APPRENEZ android-studio

eBook gratuit non affilié créé à partir des contributeurs de Stack Overflow.

#android-

studio

Table des matières

À propos1
Chapitre 1: Démarrer avec android-studio
Remarques2
Versions2
Examples
Installation ou configuration
les fenêtres
Mac
Linux
Remarques5
Problème de téléchargement6
Raccourcis utiles
Raccourcis de navigation6
Raccourcis de navigation - Mac OS X6
Raccourcis de programmation6
Aperçu de différentes tailles d'écran (périphériques) et des orientations
1. Aperçu des différents périphériques
2. Orientation de commutation
Utilisez vos raccourcis d'outil préférés dans Android Studio10
Chapitre 2: Conseils et astuces pour Android Studio11
Examples11
Emulateur pour test
Modèle personnalisé en direct11
Exemple :
Pour atteindre cet objectif, vous devez ajouter un modèle Live personnalisé:
Utiliser des styles de code personnalisés, partager avec d'autres membres de l'équipe et F15
Chapitre 3: Mises à jour Android Studio
Examples
Mise à jour d'Android Studio dans Ubuntu18

Canaux de mise à jour Android Studio	
Vue d'ensemble	
Sélection d'un canal de mise à jour	19
Chapitre 4: Optimisation d'Android Studio	
Paramètres	
Examples	
Personnaliser l'option VM	
Les fenêtres:	
Mac:	
Linux:	20
Paramètre par défaut pour Android Studio 64 bits	
Réglage optimisé	
Crédits	



You can share this PDF with anyone you feel could benefit from it, downloaded the latest version from: android-studio

It is an unofficial and free android-studio ebook created for educational purposes. All the content is extracted from <u>Stack Overflow Documentation</u>, which is written by many hardworking individuals at Stack Overflow. It is neither affiliated with Stack Overflow nor official android-studio.

The content is released under Creative Commons BY-SA, and the list of contributors to each chapter are provided in the credits section at the end of this book. Images may be copyright of their respective owners unless otherwise specified. All trademarks and registered trademarks are the property of their respective company owners.

Use the content presented in this book at your own risk; it is not guaranteed to be correct nor accurate, please send your feedback and corrections to info@zzzprojects.com

Chapitre 1: Démarrer avec android-studio

Remarques

Android Studio est l'IDE officiel de Google pour le développement d'applications Android, basé sur IntelliJ IDEA. [1]

Il utilise un système de génération basé sur Gradle via le plugin Android pour Gradle. [2]

Versions

Version	Date de sortie
0.1.x	2013-05-01
0.2.x	2013-07-01
0.3.2	2013-10-01
0.4.2	2014-01-01
0.4.6	2014-03-01
0.5.2	2014-05-01
0.8.0	2014-06-01
0.8.6	2014-08-01
0.8.14	2014-10-01
1.0	2014-12-01
1.0.1	2014-12-02
1.1.0	2015-02-01
1.2.0	2015-04-02
1.2.1	2015-05-01
1.2.2	2015-06-01
1.3.0	2015-07-01
1.3.1	2015-08-01
1.3.2	2015-08-02

Version	Date de sortie
1.4.0	2015-09-01
1.4.1	2015-10-01
1.5.0	2015-11-01
1.5.1	2015-12-01
2.0	2016-04-01
2.1.0	2016-04-02
2.1.1	2016-05-01
2.1.2	2016-06-01
2.1.3	2016-08-01
2.2.0	2016-09-01
2.3.0	2017-03-02
2.3.1	2017-04-02

Examples

Installation ou configuration

les fenêtres

- Vérifiez que vous avez le JDK correct. Vous pouvez le vérifier en ouvrant l'invite de commande (appuyez sur la touche windows et écrivez cmd). Dans l'invite de commande, tapez javac -version, cela affichera la version actuelle de JDK installée sur votre correspondance ou une erreur * si Java est manquant. Si le JDK n'est pas disponible ou que la version est inférieure à 1,8, téléchargez le kit de développement Java SE 8.
- 2. Téléchargez le dernier Android Studio .
- 3. Lancez le fichier .exe téléchargé.
- 4. Suivez l'assistant pour installer Android Studio
- 5. Une fois l'installation terminée, ouvrez Android Studio à partir du raccourci créé sur le Bureau lors du processus d'installation.
- 6. Lorsque vous ouvrez Android Studio pour la première fois, il peut vous demander de copier

vos paramètres précédents, mais comme c'est la première fois, vous pouvez simplement choisir de ne rien copier.

7. Ensuite, Android Studio demande de télécharger les API requises pour développer des applications Android. Téléchargez-les, après avoir téléchargé l'IDE, vous pourrez écrire votre première application.

Mac

- Vérifiez que vous avez le JDK correct. Vous pouvez le vérifier en ouvrant le terminal (appuyez sur commande + espace et écrivez le terminal). Dans la ligne de commande, tapez javac -version, cela affichera la version actuelle de JDK installée sur votre correspondance ou une erreur * si Java est manquant. Si le JDK n'est pas disponible ou que la version est inférieure à 1,8, téléchargez le kit de développement Java SE 8. **
- 2. Téléchargez le dernier Android Studio .
- 3. Lancez le fichier . dmg téléchargé.
- 4. Glissez et déposez Android Studio dans le dossier Applications, puis lancez Android Studio.
- 5. Ouvrez Android Studio.
- 6. Lorsque vous ouvrez Android Studio pour la première fois, il peut vous demander de copier vos paramètres précédents, mais comme c'est la première fois, vous pouvez simplement choisir de ne rien copier.
- 7. Ensuite, Android Studio demande de télécharger les API requises pour développer des applications Android. Téléchargez-les, après avoir téléchargé l'IDE, vous pourrez écrire votre première application.

Linux

- Vérifiez que vous avez le JDK correct. Vous pouvez le vérifier en ouvrant le terminal (appuyez sur commande + espace et écrivez le terminal). Dans la ligne de commande, tapez javac -version, cela affichera la version actuelle de JDK installée sur votre correspondance ou une erreur * si Java est manquant. Si le JDK n'est pas disponible ou que la version est inférieure à 1,8, téléchargez le kit de développement Java SE 8.**
- 2. Téléchargez le dernier Android Studio .
- 3. Décompressez / extrayez Android Studio dans un dossier spécifique.
- 4. Ouvrez le terminal et accédez au chemin où vous avez extrait Android Studio. (Ensuite, utilisez la commande cd pour aller dans le dossier Android Studio.) Après être entré, nous devons nous rendre dans le dossier bin nouveau, utilisez la commande cd bin et entrez.

- 5. Maintenant, nous devons changer le mod de notre fichier requis, à savoir studio.sh, pour cela, entrez la commande sudo chmod 777 –R studio.sh, appuyez sur Entrée et écrivez votre mot de passe (le cas échéant) et entrez. (Vous pouvez également voir la liste des fichiers présents dans bin par la commande ls.).
- 6. Après avoir modifié le mod, il ne nous reste plus qu'à exécuter le fichier .studio.sh pour cela: entrez la commande ./studio.sh
- 7. Lorsque vous lancez la commande Android Studio pour la première fois, il peut vous demander de copier vos paramètres précédents, mais comme c'est votre première fois, vous pouvez simplement choisir de ne rien copier.
- 8. Ensuite, Android Studio demande de télécharger les API requises pour développer des applications Android. Téléchargez-les, après avoir téléchargé l'IDE, vous pourrez écrire votre première application.

Vous pouvez rencontrer "impossible d'exécuter l'outil SDK mksdcard" lorsque vous installez Android Studio dans un système d'exploitation Ubuntu 64 bits, car le studio nécessite 32 binaires. Pour surmonter cette erreur, fermez Android Studio et allez au terminal et lancez sudo apt-get install lib32z1 lib32ncurses5 lib32bz2-1.0 lib32stdc++6. Une fois l'installation terminée pour ces fichiers binaires, revenez à l'étape 6 et reprenez le processus d'installation.

Remarques

- Si vous avez déjà installé JDK et que vous continuez à l'obtenir, assurez-vous d'avoir défini JAVA_HOME dans vos system Variables. Vous pouvez vérifier cette réponse sur la façon d'en configurer une.
- Il existe toutefois des problèmes de stabilité connus dans Android Studio sur Mac lors de l'utilisation de JDK 1.8. Tant que ces problèmes ne sont pas résolus, vous pouvez améliorer la stabilité en rétrogradant votre JDK vers une version antérieure (mais pas inférieure à JDK 1.6).
- Lorsque le téléchargement d'Android Studio est terminé, vérifiez quelle version du **JDK** vous avez: ouvrez une ligne de commande et tapez javac -version. Si le JDK n'est *pas* disponible ou que la version est inférieure à *1,8*, téléchargez le kit de développement Java SE 8.
- Si vous utilisez une version 64 bits d' **Ubuntu**, vous devez installer des bibliothèques 32 bits avec les commandes suivantes:

```
sudo dpkg --add-architecture i386
sudo apt-get update
sudo apt-get install lib32z1 lib32ncurses5 libbz2-1.0:i386 lib32stdc++6
```

• Si vous exécutez Fedora 64 bits, la commande est la suivante:

sudo yum install zlib.i686 ncurses-libs.i686 bzip2-libs.i686

Problème de téléchargement

• Si vous rencontrez un problème tel que le message que votre système n'est pas compatible avec Android Studio (cela peut arriver lorsque vous utilisez un navigateur Web différent de Chrome), téléchargez Android Studio partir de: http://tools.android.com/

C'est tout!

Raccourcis utiles

Raccourcis de navigation

Ctrl+N
Ctrl + Shift + N
ALT + Left-Arrow; ALT + Right-Arrow
CTRL + E
CTRL + G
CTRL + SHIFT + BACKSPACE
CTRL + B
CTRL + ALT + B
F4
CTRL + U
Ctrl + Alt + H
CTRL + SHIFT + F

Raccourcis de navigation - Mac OS X

Go to line number

```
CMD + L
```

Raccourcis de programmation

Reformat code	CTRL + ALT + L
Optimize imports	CTRL + ALT + O
Code Completion	CTRL + SPACE
Issue quick fix	ALT + ENTER
Surround code block	CTRL + ALT + T
Rename and refactor	Shift + F6
Line Comment or Uncomment	CTRL + /
Block Comment or Uncomment	CTRL + SHIFT +
Go to previous/next method	ALT + UP/DOWN
Show parameters for method	CTRL + P
Quick documentation lookup	CTRL + Q
Project	Alt+1
Version Control	Alt+9
Run	Shift+F10
Debug	Shift+F9
Android Monitor	Alt+6
Return to Editor	Esc
Hide All Tool Windows	CTRL +Shift+F12
Auto generate code(constructor,	
getter/setter etc)	Alt+insert
Code completion	CTRL+Space

Smart code completion (by expected type) Show quick fix Duplicate Line Delete Line

CTRL+Shift+Space CTRL+Enter Ctrl+D Ctrl+Y

Aperçu de différentes tailles d'écran (périphériques) et des orientations

2.1.3

1. Aperçu des différents périphériques

Il y a un panneau de prévisualisation à droite du studio Android. Dans ce panneau, il y a un bouton avec le nom du périphérique avec lequel vous prévisualisez l' ui de votre application comme ceci.



Cliquez sur le petit indicateur déroulant de cet élément et un panneau flottant apparaîtra avec tous les périphériques prédéfinis. Vous pouvez sélectionner l'un d'entre eux pour prévisualiser votre un application avec différents périphériques mentionnant leurs tailles d'écran.

Vérifiez cette image

er.xml ×	- ,≣2	Preview	V	\$- →
		. -	🗓 Nexus 7 + 📄 + 🛈 AppTheme 🏧 + 🌍 +	l∰123 -
			D AVD: GalaxyNexus_API_23	🕼 🔯
			Nexus 5X (5.2", 1080 × 1920: 420dpi)	
			D Nexus 6P (5.7", 1440 × 2560: 560dpi)	
	_		In Nexus 6 (6.0", 1440 × 2560: 560dpi)	
			Image: Nexus 9 (8.9", 2048 × 1536: xhdpi)	
	_		Image: Nexus 5 (5.0", 1080 × 1920: xxhdpi)	
	=		🗔 Nexus 7 (7.0", 1200 × 1920: xhdpi)	
	- 1		🗔 Nexus 4 (4.7", 768 × 1280: xhdpi)	
	- 1		Image: Nexus 10 (10.1", 2560 × 1600: xhdpi)	
	- 1		V Nexus 7 (2012) (7.0", 800 × 1280: tvdpi)	
			🗔 Galaxy Nexus (4.7", 720 × 1280: xhdpi)	
	_		Image: Nexus S (4.0", 480 × 800: hdpi)	
			Image: Nexus One (3.7", 480 × 800: hdpi)	
			Android Wear Round Chin (320 × 290: tvdpi)	
			Android Wear Square (280 × 280: hdpi)	
			🚦 Android Wear Round (320 × 320: hdpi)	
1.5			Android TV (1080p) (1920 × 1080: xhdpi)	₩* <u>*</u>
atform a	nd Pl	lugin	Android TV (720p) (1280 × 720: tvdpi)	ace.
ad Setti	ngs:	Canno	Convis Dhanna and Tablata	imam\Docume:
the fil	e cor	ntent	Add Druice Definition	
adle syn	c sta	arted		
adle syn	C COI	nplete	Preview All Screen Sizes	a hu a la du a i d'
adle bui	Lasks	s: [:a] iniche	pp.generatebebagboartes, .app.generateb A in 1m Go 1/2mo	ebuganaro1a.

2. Orientation de commutation

Et à côté de ce bouton, il y a un autre bouton comme



En cliquant sur son menu déroulant, un panneau flottant apparaîtra avec des options telles que portrait, landscape etc. sélectionnez-en une pour prévisualiser dans différentes orientations. Vérifiez cette image



2.2.0

La possibilité de prévisualiser toutes les tailles d'écran n'est pas disponible sur Android Studio 2.2 Beta 2. Cependant, vous pouvez redimensionner l'écran de prévisualisation pour voir à quoi ressemblerait votre mise en page sur différentes tailles d'écran, comme indiqué dans la pièce jointe.



Vérifier: Android Studio 2.2 Prévisualiser toutes les tailles d'écran manquantes

Utilisez vos raccourcis d'outil préférés dans Android Studio

Allez dans Fichier> Paramètres> Keymap et sélectionnez l'option Keymaps parmi:

- Mac OS X
- Emacs
- Visual Studio
- Ecliser
- Netbeans
- Jbuilder

et autres, pour mapper les raccourcis vers les outils souhaités.

Lire Démarrer avec android-studio en ligne: https://riptutorial.com/fr/androidstudio/topic/830/demarrer-avec-android-studio

Chapitre 2: Conseils et astuces pour Android Studio

Examples

Emulateur pour test

S'il n'y a pas de besoin spécifique, choisissez x86_64 images système x86_64 avec les API Google pour créer l'émulateur.

Il fonctionne beaucoup plus rapidement que armeabi-v7a sur les ordinateurs Intel x86 x64.

Certaines bibliothèques SDK ont été compilées et conçues avec une architecture ARM.

Si vous essayez de les installer sur des émulateurs Intel, vous recevrez un message d'erreur indiquant:

"INSTALL_FAILED_NO_MATCHING_ABIS"

Vous pouvez opter pour un émulateur tiers tel que Genymotion ou Android Emulator autonome de Visual Studio .

Modèle personnalisé en direct

Exemple :



Pour atteindre cet objectif, vous devez ajouter un modèle Live personnalisé:

- 1. Ouvrir les paramètres [Ctrl + Alt + s]
- 2. Tapez "Modèles en direct" dans la barre de recherche en haut à gauche.
- 3. Cliquez sur le + pour ajouter un "groupe de modèles" et tapez un nom de groupe (par exemple: MyTemplate) pour continuer



- 4. Choisissez votre groupe de templates personnalisé (ex: MyTemplate), puis cliquez sur + et choisissez "Live Template".
- 5. Je vais prendre "dire" par exemple:

say -> startActivity(new Intent(\$from\$.this,\$to\$.class));

6. Tapez la clé que vous souhaitez simplifier dans "Abréviation" (par exemple: say), et tapez l'instruction dans le "Text Template" (par exemple: startActivity(new Intent(\$from\$.this,\$to\$.class));)

٩	Editor > Live Templates		
 Appearance & Behavior Keymap 	By default expand with Tab ▼		
 ▼ Editor ▶ General ▶ Colors & Fonts ▶ Code Style □ Inspections □ File and Code Templates □ File Encodings □ Live Templates 	 Fancy's Template Groovy Mhtml/xml Miterations Kotlin Kotlin Mytemplate Say (start activity) ✓ other 		
File Types ▶ Copyright @	 ▶ ✓ output ▶ ✓ plain 		
Emmet Images Intentions ▶ Language Injections া	Abbreviation: say Description: start activity		
Spelling TODO Plugins > Version Control > Build, Execution, Deployment > Languages & Frameworks > Toolo	<pre>startActivity(new Intent(\$from\$.this,\$to\$.class));</pre>		
 Other Settings 	No applicable contexts yet. Define		

- 7. Cliquez sur le "définir" dans le bas à gauche et choisissez la situation (par exemple: Java)
- 8. Cliquez sur "Modifier les variables", définissez l'expression (voir détail: Boîte de dialogue Modifier les variables de modèle).

👳 Edit Template Variables					
Name	Expression	Default value	Skip if defin 🛧		
from	className()				
to					
		ок	ancel Help		

9. Cliquez sur "OK" et "Appliquer". Et essayez de taper "say" dans votre éditeur.

Utiliser des styles de code personnalisés, partager avec d'autres membres de l'équipe et Format automatique avec un raccourci

Il est possible d'avoir vos propres styles de code personnalisés, de les partager avec d'autres membres de l'équipe et d'utiliser un raccourci pour formater automatiquement le code dans un fichier.

Pour créer votre propre style de code personnalisé, accédez à: Préférences -> Editeur -> Style de code

Il existe des paramètres de style de code généraux ici. Vous pouvez également sélectionner la langue (Java pour Android) et définir les paramètres comme vous le souhaitez. Il y a *beaucoup* de paramètres.

Une fois cela fait, enregistrez votre style de code. Ceci est juste pour garder en sécurité.

Scheme: P	roject	•	Manage				
kight mai	gin (co	iumns <i>)</i> :	Code St	4 yle Scher	nes	on typing	
Indents Detec	t t Pro	ject	Name		Save As		
		Сор	by Project So	cheme to	Global List		
Form	F	inter new	v scheme na	me:			
	ς [MyTeam	Style				
Ν.					Cance	ЫОК	
Forma					Close		
Option	s						

Une fois enregistré, sélectionnez "Copier dans le projet" dans la boîte de dialogue "Gérer". Il vous sera demandé si vous souhaitez passer à ce schéma créé. Répondez oui.

Code Style Schemes				
t Name t Project	Save As			
MyTeamStyle	Delete			
n l	Copy to Project			
6 *	Export			
	Import			
	Close			
Copy Scheme to Project Scheme 'MyTeamStyle' was copied to be used as the project scheme. Switch to this created scheme?				
able	No Yes			

Une fois cela fait, fermez la boîte de dialogue des préférences de style de code.

Maintenant, vérifiez que vos paramètres de style de code ont été enregistrés dans: .idea / codeStyleSettings.xml

Assurez-vous que ce fichier n'est pas ignoré dans votre système de contrôle de version afin que vous puissiez le partager avec vos coéquipiers.

Une fois que vos coéquipiers ont ce fichier, ils doivent également avoir les mêmes paramètres.

Maintenant, lors de l'édition d'un fichier, vous pouvez formater le code en sélectionnant:

Code -> Reformat Code

Raccourcis pour reformater le code (tiré de <u>cette</u> réponse - voir la réponse pour plus de détails sur la résolution des problèmes avec un raccourci dans Ubuntu):

Gagner

Ctrl 🕇 Alt 🕇 L

Linux:

Ctrl + Maj + Alt + L

Mac:

Option + Commande + L

Lorsque vous effectuez un reformatage de code, une petite boîte de dialogue devrait apparaître pour vous informer des modifications apportées. Vous pouvez cliquer sur "Afficher" dans cette boîte de dialogue pour afficher le "Reformat File File Dialogue".



Vous pouvez également afficher cette boîte de dialogue à partir du menu Code et de son raccourci correspondant.



Soyez averti que "Seul le texte modifié par VCS" ne fonctionne pas toujours selon la manière dont le code a été modifié (il peut ignorer une règle si une partie du code n'a pas été modifiée).

Vous pouvez également sélectionner du texte et afficher la boîte de dialogue du code de reformatage pour ne formater que le texte sélectionné.

Lire Conseils et astuces pour Android Studio en ligne: https://riptutorial.com/fr/androidstudio/topic/2228/conseils-et-astuces-pour-android-studio

Chapitre 3: Mises à jour Android Studio

Examples

Mise à jour d'Android Studio dans Ubuntu

Si vous pouvez télécharger une mise à jour d'Android Studio, mais après le redémarrage, rien ne se passe, consultez l'exemple suivant:

- 1. Une fois le patch téléchargé et Android Studio fermé, ouvrez le terminal
- 2. Allez dans votre dossier de studio Android, par exemple ${\tt cd}$ ~/android-studio
- 3. Allez dans le sous-dossier $\tt bin$: <code>cd bin</code>
- 4. Assurez-vous que votre fichier studio.sh dispose des autorisations d'exécution: chmod +x studio.sh
- 5. Exécuter Android Studio à partir d'ici: ./studio.sh

Après cela, Android Studio trouvera le patch et l'installera. Ensuite, vous pouvez fermer Android Studio et l'exécuter normalement (quelle que soit votre préférence, je l'exécute depuis le pavé Launcher).

Canaux de mise à jour Android Studio

Vue d'ensemble

Le mécanisme de mise à jour intégré d'Android Studio peut être configuré pour recevoir des mises à jour via l'un de ces **quatre** canaux:

- Canary: Bleeding edge, publié à propos de l'hebdomadaire. Ce sont des aperçus préliminaires publiés afin d'obtenir un retour d'expérience au cours du développement. La chaîne canarienne aura toujours la dernière version, y compris les mises à jour des versions bêta ou stables. Nous vous recommandons de lancer des versions de Canary côte à côte avec une installation bêta ou stable.
- **Dev:** Canary se construit après une série complète de tests internes promus dans le Dev Channel.
- Bêta: Lancez des candidats basés sur des versions stables de Canary, publiés et mis à jour pour obtenir des commentaires avant la publication de la version stable. Le canal bêta sera mis à jour avec de nouvelles versions stables jusqu'à ce que la nouvelle version de la version canari passe en version bêta.
- Stable: la version stable officielle, disponible sur le site Android Developer.

Téléchargez les installations complètes de la compilation offertes dans chacun de ces canaux: Canary, Dev, Beta, Stable.

Vous pouvez également le créer vous-même en suivant les instructions de la vue d'ensemble de la construction .

Sélection d'un canal de mise à jour

Pour sélectionner le canal de mise à jour pour une installation Android Studio, procédez comme suit:

File > Settings > System Settings > Updates

et choisissez le canal approprié sur lequel vérifier les mises à jour:

Lorsqu'une mise à jour est disponible, vous serez invité par l'EDI:

A new Android Studio 2.2 Preview 3 is available in the	canary channel.			
If you're using 2.0.0 or 2.1.0, you should update to You can also update to 2.2 to try new features. Canary builds are the bleeding edge, released about w they are still subject to bugs, as we want people to se	address a serious s reekly. While these bui e what's new as soon	ecurity vulnerability. If Ids do get tested, as possible.	you're using 2.1.1+ o	r 2.2.0+, yo
For slightly more predictable builds, use Settings U	pdates and select the	Dev Channel.		
Current version: 2.1.1 (build 143.2821654) New version: 2.2 Preview 3 (build 145.2949926)				
To configure automatic update settings, see the Update	ates dialog of your IDB	Preferences.		
_				
	Remind Me Later	Download	Release Notes	Ignore Thi

La sélection de la mise à jour affichera la boîte de dialogue d'informations de mise à jour avec des informations détaillées sur le correctif disponible, sa taille et son canal:

Lire Mises à jour Android Studio en ligne: https://riptutorial.com/fr/android-studio/topic/1998/misesa-jour-android-studio

Chapitre 4: Optimisation d'Android Studio

Paramètres

Paramètre	Détail
Xms	Allocation de mémoire initiale
Xmx	Max mémoire allouer

Examples

Personnaliser l'option VM

Vous pouvez remplacer les vmoptions par défaut par vos propres paramètres personnels en choisissant **Aide> Modifier les options VM personnalisées** dans la barre d'outils Android Studio. Cela créera une copie locale que vous êtes libre de modifier.

Vous pouvez également modifier les vmoptions par défaut directement en utilisant les chemins indiqués ci-dessous. Notez que cette méthode n'est pas recommandée et que vos modifications peuvent être écrasées lors de la mise à jour d'Android Studio.

Les fenêtres:

```
%USERPROFILE%\.{FOLDER_NAME}\studio.exe.vmoptions and/or
%USERPROFILE%\.{FOLDER_NAME}\studio64.exe.vmoptions
%USERPROFILE%\.{FOLDER_NAME}\idea.properties
```

Mac:

```
~/Library/Preferences/{FOLDER_NAME}/studio.vmoptions
~/Library/Preferences/{FOLDER_NAME}/idea.properties
```

Linux:

```
~/.{FOLDER_NAME}/studio.vmoptions
~/.{FOLDER_NAME}/studio64.vmoptions
~/.{FOLDER_NAME}/idea.properties
```

Paramètre par défaut pour Android Studio 64 bits

-Xms128m

```
-Xmx750m
-XX:MaxPermSize=350m
-XX:ReservedCodeCacheSize=96m
-ea
-Dsun.io.useCanonCaches=false
-Djava.net.preferIPv4Stack=true
-Djna.nosys=true
-Djna.boot.library.path=
-Djna.debug_load=true
-Djna.debug_load=true
-Djna.debug_load.jna=true
-Djsse.enableSNIExtension=false
-XX:+UseCodeCacheFlushing
-XX:+UseConcMarkSweepGC
-XX:SoftRefLRUPolicyMSPerMB=50
```

```
-Didea.platform.prefix=AndroidStudio
```

```
-Didea.paths.selector=AndroidStudio
```

Réglage optimisé

```
-Xms1024m
-Xmx4096m
-XX:MaxPermSize=1024m
-XX:ReservedCodeCacheSize=256m
-ea
-Dsun.io.useCanonCaches=false
-Djava.net.preferIPv4Stack=true
-Djna.nosys=true
-Djna.boot.library.path=
-Djna.debug_load=true
-Djna.debug_load.jna=true
-Djsse.enableSNIExtension=false
-XX:+UseCodeCacheFlushing
-XX:+UseConcMarkSweepGC
-XX:SoftRefLRUPolicyMSPerMB=50
-Didea.platform.prefix=AndroidStudio
-Didea.paths.selector=AndroidStudio
```

Lire Optimisation d'Android Studio en ligne: https://riptutorial.com/fr/androidstudio/topic/3247/optimisation-d-android-studio



S. No	Chapitres	Contributeurs
1	Démarrer avec android-studio	Ajay, Anurag, ARUN.G, Atif Farrukh, CJBS, Community, Daniel Nugent, H. Pauwelyn, Inzimam Tariq IT, kalan, murf, Nissim R, Paul Thompson, petrumo, piotrek1543, pRaNaY, Rob R., SaravInfern, Shashanth, sudo, Valentin Kuhn
2	Conseils et astuces pour Android Studio	Damindu, hellyale, meunicorn, piotrek1543, Shashanth, the_new_mr
3	Mises à jour Android Studio	piotrek1543, Yury Fedorov
4	Optimisation d'Android Studio	Kevin Cooper, manfcas, piotrek1543, UmAnusorn, Yury Fedorov