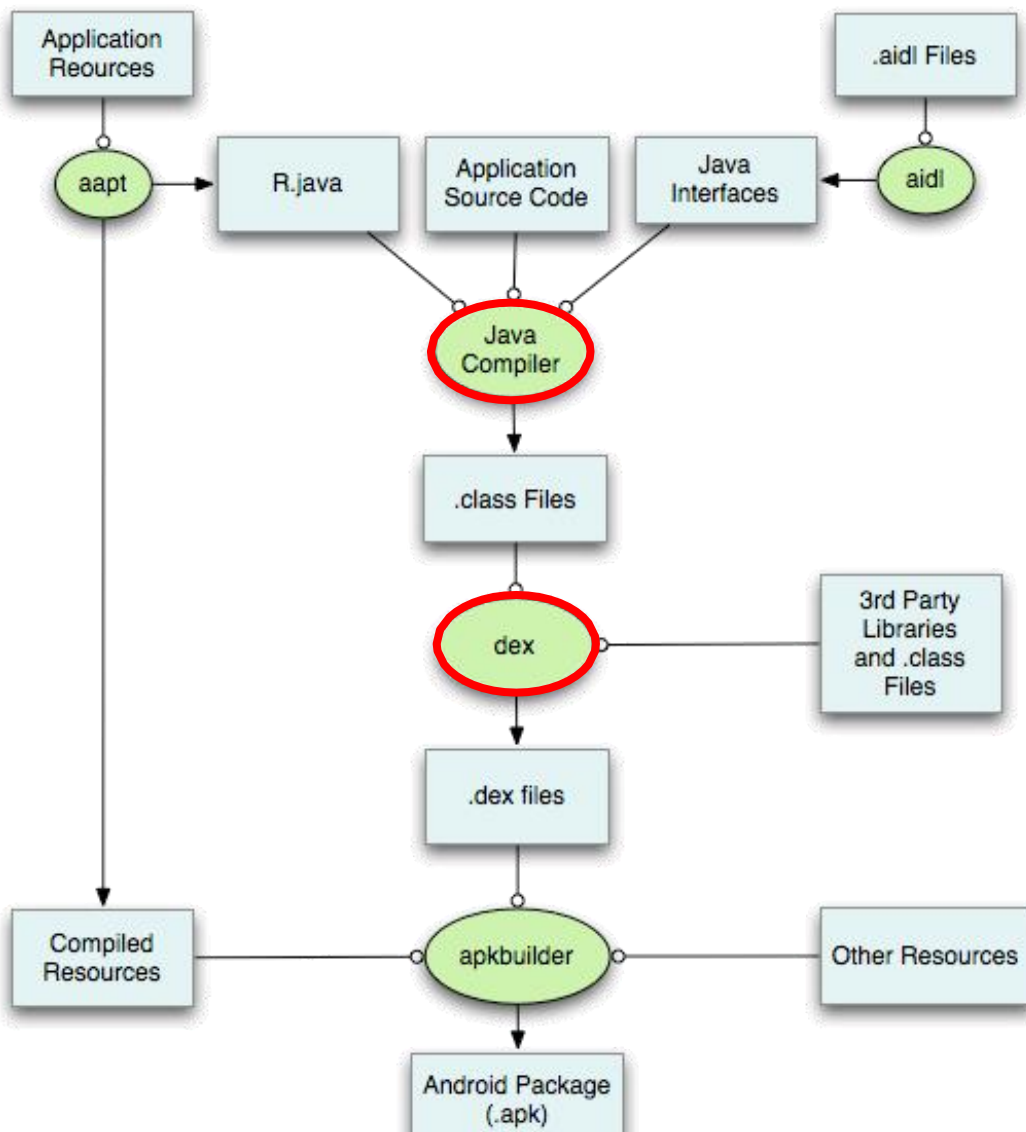
- Les applications doivent être signées pour être déployées (pas nécessaire en mode développement)

Exécution

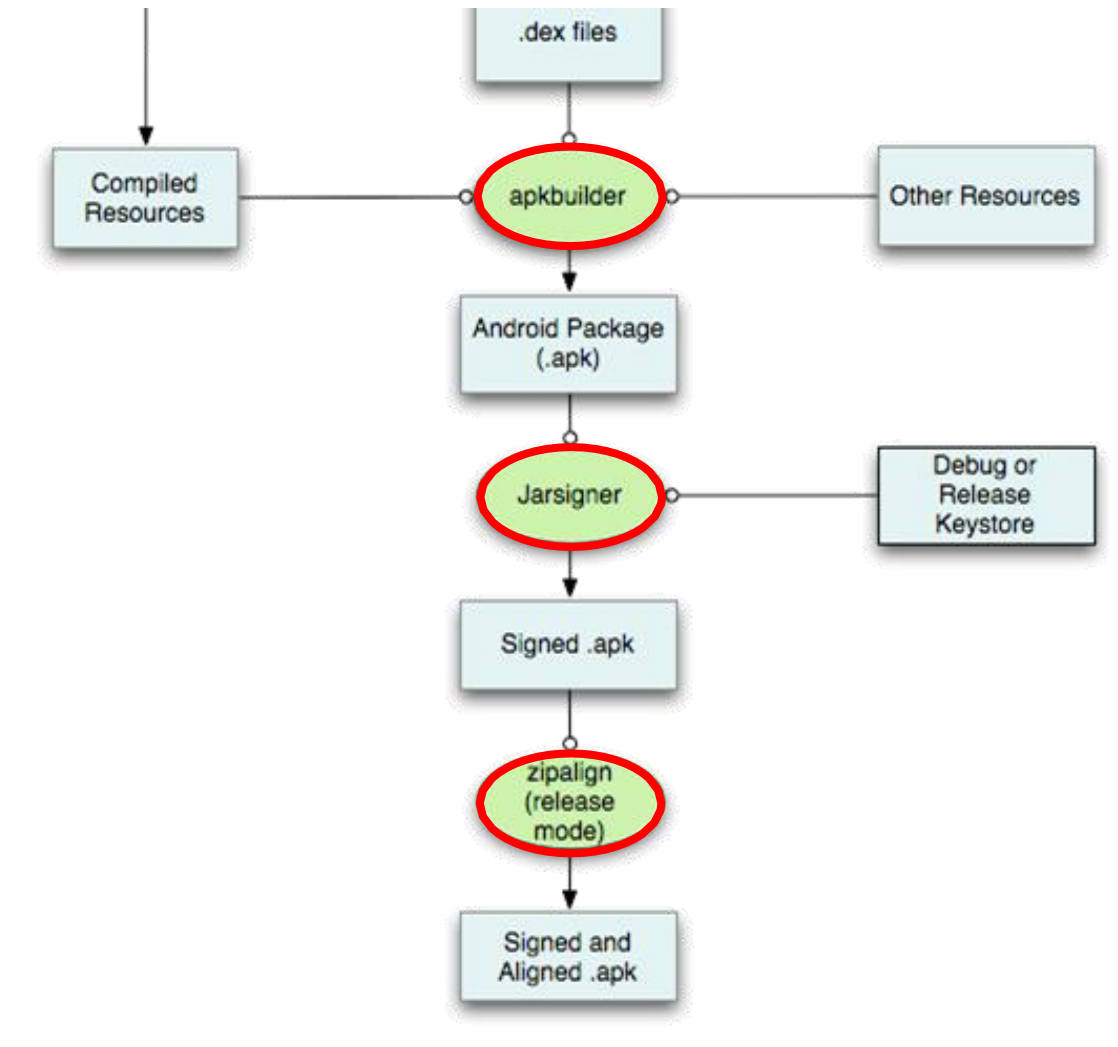
- Sur un support mobile physique (smartphone par exemple)
- Sur un émulateur



Processus de développement (2/2)



Processus de développement (2/2)



Exécution sur l'émulateur intégré (1/4)

AVD (Android Virtual Device)

- AVD Manager
- Lent au démarrage et à l'usage
- Gourmant en ressources
- Emulation vs. simulation

Fonctionnalités non-disponibles

- Appels téléphoniques réels
- Capteurs en général
- Connexions USB
- Évolution de la charge de la batterie
- ...



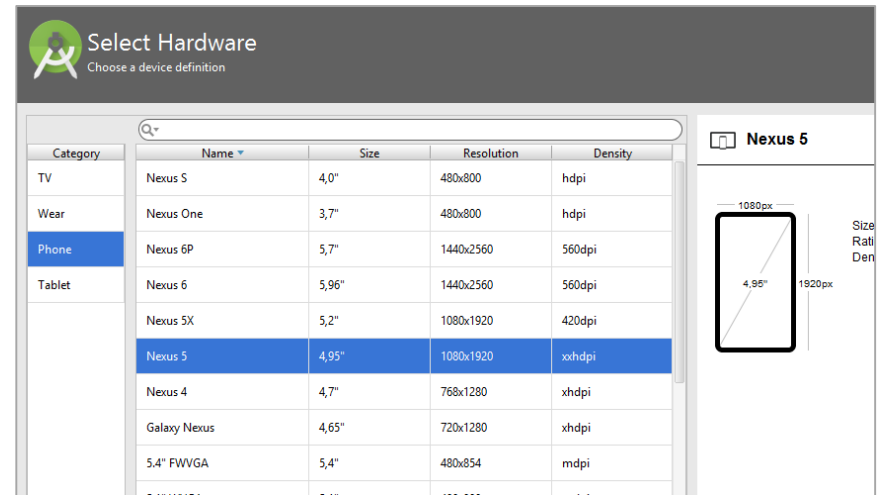
Exécution sur l'émulateur intégré (2/4)

Création d'un nouvel émulateur fourni par le SDK

1. Vérifier l'image de l'émulateur

- Ouvrir le SDK Manager
- Android 8.0 > Intel x86 Atom System Image > **installer le package (si nécessaire)**

2. Créer un AVD



Exécution sur l'émulateur intégré (3/4)

Création d'un nouvel émulateur fourni par le SDK

System Image
Select a system image

Release Name	API Level	ABI	Target
Marshmallow	23	x86	Android 6.0 (with Google APIs)
Marshmallow	23	armeabi-v7a	Android 6.0
Marshmallow	23	x86	Android 6.0
Marshmallow	23	x86_64	Android 6.0

Marshmallow

 API Level
23
Android
6.0
Android Open Source Project
System Image
armeabi-v7a

Recommendation
Consider using a system image with Google APIs to enable testing with Google Play Services.

Questions on API level?
See the [API level distribution chart](#)

☐ Show downloadable system images

Android Virtual Device (AVD)
Verify Configuration

Front: Back:

Speed: Latency:

☒ Use Host GPU
☐ Store a snapshot for faster startup
You can either use Host GPU or Snapshots

RAM: MB
VM heap: MB
Internal Storage: MB
SD card: ☒ Studio-managed MB

Built-in SD Card Size

The amount of removable space available to store data on the device. We recommend at least 100MB in order to use the camera in the emulator.

Recommendation
Consider using a system image with Google APIs to enable testing with Google Play Services.

Hide Advanced Settings

Exécution sur l'émulateur intégré (4/4)

Accélérer l'émulateur sous Intel

1. Activer la virtualisation dans le BIOS

- Redémarrer le PC et accéder au bios
- **System Configuration > VT-x > Enable**

2. Télécharger l'accélérateur Intel HAXM

- Ouvrir le SDK Manager
- Installer le package **Intel x86 Emulator Accelerator (HAXM installer)**

3. Installer l'accélérateur Intel HAXM

- Aller à
[Android_SDK]\extras\intel\Hardware_Accelerated_Execution_Manager
- Installer **intelhaxm-android.exe**

Exécution sur l'émulateur externe

- Rapide en lancement
- Moins gourmand en mémoire

NOX App Player v5.2.2.0 (6 août 2018)

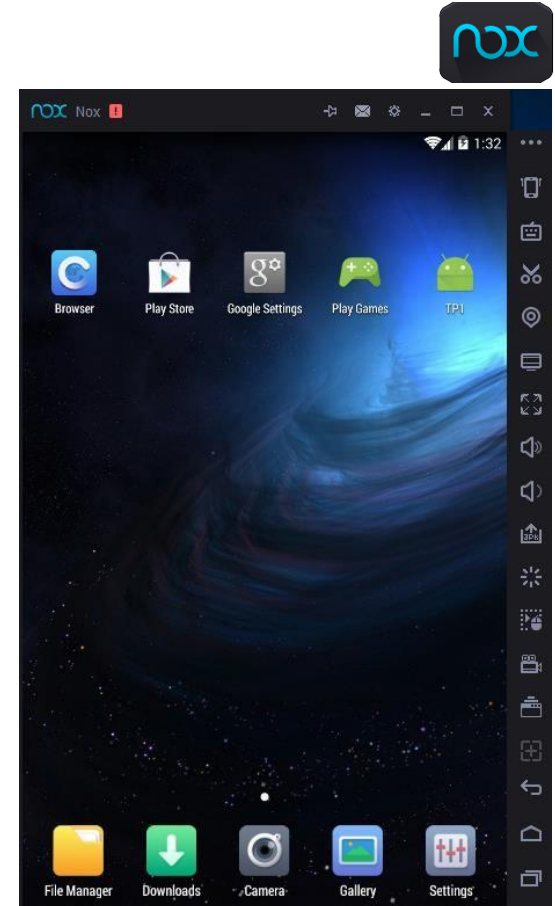


- Gratuit
- Lien : <http://fr.bignox.com>
- Version Android : 4.4.2

GenyMotion v2.12.1 (9 mai 2018)



- Payant
- Lien : <https://www.genymotion.com/>
- Multi-version
- Fonctionne sous Oracle VM VirtualBox



Exécution sur un appareil mobile (1/5)

Fonctionnalités de base

- Capteurs et composants intégrés (Vibreux, batterie, bluetooth, ...)

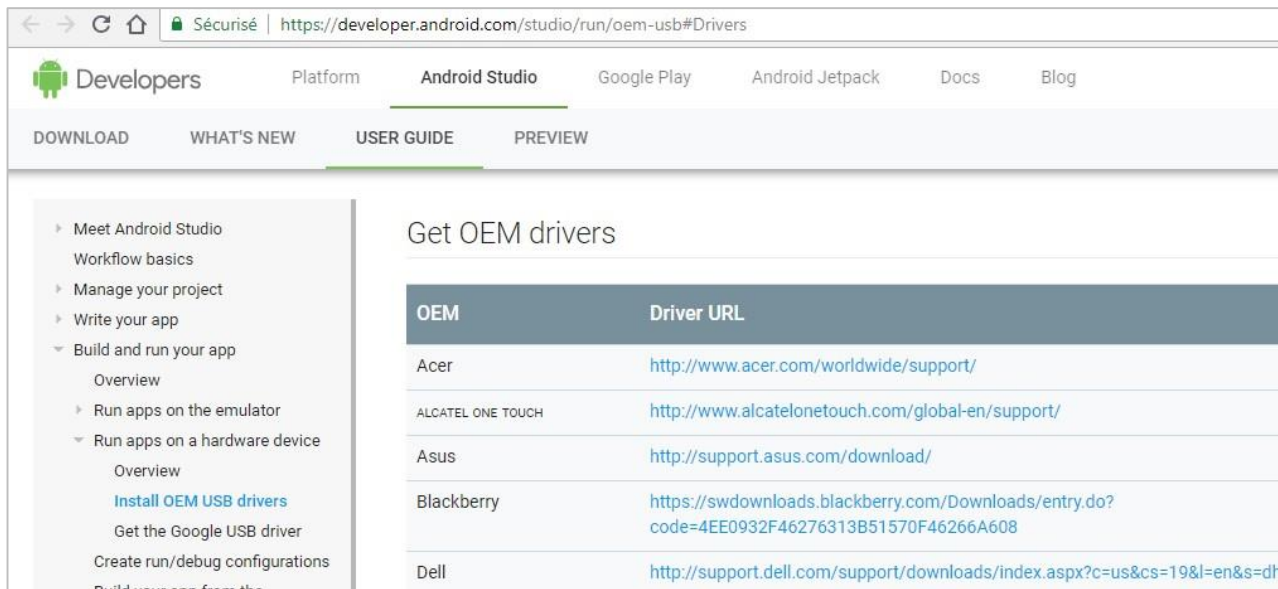
Caractéristiques d'un appareil mobile

- Un processeur à puissance limitée
- Une RAM et des capacités de stockage limitées
- Un écran avec une résolution fixe
- Des coûts élevés de transfert de données
- Une batterie à autonomie limitée

Exécution sur un appareil mobile (2/5)

Etape 1 : Installer le driver de l'appareil

1. Aller à : <http://developer.android.com/tools/extras/oem-usb.html#Drivers>
2. Choisir le constructeur de l'appareil et cliquer sur le lien correspondant
3. Suivre les instructions pour installer le driver approprié



The screenshot shows the 'Developer' section of the Android Studio website. The 'USER GUIDE' tab is selected, and the 'Get OEM drivers' section is displayed. A table lists various OEMs and their corresponding driver URLs.

OEM	Driver URL
Acer	http://www.acer.com/worldwide/support/
ALCATEL ONE TOUCH	http://www.alcatelonetouch.com/global-en/support/
Asus	http://support.asus.com/download/
Blackberry	https://swdownloads.blackberry.com/Downloads/entry.do?code=4EE0932F46276313B51570F46266A608
Dell	http://support.dell.com/support/downloads/index.aspx?c=us&cs=19&l=en&s=dh

Exécution sur un appareil mobile (3/5)

Etape 1 : Installer le driver de l'appareil

Installer le driver pour les appareils Google (ex : Nexus 5)

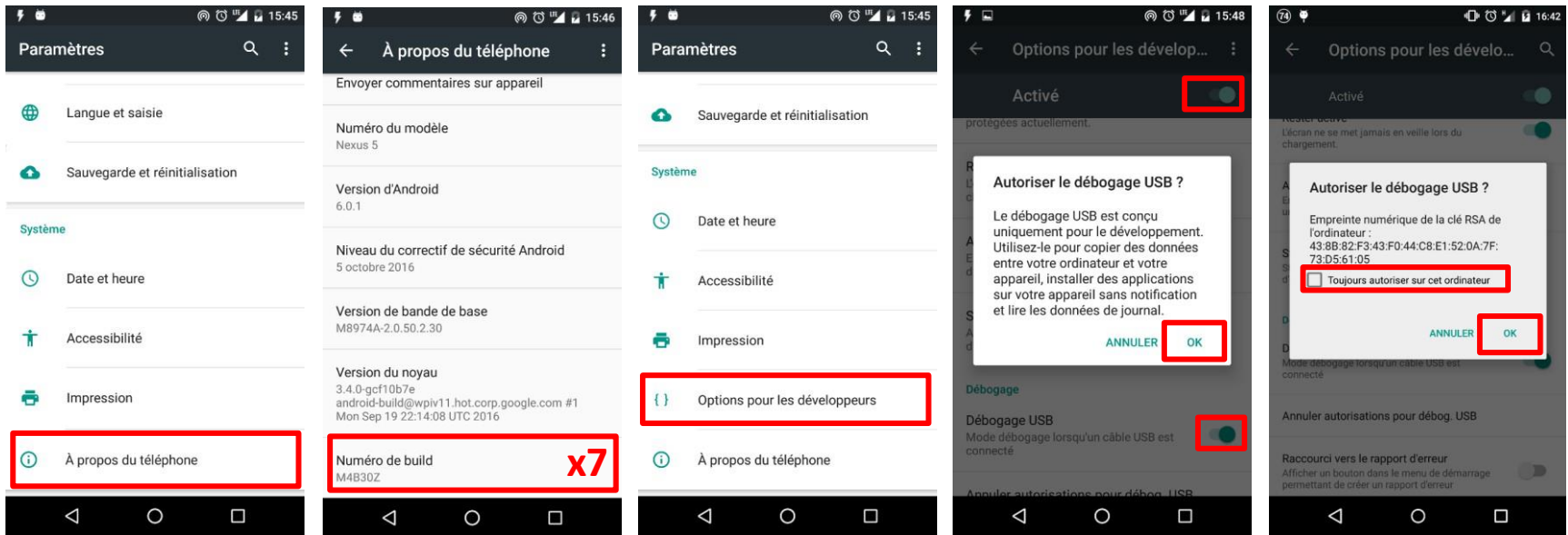
1. Ouvrir le SDK Manager
2. Extras > Google USB Driver > Installer le package
3. Aller à [CHEMIN_DU_SDK]\extras\google\usb_driver
4. `android_winusb.inf`  > Installer les drivers

Installer le driver pour les appareils Samsung (ex : Galaxy S6)

1. Aller à <http://www.samsung.com/us/support/downloads>
2. Choisir le modèle de l'appareil > Get Downloads
3. Downloads > USB (Driver) ver.x.x.xx.x > Download (EXE)

Exécution sur un appareil mobile (4/5)

Etape 2 : Activer le débogage USB dans l'appareil

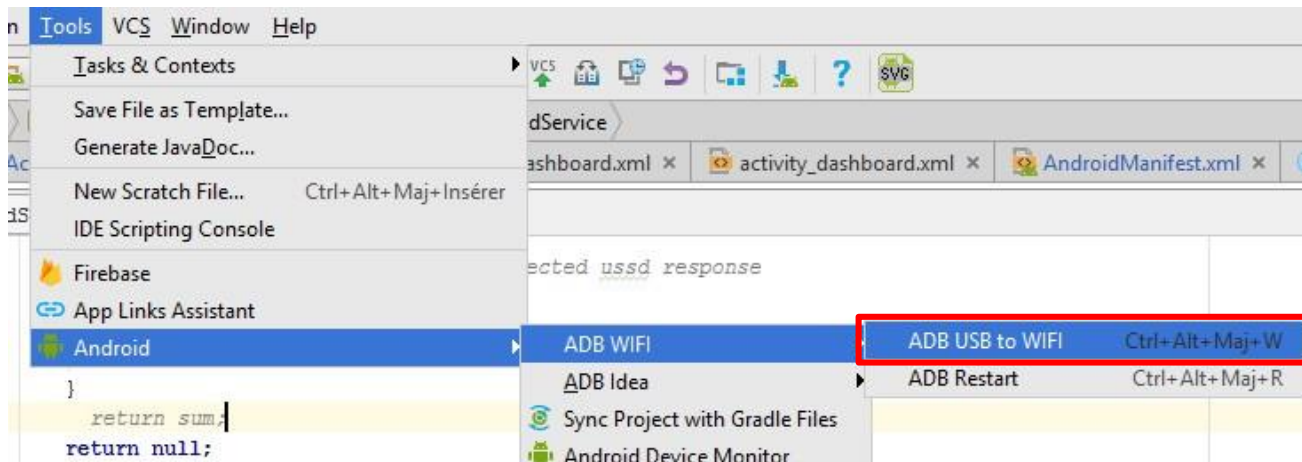


Exécution sur un appareil mobile (5/5)

Débogage sans-fil : Exécution sur un appareil via le Wi-Fi

Débogage sans-fil : Exécution sur un appareil via le Wi-Fi

1. **Installer le plugin ADB WIFI :**
 - File > Settings > Plugins > Browse repositories > ADB WIFI > Installer
2. **Connecter l'appareil au PC par câble USB et au même réseau Wi-Fi du PC**
3. **Activer le plugin :**
 - Tools > Android > ADB WIFI > ADB USB to WIFI



Types d'applications sous Android

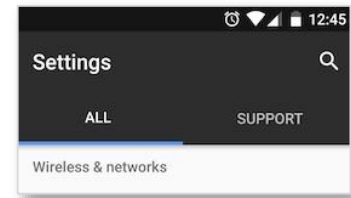
Applications de premier plan

- Utilisable quand elle est visible et mis en pause lorsqu'elle ne l'est pas
- **Exemples** : Jeux, Apps de cartes géographiques



Applications d'arrière-plan

- Interaction limitée en dehors de sa configuration
- **Exemples** : Alarme, Filtrage d'appels, réponse automatique aux SMS, ...



Widgets

- Composant visuel interactif que les utilisateurs peuvent ajouter à leur écran d'accueil
- **Exemples** : Affichage de l'heure, prévisions météo

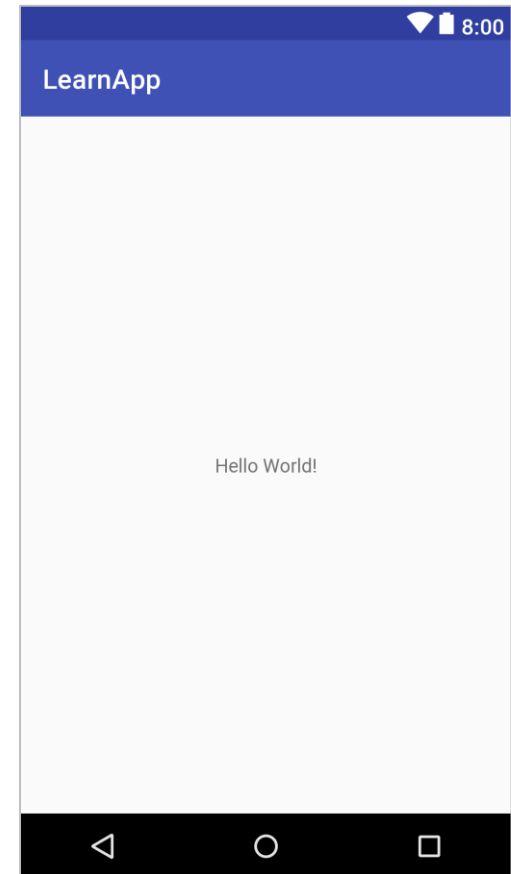
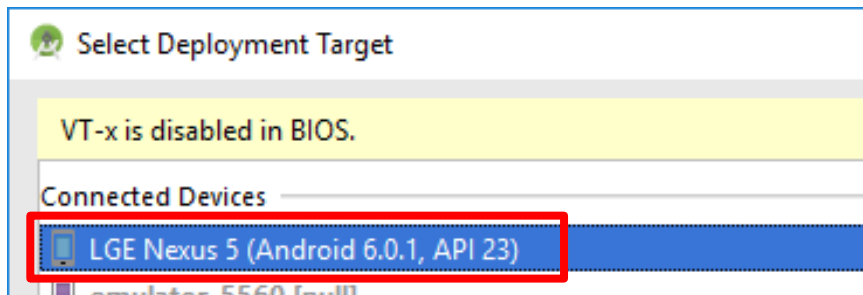


TP0 : Exécution de l'application

1. Lancer l'exécution



2. Sélectionner l'appareil d'exécution



Quelques liens utiles

Configuration avec Gradle

- <https://rominirani.com/gradle-tutorial-part-6-android-studio-gradle-c828c5639bb>

Processus de développement

- <https://github.com/dogriffiths/HeadFirstAndroid/wiki/How-Android-Apps-are-Built-and-Run>

Top 5 des émulateurs Android

- <http://appinformers.com/android-emulators-pc/13028/>

Références

- «Installing the Eclipse Plugin,» Massachusetts Institute of Technology, 2017. Lien : stuff.mit.edu/afs/sipb/project/android/docs/sdk/installing/installing-adt.html
- developer.android.com, «Android Studio - The Official IDE for Android,» Septembre 2017. Lien : <https://developer.android.com/studio/index.html>.
- R. Dima, «Tutoriel Android sous Android Studio,» 6 Février 2017. Lien : <https://hal.archives-ouvertes.fr/cel-01241650v2/document>
- D. Griffiths, «How Android Apps are Built and Run,» 28 Janvier 2017. Lien : <https://github.com/dogriffiths/HeadFirstAndroid/wiki/How-Android-Apps-are-Built-and-Run>.
- Nox Digital Entertainment Co., «L'émulateur Android parfait pour jouer aux jeux mobiles sur PC,» 2017. Lien : <https://fr.bignox.com/>.
- Genymobile, «Make better apps : Over 3000 Android configurations,» 2016. Lien : <http://www.genymotion.com/>.
- S. Wichai, «ADB WIFI : Android Studio plugin for debug android app over Wi-Fi,» 6 Mai 2016. Lien : plugins.jetbrains.com/plugin/7856-adb-wifi.

Chapitre 4 : Langage Java et programmation orienté objet



Objectifs :

- Comprendre le paradigme orienté objet
- Programmer des classes JAVA
- Utiliser la bibliothèque de Java Standard Edition (JSE)