



آموزش برنامه نویسی اندروید در محیط اندروید استودیو

نصب JDK و Android Studio

مدرس : سید مهدی مطهری

www.android-studio.ir



به نام خدا

آشنایی با (JDK) Java SE Development Kit و نصب آن

همانطور که در فصل اول در مورد جاوا به مختصر توضیحاتی ارائه شد، این زبان Native نیست و برای اجرا به یک ماشین واسط یا ران تایم (Runtime) نیازمند است که این وظیفه در مورد زبان جاوا بر عهده Java Runtime Environment (JRE) می باشد. برای کامپایل کدهای جاوا، به JDK نیاز است که بسته JDK، ران تایم JRE را نیز دربر گرفته و با دانلود و نصب JDK نیازی به نصب جداگانه JRE نیست.

نکته: Native بودن جاوا به معنی Native نبودن اپ های نوشته شده به این زبان برای اندروید نیست! جاوا زبان رسمی اندروید است که از سوی گوگل پشتیبانی می شود. ضمن اینکه بیشترین داکيومنتها، کتابخانه ها و همچنین مباحث مطرح شده در فروم ها و وبسایت ها پیرامون مباحث توسعه اندروید، مربوط به جاوا است. البته به تازگی و در نسخه ۳ اندروید استودیو، پشتیبانی از زبان Kotlin (کاتلین) نیز اضافه شده که فعلا به آن ورود نمی کنیم و در آینده به آن خواهیم پرداخت.

نصب JDK در فصل ۱ (آموزش زبان جاوا) نیز توضیح داده شده بنابراین اگر قبلا JDK را نصب کرده اید، نیاز به نصب مجدد نیست. به جهت سهولت، تمامی ابزارهای موردنیاز برای برنامه نویسی اندروید در پست مربوط به همین مبحث آموزشی در وب سایت با عنوان "نصب و راه اندازی اندروید استودیو" برای دانلود در دسترس شما قرار گرفته است تا اگر به هر دلیل قادر به دانلود ابزار معرفی شده از وب سایت اصلی شان نباشید، به یک نسخه پشتیبان دسترسی داشته باشید:

<http://android-studio.ir/?p=249>

ابزار JDK متعلق به شرکت Oracle (اوراکل) بوده که به دلیل اعمال تحریم ها بر ضد کشور ایران، کاربران با IP این کشور قادر به مشاهده صفحات وب سایت این شرکت و همچنین دانلود ابزارها نیستند. برای دانلود جدیدترین نسخه های JDK می توانید از نرم افزارهای تغییر آی پی استفاده کرده و یا به سایر وب سایت ها از جمله وب سایت ما (لینک بالا) مراجعه کنید.

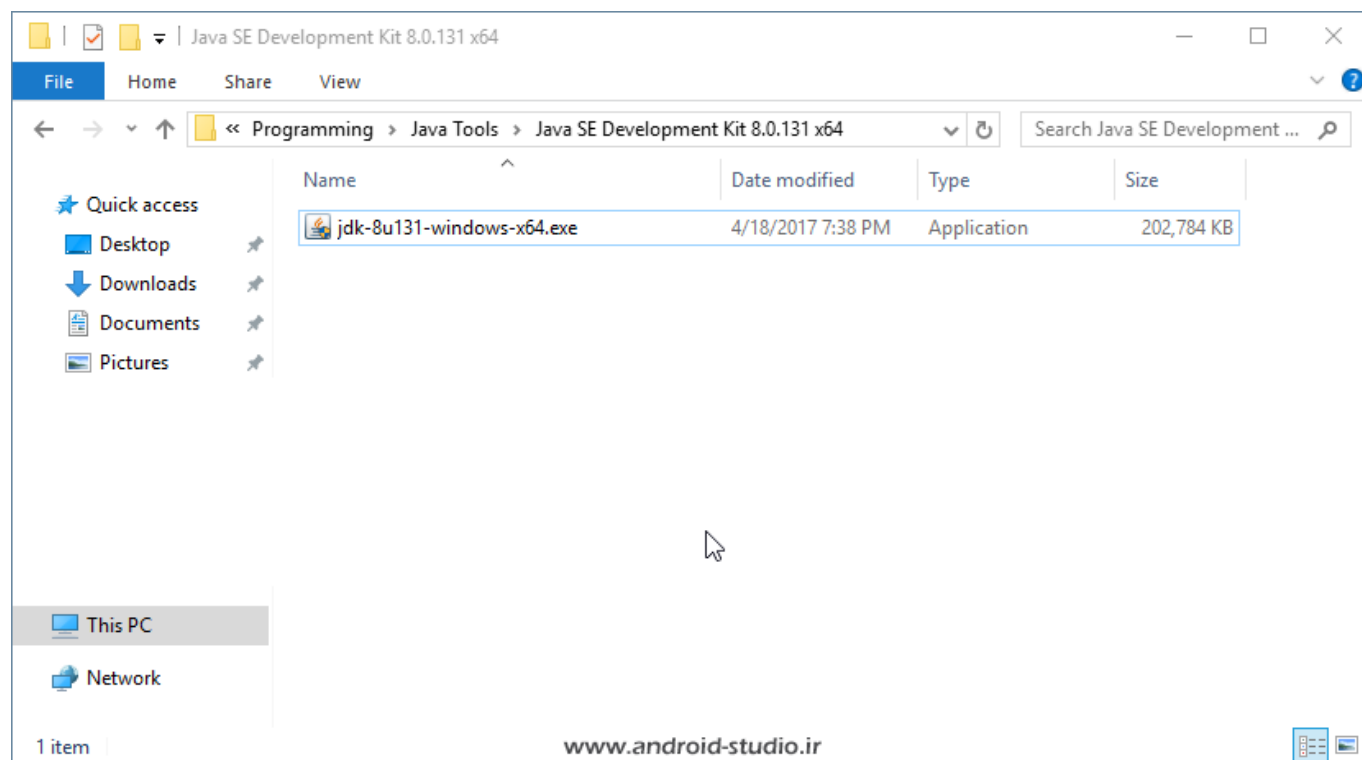
لینک دانلود در سایت شرکت اوراکل :

<https://www.oracle.com/technetwork/java/javase/downloads/index.html>

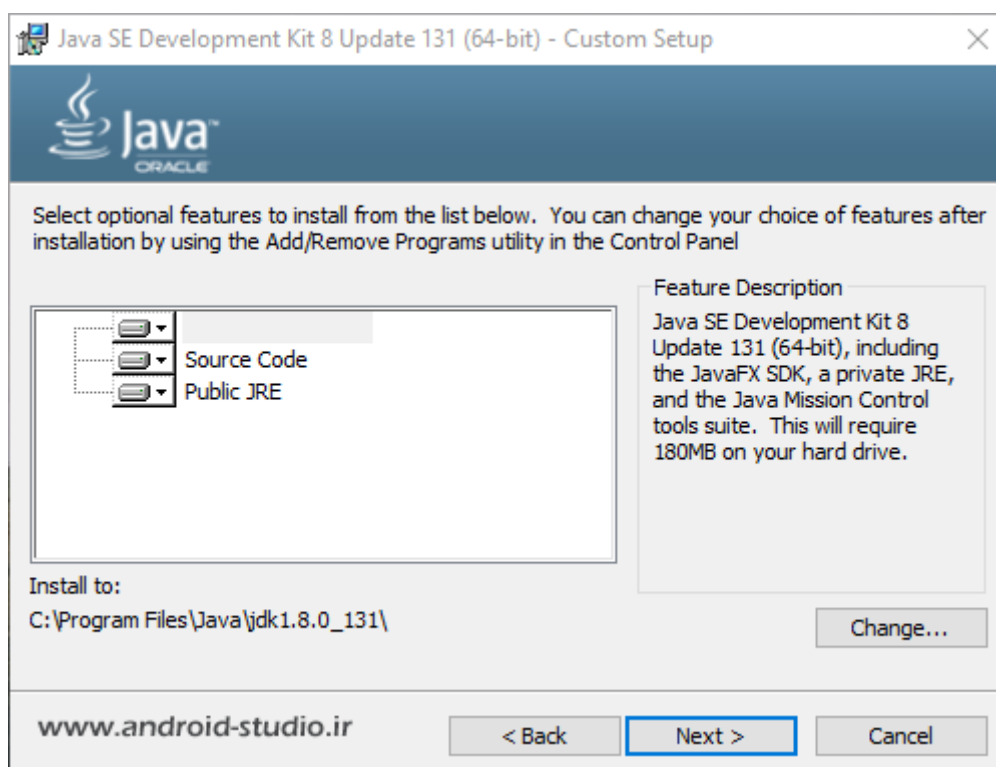
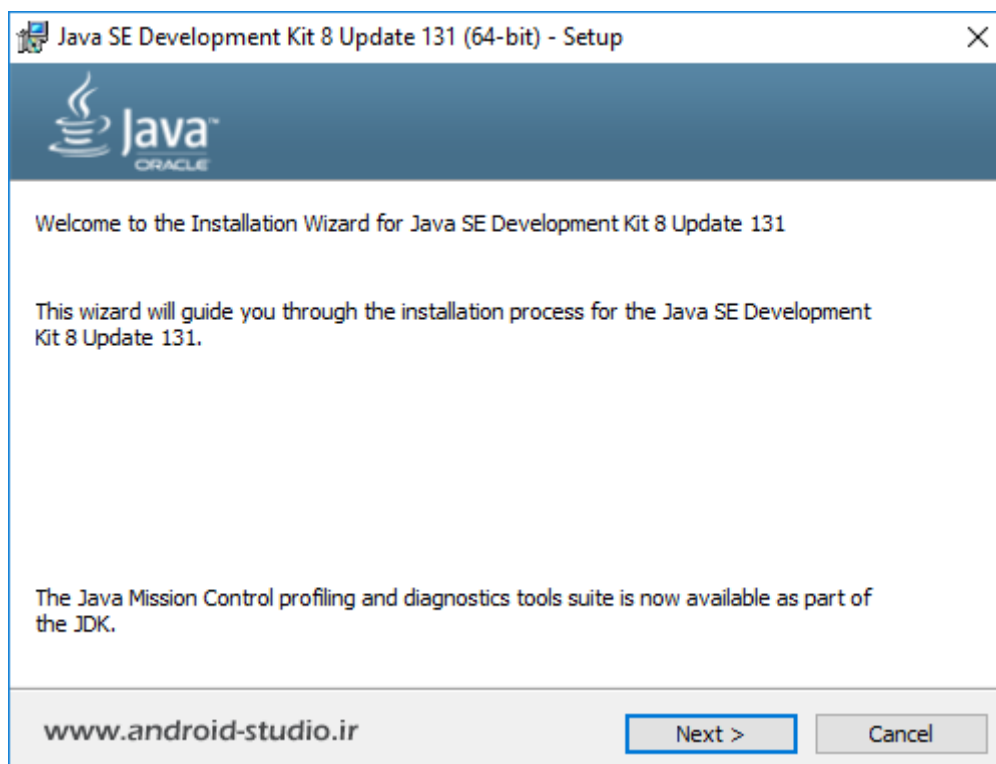


شیوه کار در سیستم عاملهای ویندوز ، لینوکس و مکینتاش تا حدود زیادی یکسان بوده و بنده مباحث آموزشی را در محیط ویندوز تهیه می کنم.

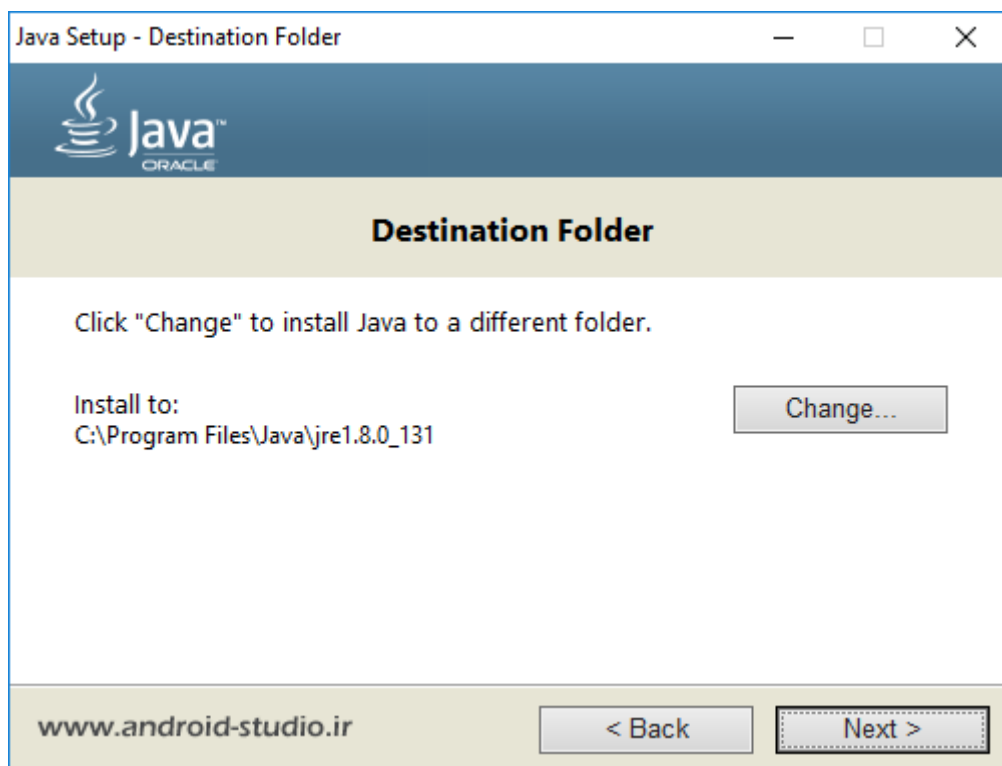
JDK در دو نسخه ۳۲ و ۶۴ بیتی منتشر شده که می بایست متناسب با معماری سیستم عامل خود (۳۲ یا ۶۴ بیتی بودن آن)، نسخه مناسب را دانلود و نصب کنید.



به مرور زمان نسخه های جدیدتری از ابزار و نرم افزارها منتشر می شود و ممکن است فایلی که شما در اختیار دارید نام آن متفاوت با نام تصویر فوق باشد. 8u131 یعنی نسخه ۸ آپدیت ۱۳۱. فایل را اجرا کرده و مراحل نصب را ادامه می دهم:



در این مرحله امکان تغییر مسیر نصب JDK وجود دارد که من بدون تغییر مسیر به مرحله بعد می روم:



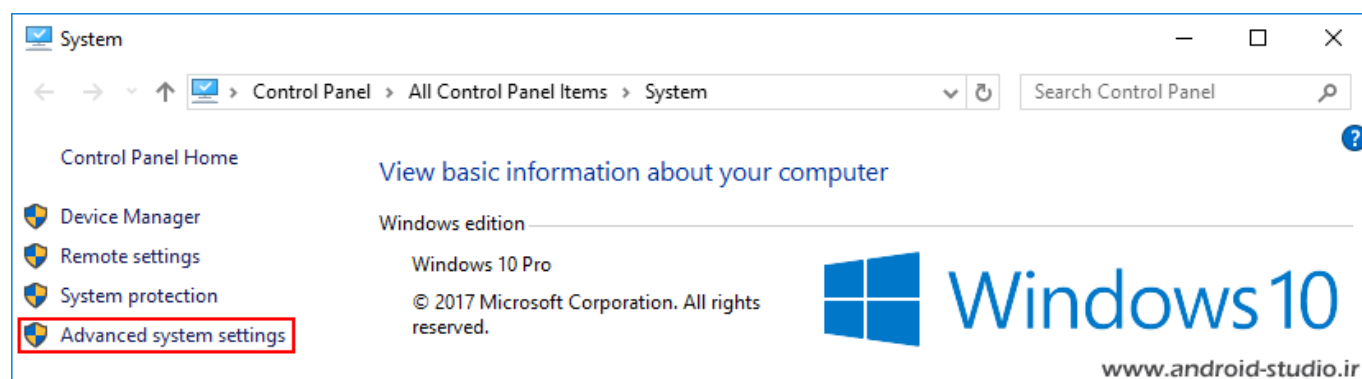
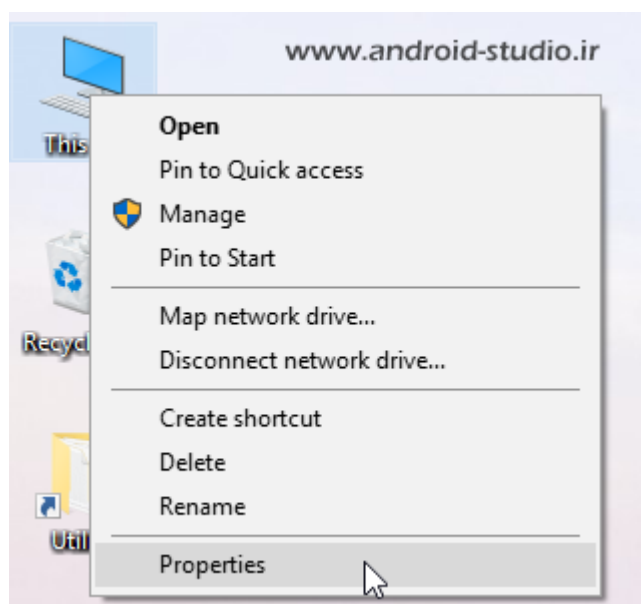
این مرحله نیز مربوط به تغییر مسیر نصب JRE بوده که تغییری ایجاد نمی‌کنم.



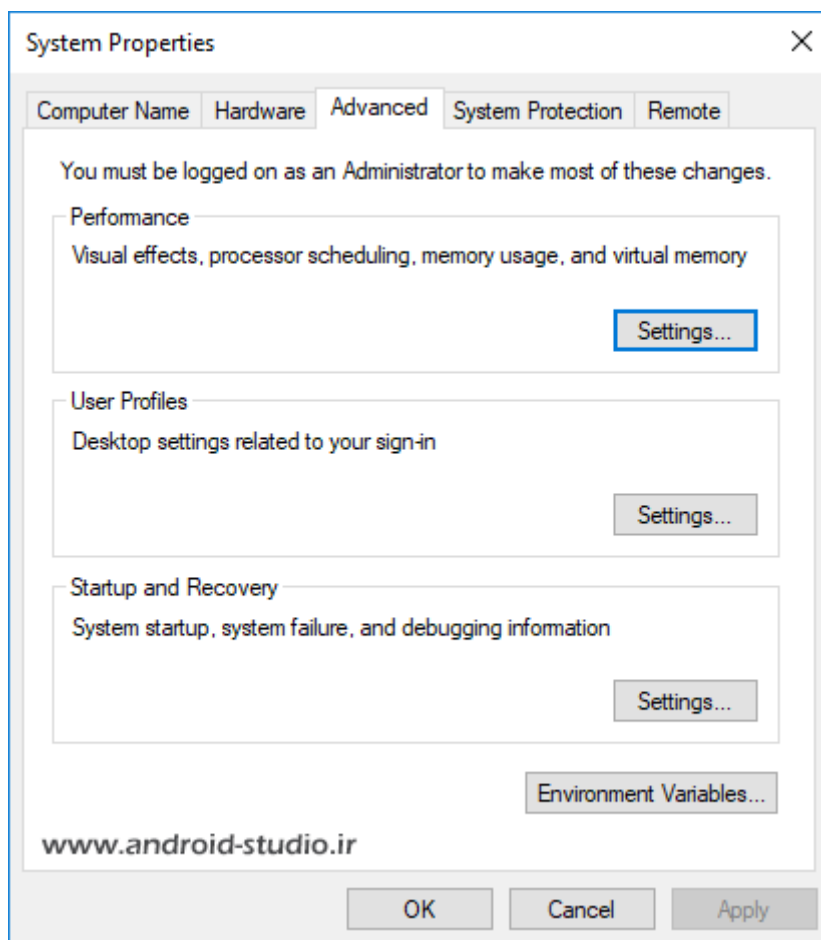
نصب با موفقیت به اتمام رسید. قبل از نصب اندروید استودیو لازم است مسیر نصب جاوا را به سیستم عامل معرفی کنیم. در غیر اینصورت اندروید استودیو قادر به شناسایی جاوا نیست. مطابق تصاویر زیر در



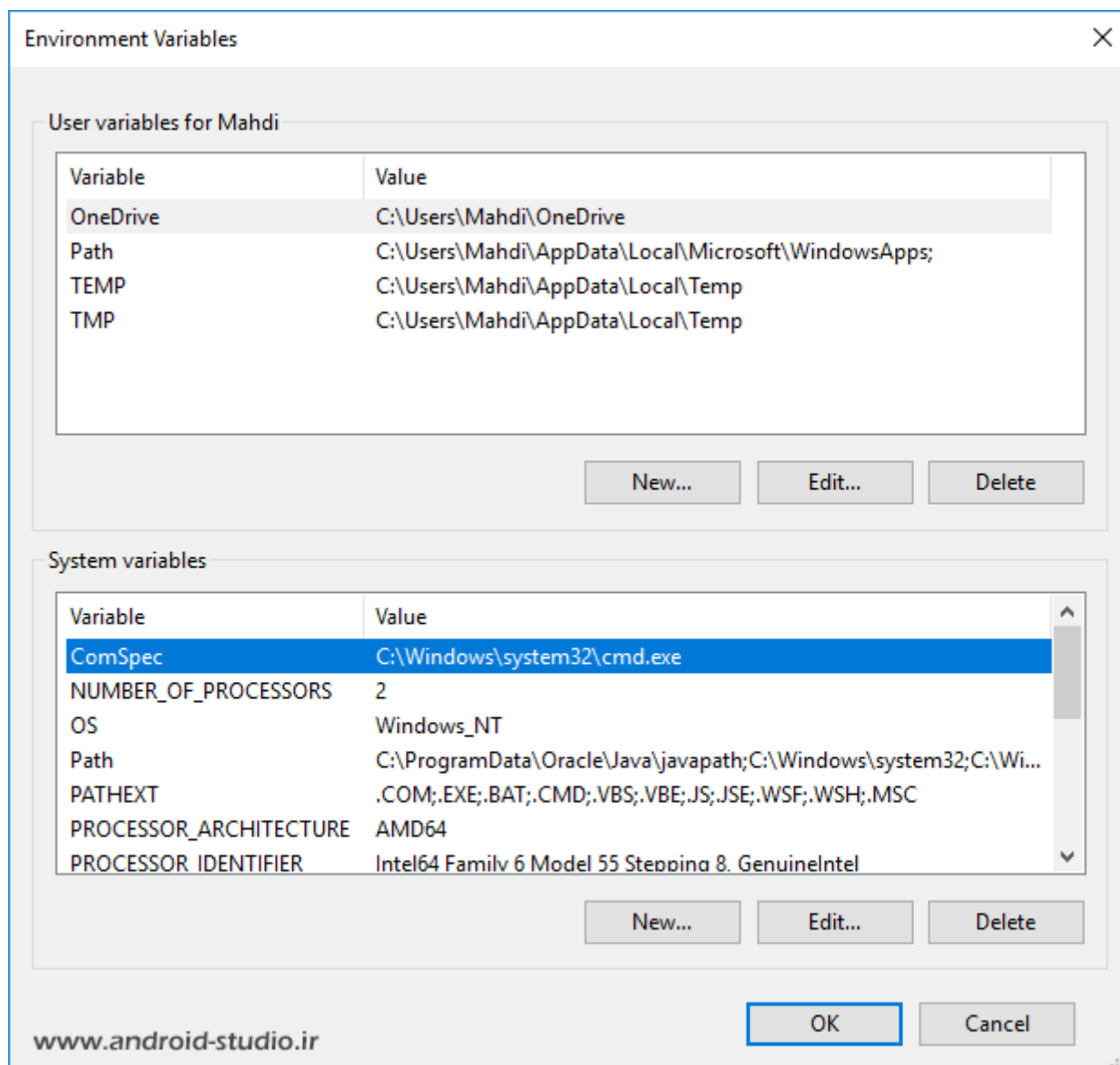
تنظیمات ویندوز و قسمت Environment Variables مسیر نصب JDK را با نام JAVA_HOME به عنوان یک System Variable به سیستم عامل معرفی می کنیم:



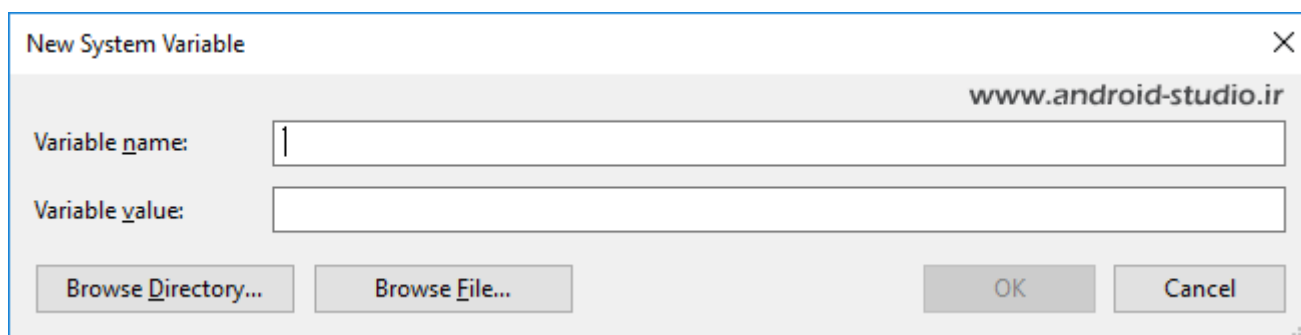
گزینه advanced system settings



گزینه Environment Variables



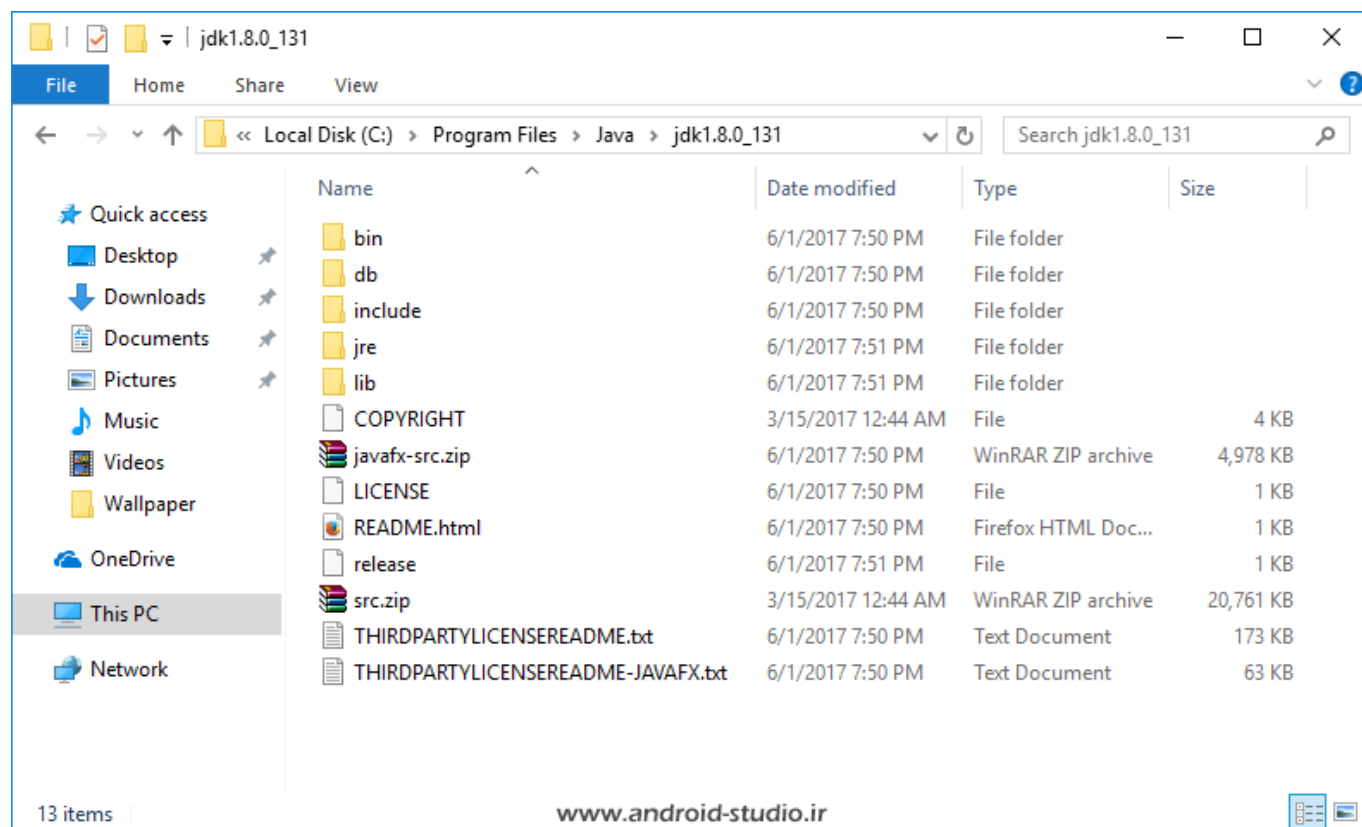
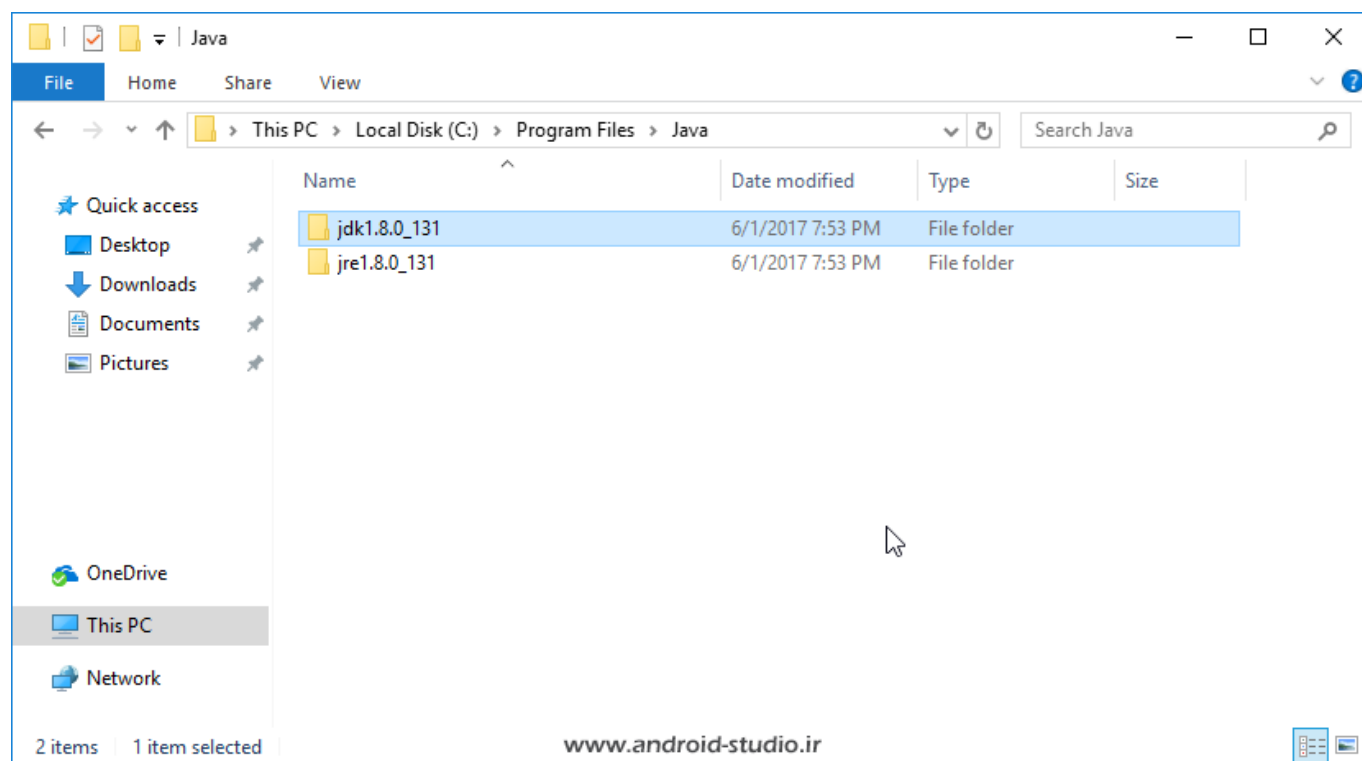
گزینه New

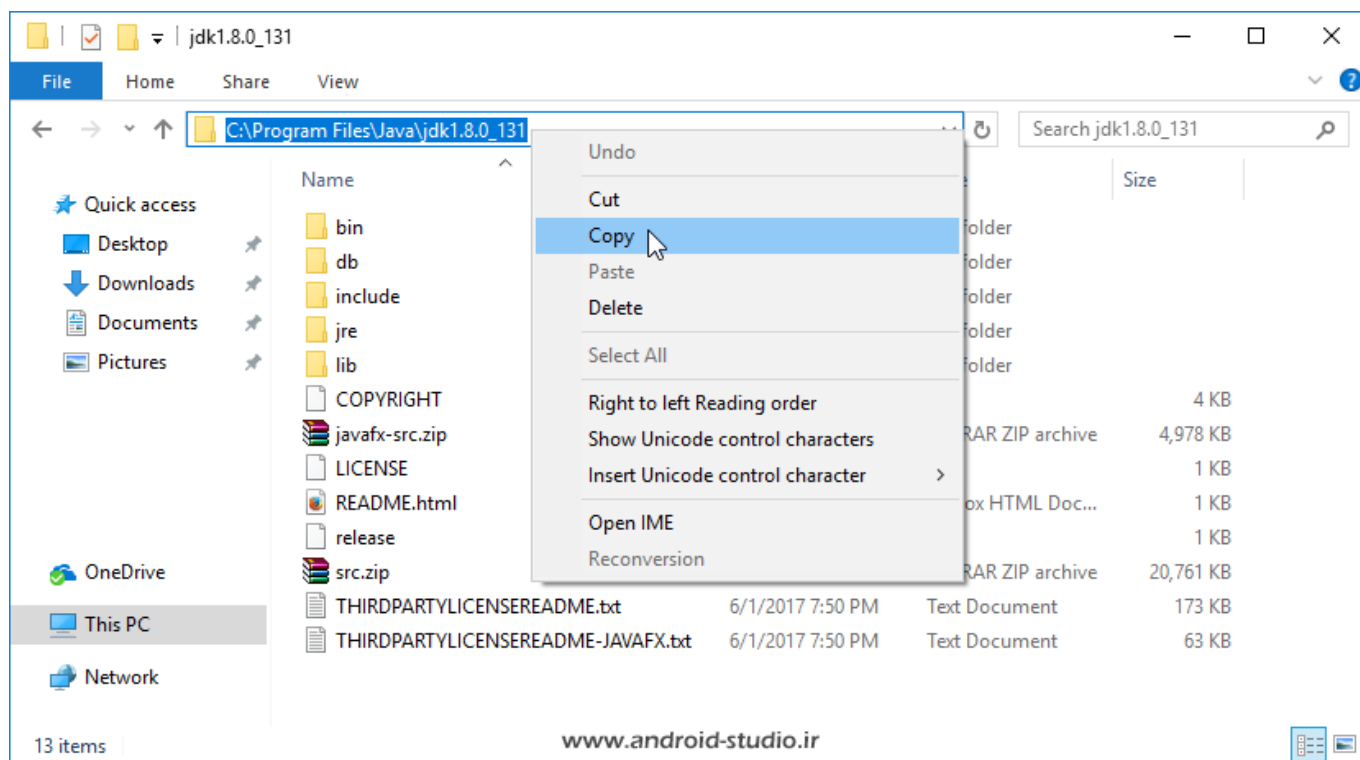


در قسمت Variable name عبارت JAVA_HOME و در value مسیر نصب JDK را وارد کردم. توجه داشته باشید عبارت JAVA_HOME عینا به همین صورت باید نوشته شود، یعنی حروف بزرگ و کاراکتر

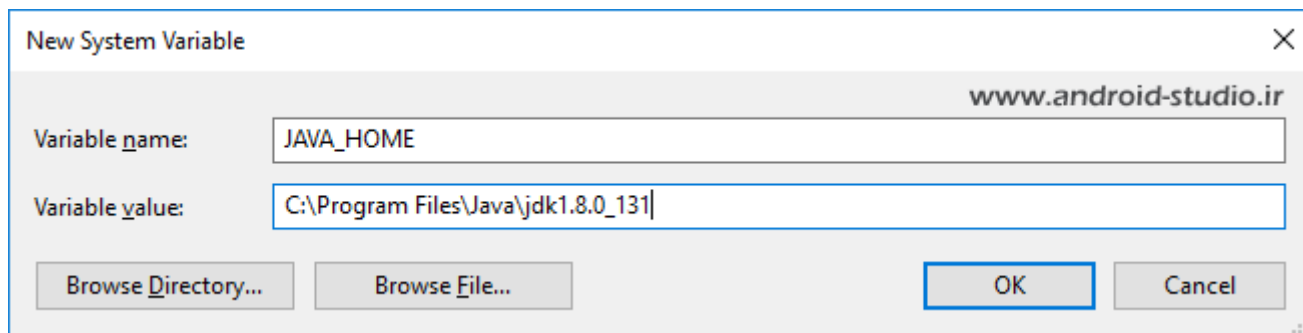


Underline مابین دو کلمه. مسیر نصب JDK هم بهتر است از نوار آدرس Windows Explorer کپی شود تا از صحت آدرس اطمینان داشته باشیم:

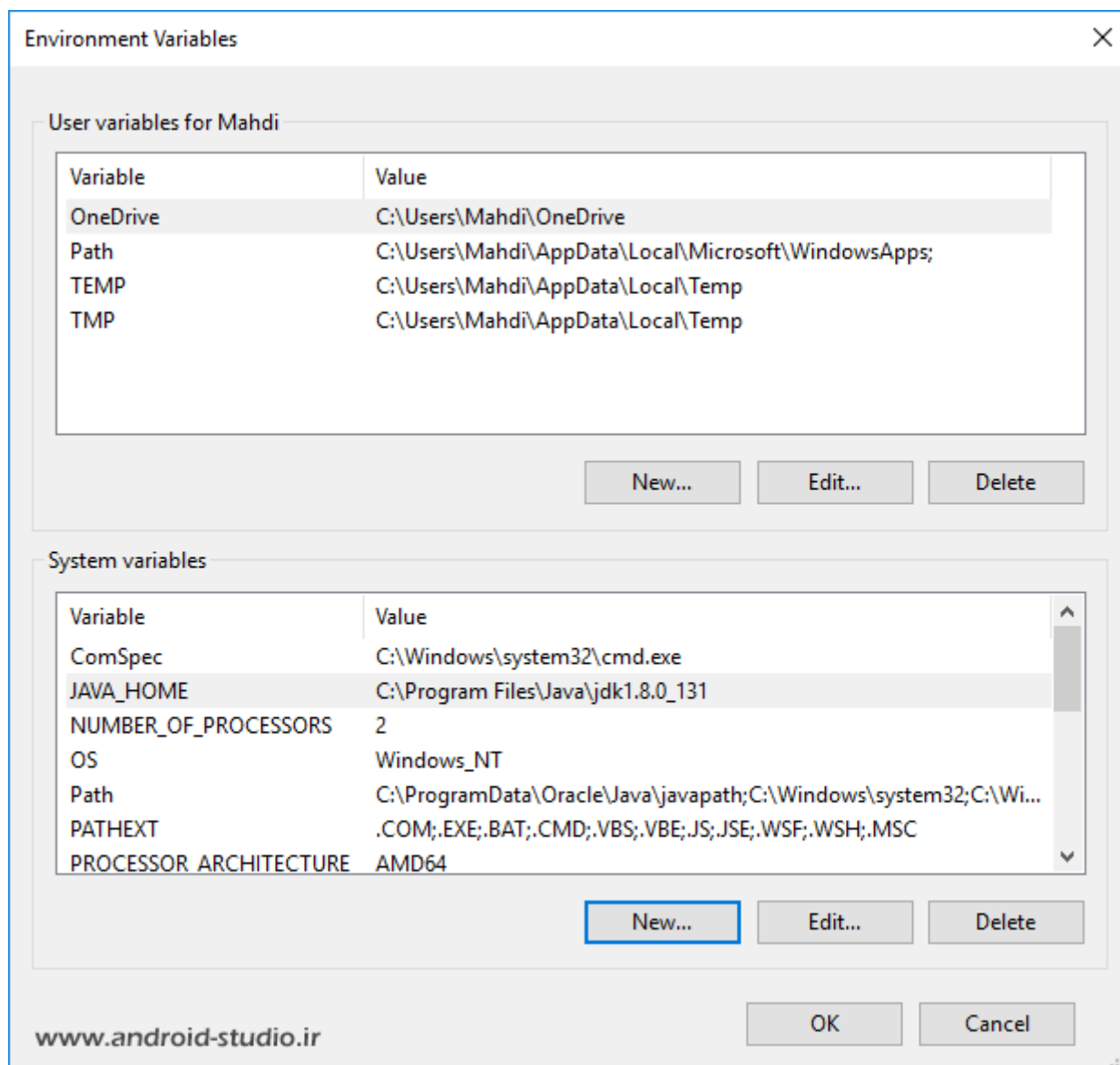




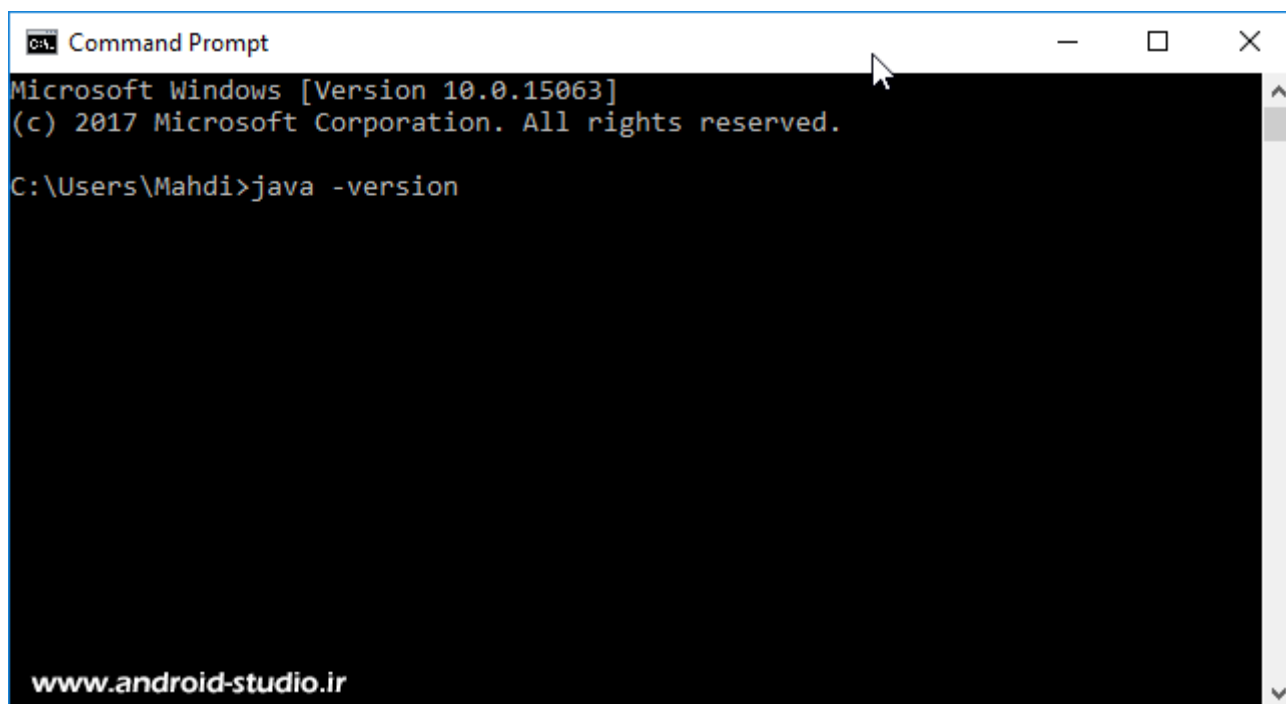
در نهایت به اینصورت تکمیل می شود:



OK می کنم:



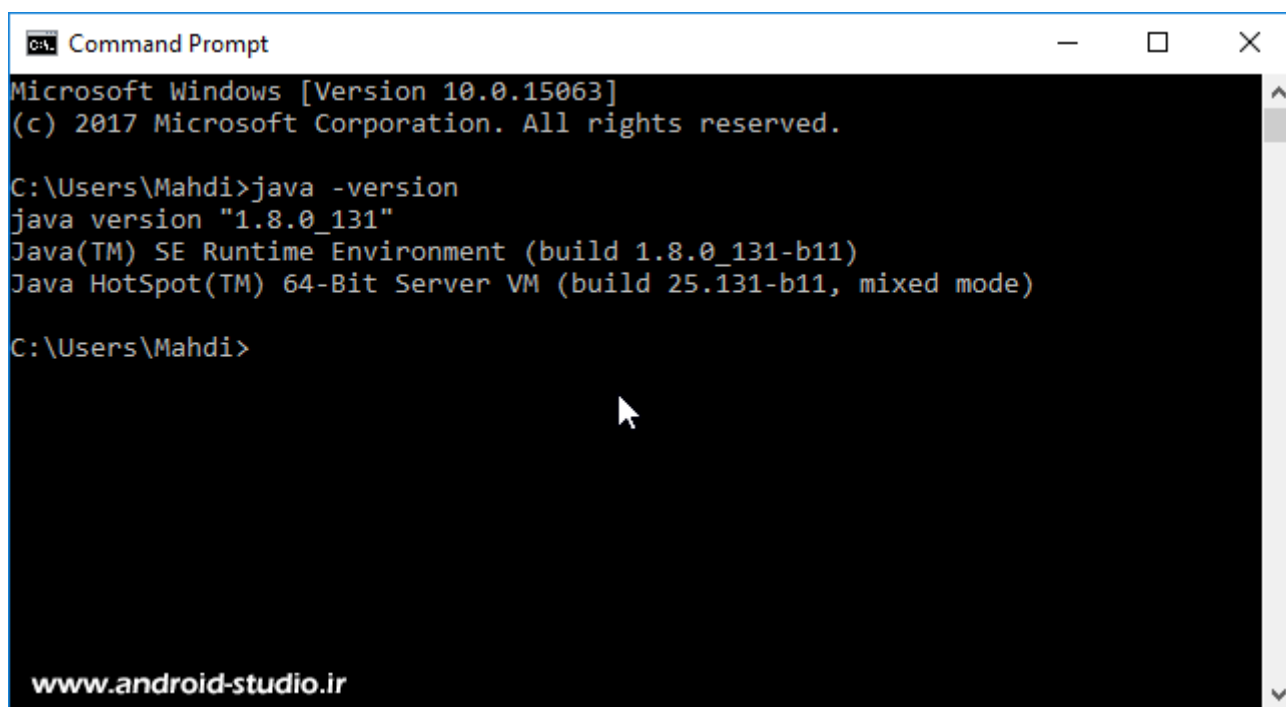
مشاهده می کنید JAVA_HOME به لیست اضافه شده. می خواهیم مطمئن شوم جاوا را به درستی به سیستم عامل معرفی کرده ام. کافیه در Command Prompt دستور `java -version` را وارد کنم. برای دسترسی به Command Prompt درون منوی Start ویندوز نام آن را جستجو کرده و یا درون Run (کلید ترکیبی Windows + R) عبارت `cmd` را وارد کنید:



```
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Mahdi>java -version
```

دستور `java -version` را نوشته و سپس Enter :



```
Microsoft Windows [Version 10.0.15063]
(c) 2017 Microsoft Corporation. All rights reserved.

C:\Users\Mahdi>java -version
java version "1.8.0_131"
Java(TM) SE Runtime Environment (build 1.8.0_131-b11)
Java HotSpot(TM) 64-Bit Server VM (build 25.131-b11, mixed mode)

C:\Users\Mahdi>
```

اطلاعات جاوای نصب شده به درستی نمایش داده شد. یعنی این مرحله را با موفقیت به پایان رساندیم.



نصب Android Studio

در حال حاضر که مشغول تهیه این مبحث آموزشی هستم، اندروید استودیو نسخه ۳.۰.۱ جدیدترین نسخه منتشر شده از سوی گوگل است. در نسخه های جدیدتر عموماً باگ ها رفع شده و یا امکانات جدیدی اضافه می شود.

در نسخه ۳ اندروید استودیو نیز تغییراتی صورت گرفته و امکانات جدیدی اضافه شده که به مهمترین موارد اشاره می کنم:

- ✓ پشتیبانی از زبان Kotlin در کنار زبان Java
- ✓ افزایش سرعت بیلد سیستم گریدل (Gradle)
- ✓ پشتیبانی از ویژگی Instant App
- ✓ معرفی ابزار جدید مانیتورینگ با نام Android Profiler که جایگزین Android Monitor نسخه های گذشته است

(به احتمال زیاد مفاهیم ذکر شده فوق برای شما نا آشناست که جای نگرانی نیست و در ادامه مباحث آموزشی و در جایی که نیاز باشد توضیحات لازم ارائه خواهد شد).

یک تغییر دیگر که در ورژن ۳ رخ داده، حذف نسخه Bundle است و از این به بعد فقط نسخه IDE اندروید استودیو منتشر می شود. یعنی ابزار SDK مانند قبل بطور خودکار و آفلاین نصب نمی شود.

نکته: IDE یا Integrated Development Environment به معنی محیط توسعه است.

نکته: SDK یا Software Development Kit شامل ابزار / کتابخانه / کامپوننت / API های لازم برای توسعه نرم افزار می باشد.

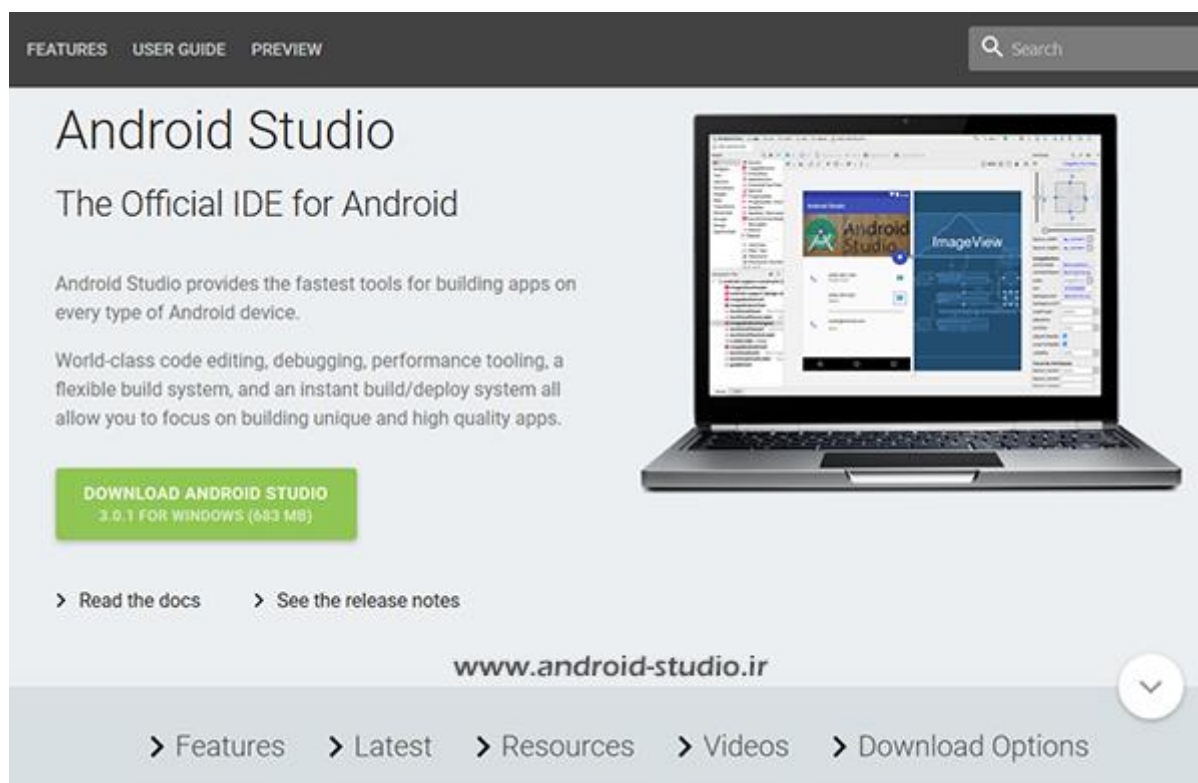
نسخه Bundle که در بالا اشاره شد، مجموع IDE و Android SDK در کنار یکدیگر و در قالب یک فایل نصبی بود که انتشار آن از سوی اندروید متوقف شد.

اندروید استودیو در دو نسخه ۳۲ و ۶۴ بیتی منتشر می شود. با توجه به اینکه در چند سال گذشته اکثر پردازنده (CPU) ها با معماری ۶۴ بیتی تولید و عرضه می شود لذا توصیه من این است اگر از سیستم عامل ۳۲ بیتی استفاده می کنید و رایانه شما از نسخه ۶۴ بیتی پشتیبانی می کند، قبل از شروع کار ابتدا حتماً سیستم عامل خود را تعویض کنید. به طور خلاصه سرعت پردازش در این معماری بالاتر بوده و مواردی همچون محدودیت استفاده از RAM نیز از بین می رود.



دانلود اندروید استودیو در قسمت Developer وب سایت android.com برای سیستم عاملهای ویندوز، لینوکس و مکینتاش فراهم شده است:

<https://developer.android.com/studio/index.html>



با کلیک روی گزینه Download Options لیست فایل‌های دانلودی باز می شود:



FEATURES USER GUIDE PREVIEW

Search

Start using Android Studio today

Android Studio includes all the tools you need to build apps for Android.

DOWNLOAD ANDROID STUDIO
3.0.1 FOR WINDOWS (683 MB)

VERSION: 3.0.1.0

RELEASE DATE: NOVEMBER 20, 2017

Select a different platform

Platform	Android Studio package	Size	SHA-256 checksum
Windows (64-bit)	android-studio-ide-171.4443003-windows.exe	683 MB (716,684,712 bytes)	1f71333fc8f31281b643a12d7f9d4c22e75ee7c14dc1faf00ce3d5291ef40bb0
	android-studio-ide-171.4443003-windows.zip No installer	739 MB (775,207,909 bytes)	b9f73dd6d2159f226c5e507275c7a5e7cc38106fa0ee1189286675ccfc4a330e
Windows (32-bit)	android-studio-ide-171.4443003-windows32.zip No installer	738 MB (774,678,163 bytes)	cd644af01126a9d8cf372ada1f520579d443946e5cfa99e42234d539db59f842
Mac	android-studio-	738 MB	2c8a24ad2de21e8d1d77a748a35cc5ca47812d9f5e2a0865981e206073d1d981

www.android-studio.ir

FEATURES USER GUIDE PREVIEW

Search

Get just the command line tools

If you do not need Android Studio, you can download the basic Android command line tools below. You can use the included [sdkmanager](#) to download other SDK packages.

These tools are included in Android Studio.

Platform	SDK tools package	Size	SHA-256 checksum
Windows	sdk-tools-windows-3859397.zip	132 MB (138,449,982 bytes)	7f6037d3a7d6789b4fdc06ee7af041e071e9860c51f66f7a4eb5913df9871fd2
Mac	sdk-tools-darwin-3859397.zip	82 MB (86,182,133 bytes)	4a81754a760fce88cba74d69c364b05b31c53d57b26f9f82355c61d5fe4b9df9
Linux	sdk-tools-linux-3859397.zip	130 MB (136,964,098 bytes)	444e22ce8ca0f67353bda4b85175ed3731cae3ffa695ca18119cbacef1c1bea0

See the [SDK tools release notes](#).

www.android-studio.ir

متأسفانه گوگل دسترسی به بخش Developer را بر روی IP های ایران مسدود کرده لذا برای دانلود نیاز به تغییر آی پی دارید و یا راه حل دوم که دانلود از وب سایت های ثانویه (مانند [وب سایت ما](#)) می باشد.



در نسخه ۳ اندروید استودیو، پروسه نصب به اینصورت است که بعد از نصب IDE، ابزار موردنیاز SDK به صورت آنلاین و خودکار دانلود می شود. مزیت این شیوه در این است که با اتصال اندروید استودیو به سرورهای گوگل، آخرین نسخه از ابزار SDK دانلود و نصب می شوند. اما باز هم مسئله محدودیت دسترسی برای کاربران ایرانی مشکل ساز می شود. در اینجا دو راه حل پیش رو داریم:

۱- استفاده از روش های تغییر IP جهت دور زدن تحریم و دریافت ابزار به صورت خودکار و آنلاین. به لحاظ سهولت در فرآیند راه اندازی محیط اندروید استودیو و SDK، این روش زمان کمتری را از توسعه دهنده می گیرد و نصب تک تک ابزار موردنیاز در SDK بطور خودکار انجام می شود. اما در نظر داشته باشید ابزار تغییر آی پی شما باید از سرعت بالایی برخوردار باشد زیرا حدود ۱/۵ گیگابایت مجموع فایل هایی است که باید دریافت شود.

۲- دانلود ابزار SDK به صورت غیر مستقیم و ساخت دایرکتوری SDK به صورت آفلاین و دستی و سپس نصب اندروید استودیو و معرفی محل قرارگیری SDK به آن.

دو روش ذکر شده هرکدام مزایای مختص خود را داراست اما با توجه حجم نسبتاً زیاد ابزار و سرعت پایین اینترنت کاربران داخل ایران (به خصوص در زمان استفاده از ابزار تغییر IP) پیشنهاد بنده، استفاده از روش دوم است. این شیوه ممکن است برای یکبار ۱۰ دقیقه از وقت شما را بگیرد اما در عوض با ساختار و جزئیات ابزار موجود در SDK بیشتر آشنا خواهید شد که برای هر توسعه دهنده یک امتیاز محسوب می شود.

ابتدا روش نصب آفلاین را بررسی می کنیم. سپس توضیحات مربوط به نصب آنلاین برای آن دسته از کاربرانی که مایل به نصب خودکار و آنلاین SDK هستند را نیز ارائه خواهم داد.

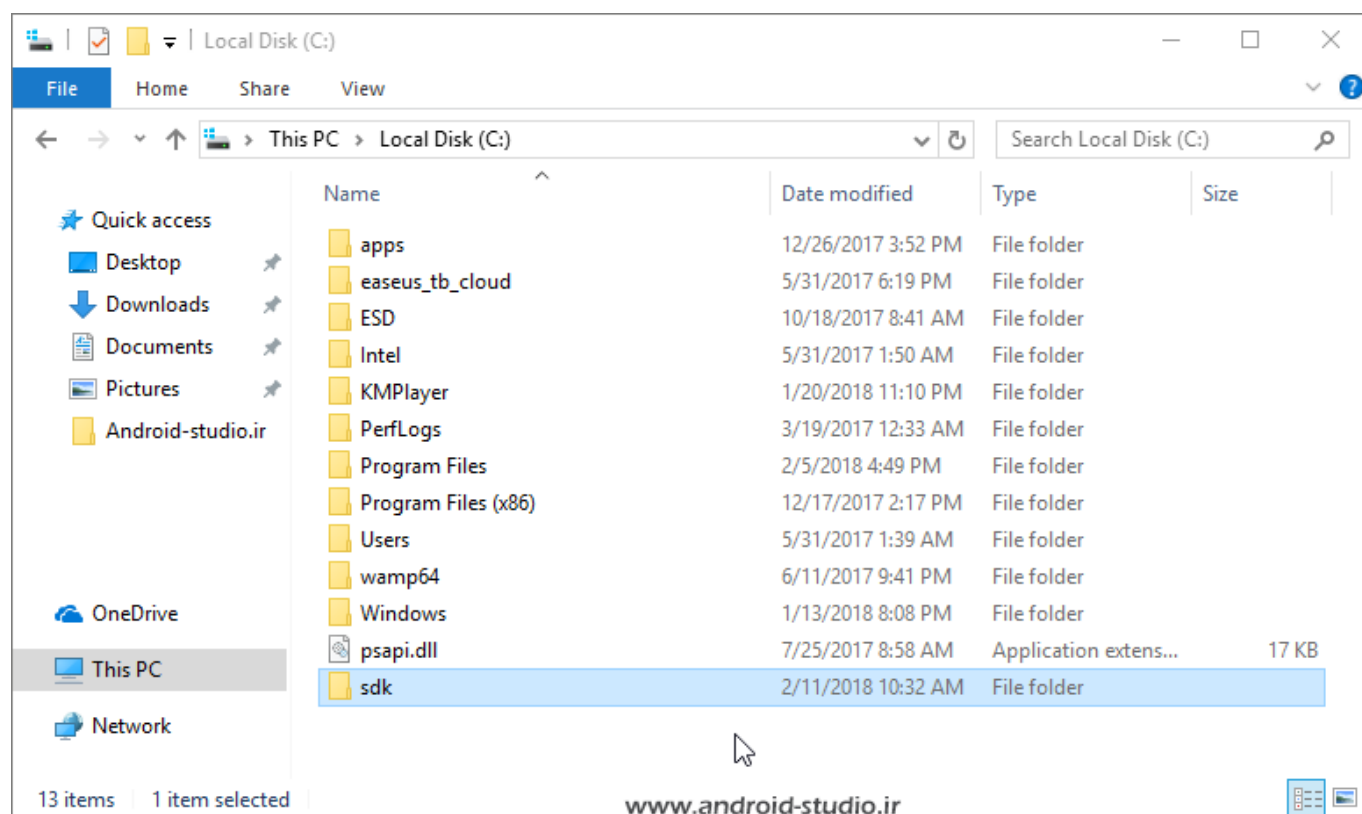
نکته: این مبحث و هر دو روش نصب آنلاین و آفلاین را به طور کامل مطالعه کنید.



« نصب آفلاین SDK »

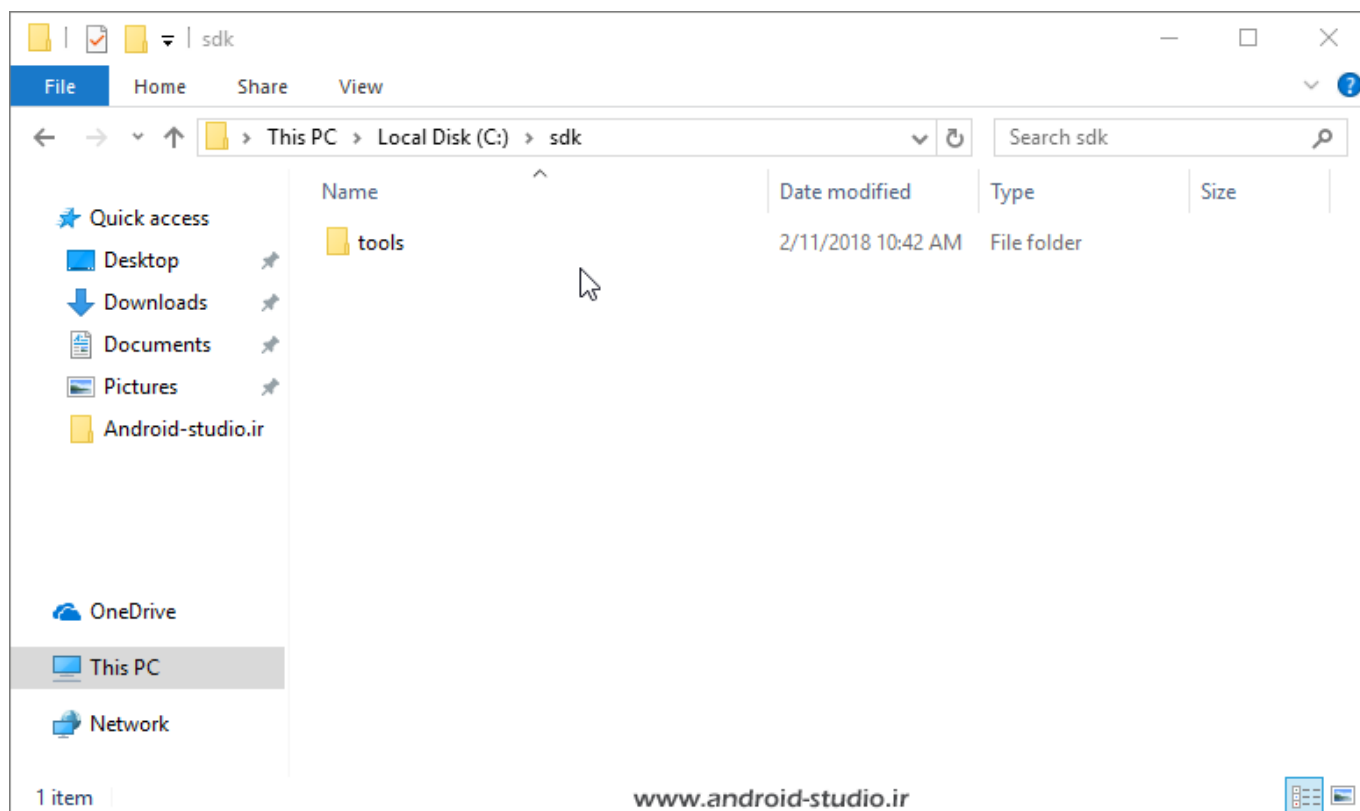
قبل از نصب اندروید استودیو، ابتدا SDK را به همراه حداقل ابزار ضروری می سازم تا هنگام معرفی محل قرارگیری SDK در اندروید استودیو، با اخطار عدم شناسایی SDK مواجه نشوم.

یک فولدر با نام sdk داخل درایو C ایجاد می کنم:



نام و محل قرارگیری فولدر به دلخواه شماست.

✓ نصب SDK Tools: اولین ابزاری که به sdk اضافه می کنم Tools است. فایل sdk-tools-windows-26.1.1.zip را از حالت فشرده خارج کرده و فولدر خروجی را به فولدر sdk منتقل می کنم:



توجه داشته باشید اگر سیستم عامل شما ویندوز است و sdk را در درایوی که ویندوز نصب شده ایجاد کرده اید، فایل‌های فشرده را ابتدا در محل دیگری (مانند دسکتاپ) Extract کرده و سپس به مسیر مدنظر منتقل کنید.

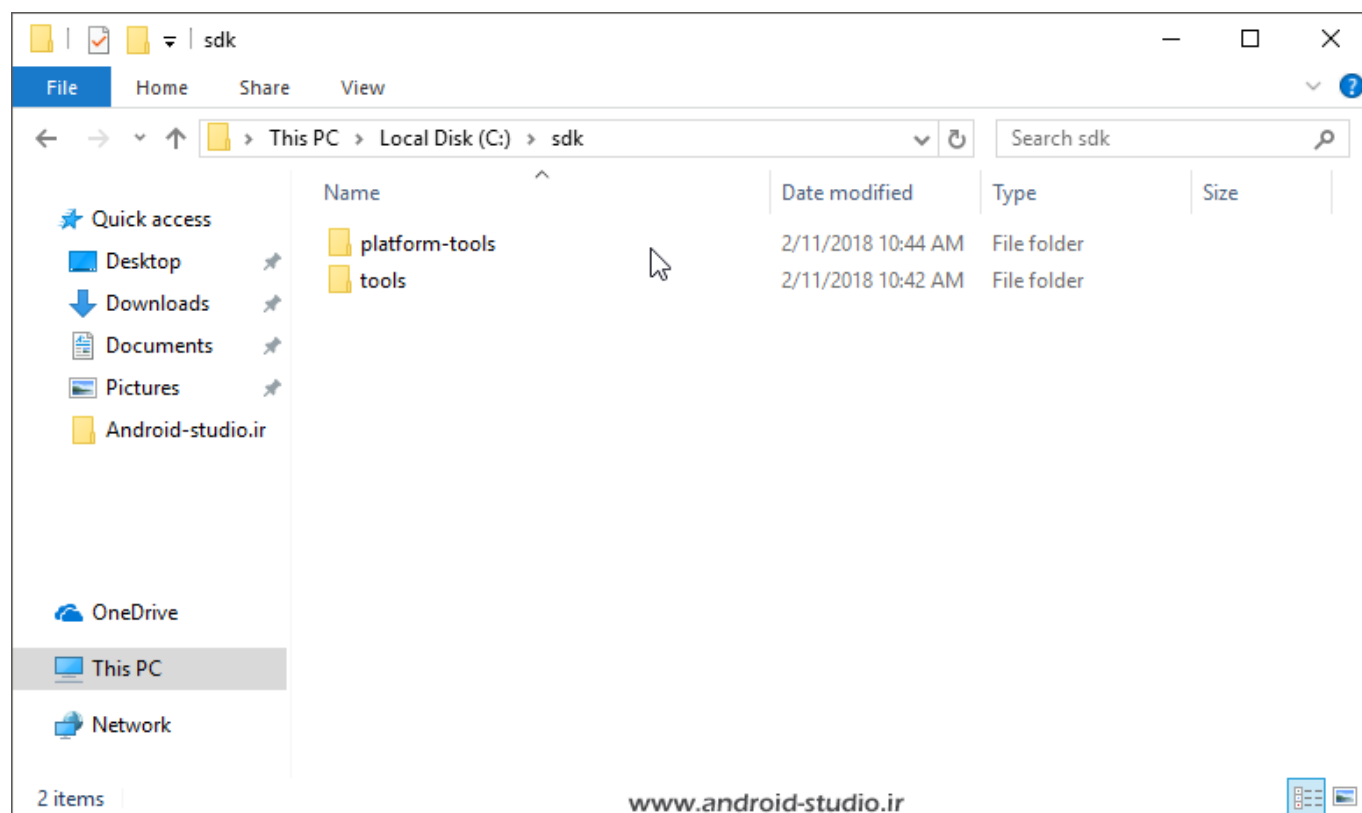
نکته: عدد درج شده در انتهای نام فایل ابزار معرف نسخه آن است. به عنوان مثال فایل sdk-tools-windows-26.1.1.zip

مربوط به SDK Tools نسخه 26.1.1 است که به مرور زمان نسخه های جدید منتشر می شود. در وب سایت ما نیز هر چند مدت یکبار لینک دانلود ابزار بروز شده و نسخه های جدید در اختیار کاربران قرار می گیرد.

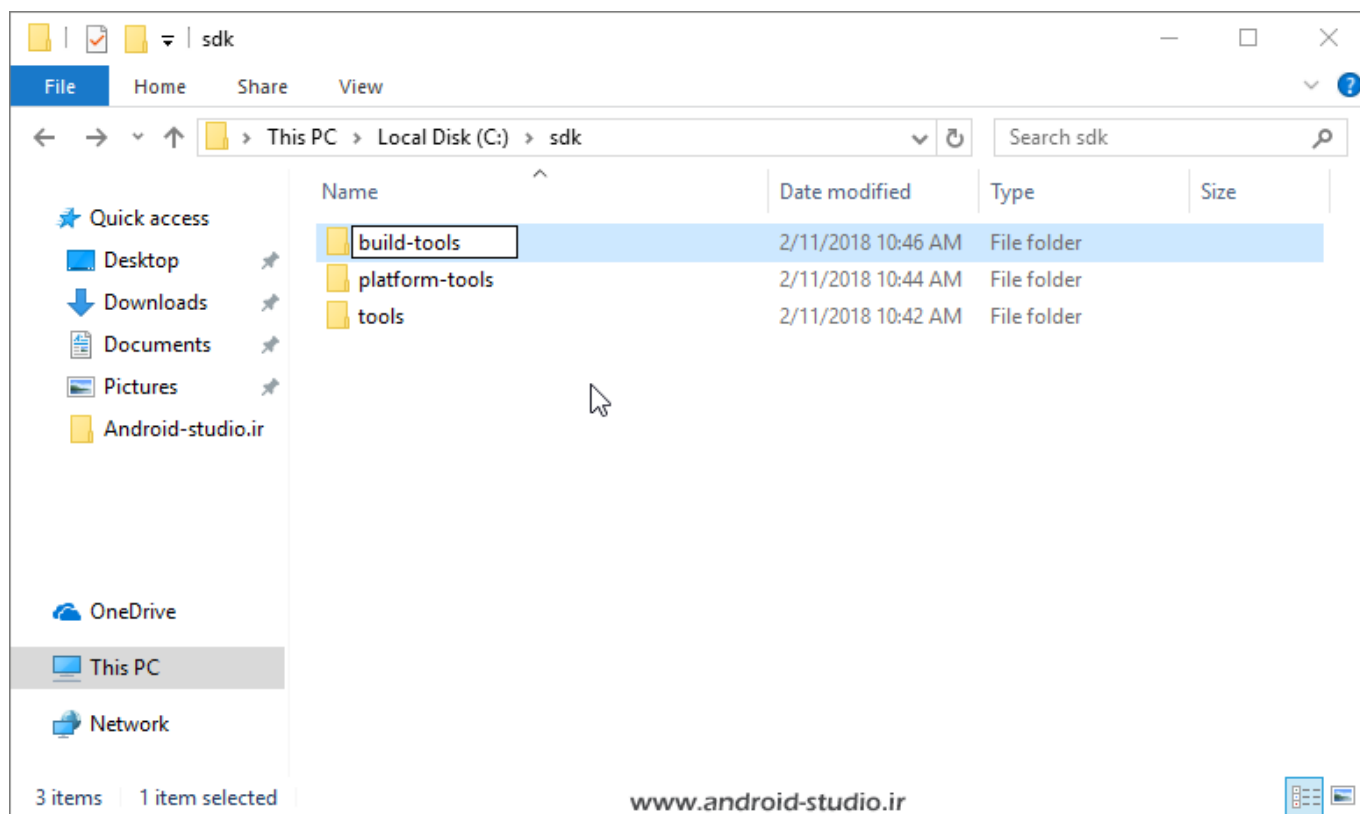
نکته: منظور از نصب ابزار، صرفا قرارگیری فولدر در مکان تعیین شده است و مانند نصب نرم افزارها نیاز به اجرای یک فایل Setup نیست (بجز یک مورد که اشاره خواهد شد).



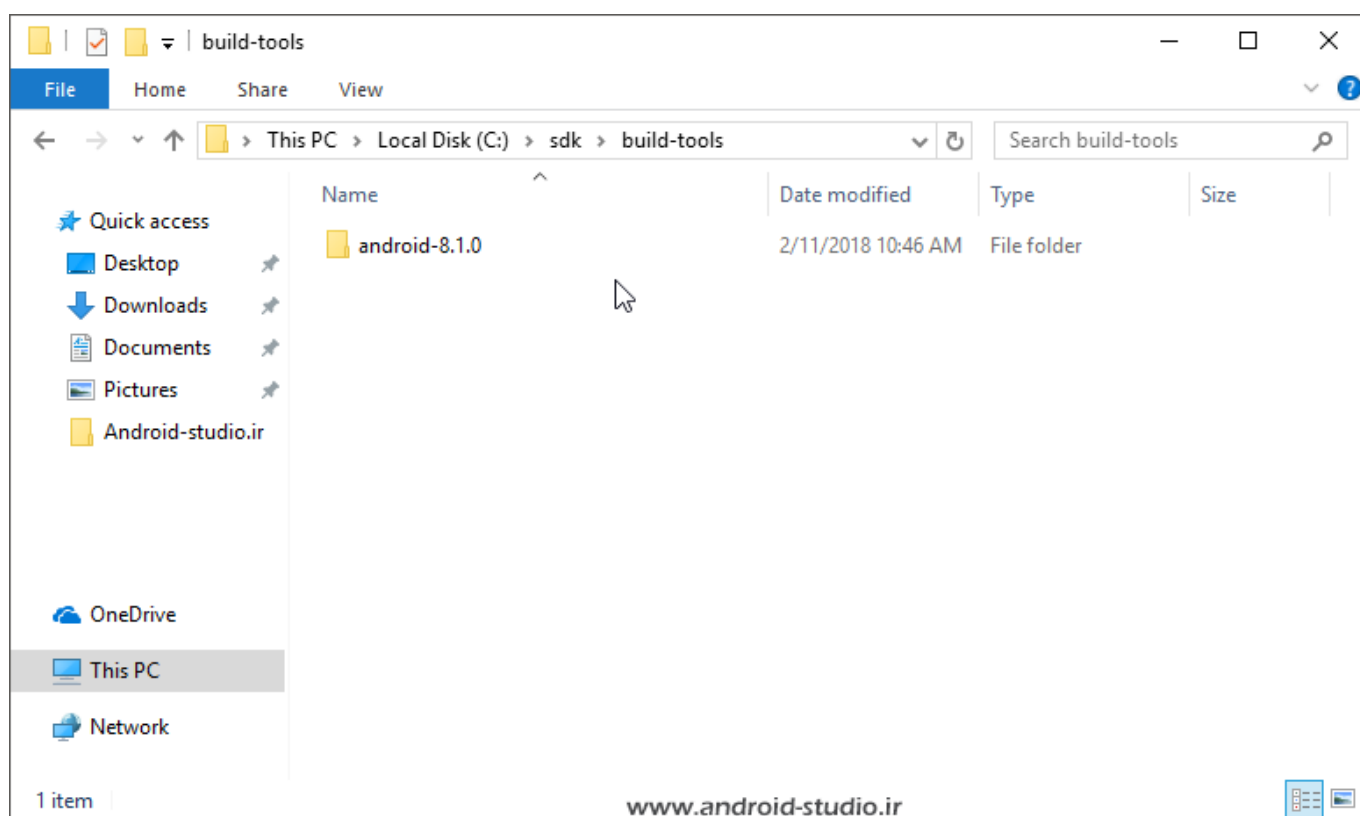
✓ Platform Tools: فایل platform-tools_r27.0.1-windows.zip را از حالت فشرده خارج کرده و فولدر خروجی را به sdk اضافه می کنیم:



✓ Build Tools: ابتدا یک فولدر با نام build-tools به sdk اضافه می کنیم:



سپس فایل build-tools_r26.0.2-windows.zip را از حالت فشرده خارج کرده و فولدر خروجی را درون این فولدر قرار می دهیم:

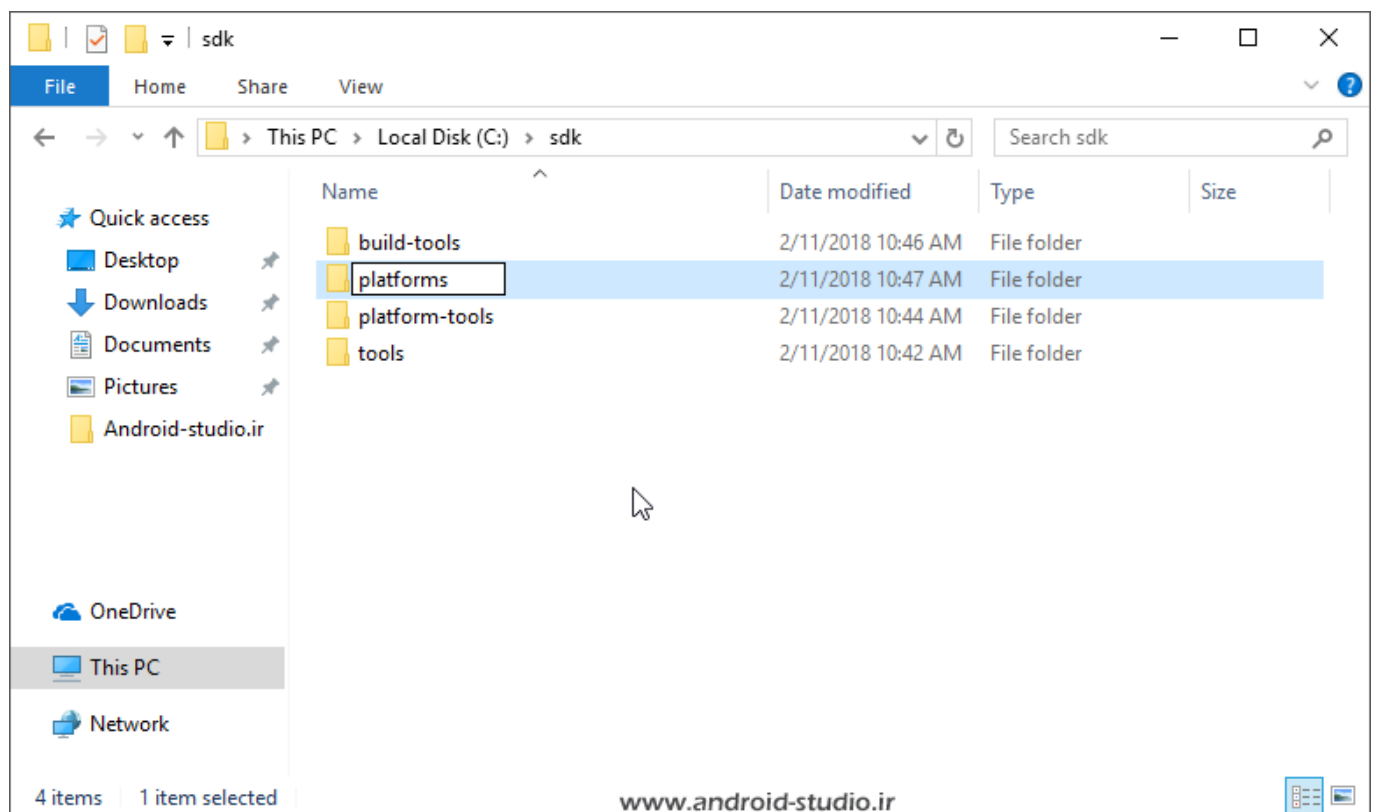




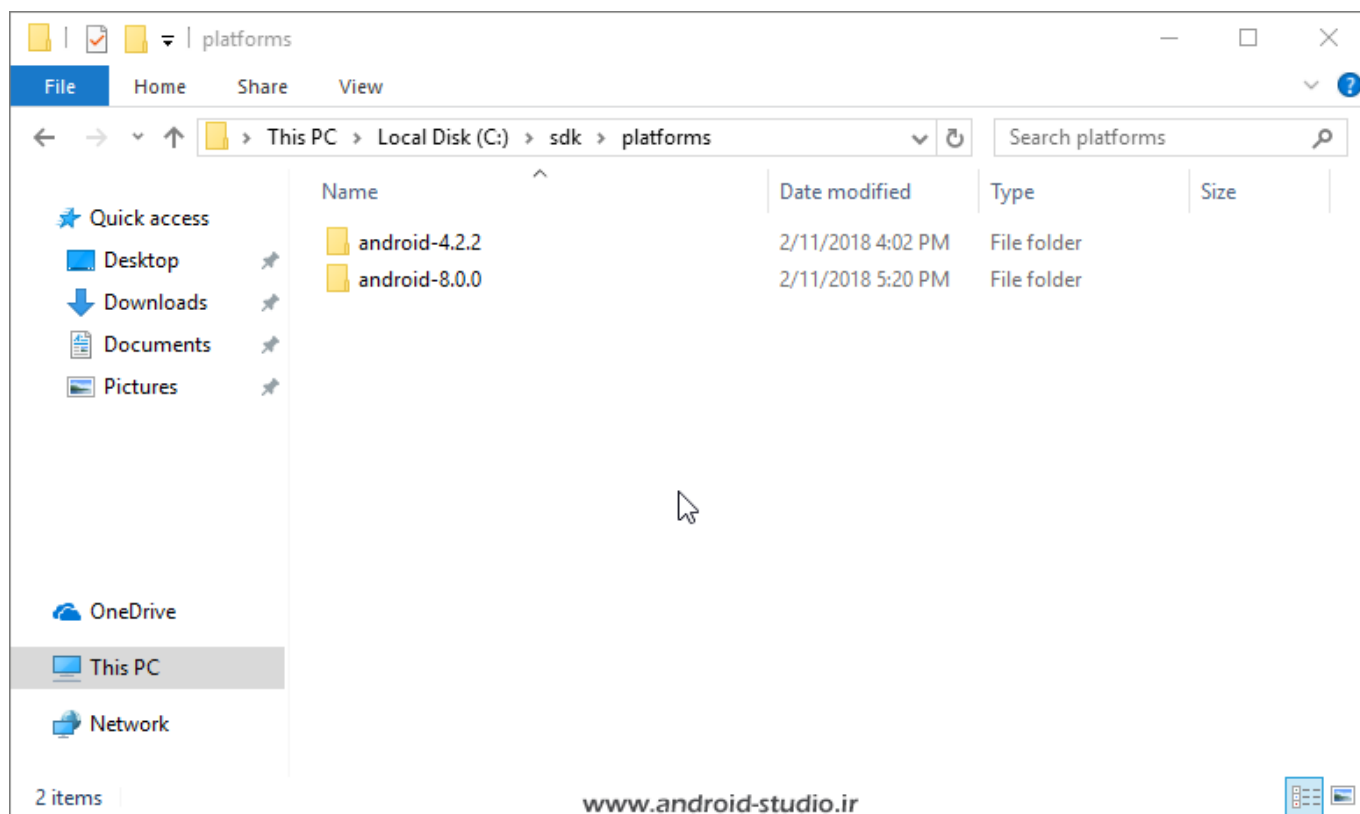
احتمالا این سوال در ذهن شما بوجود آمده که چرا SDK Tools و Platform Tools مستقیما داخل فولدر sdk قرار گرفت اما Build Tools باید داخل یک فولدر فرعی اضافه شود؟

برخی از ابزار مانند دو مورد اول، در آن واحد فقط یک نسخه از آنها قابل استفاده است. به عنوان مثال توسعه دهنده نمی تواند همزمان دو نسخه PlatformTools 26.0.2 و PlatformTools 27.0.1 را در sdk داشته باشد و استفاده کند و هنگام نیاز به بروزرسانی آن لازم است ابتدا فولدر فعلی platform-tools از sdk به طور کامل حذف شده سپس نسخه جدید جایگزین شود. اما مواردی مانند Build Tools برنامه نویس می تواند چندین نسخه متفاوت از آن را در sdk داشته باشد و بنا به نیاز خود از همه آنها در پروژه های مختلف استفاده کند. یعنی می توان BuildTools 26.0.2 و BuildTools 27.0.3 را در کنار هم در فولدر build-tools قرار داد بدون آنکه نیاز به حذف نسخه قبل باشد.

✓ Platforms: یک فولدر با نام platforms به sdk اضافه می کنم:



مانند مورد قبل، در فولدر platforms نیز می توان چندین API مختلف اضافه کرد. به عنوان مثال من android-17_r03.zip و platform-26_r02.zip را از حالت فشرده خارج کرده و به فولدر platforms اضافه می کنم:

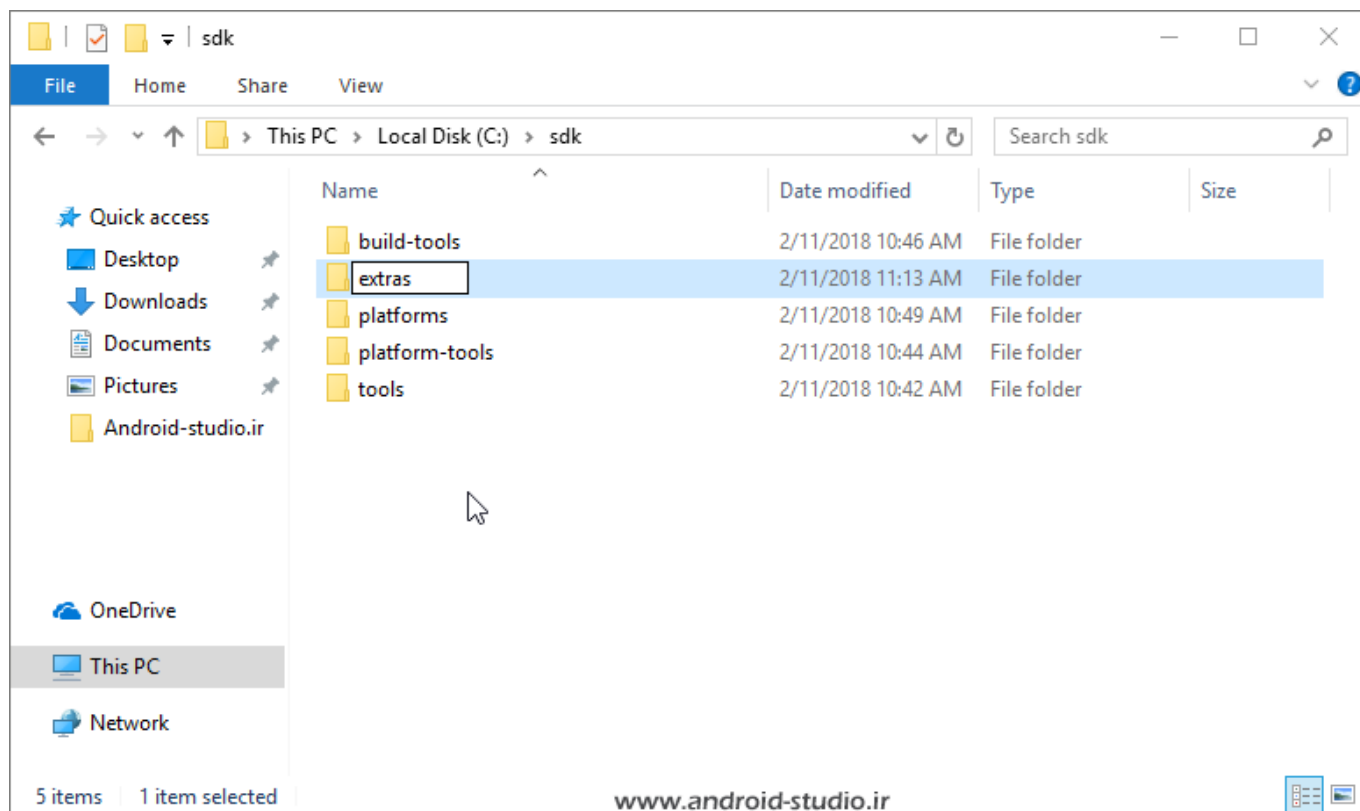


برای هر نسخه از اندروید یک API منتشر می شود. به عنوان مثال platform-26_r02 مربوط به API 26 (Android 8.0 Oreo) می باشد. عدد r02 مربوط به نسخه منتشر شده از این API است. گوگل معمولا برای هر API (تا قبل از معرفی نسخه بعدی اندروید) دو یا سه بروزرسانی منتشر می کند. بنابراین اگر platform-26_r03 معرفی شد بهتر است فولدر نسخه فعلی را از sdk حذف کرده و نسخه جدید را جایگزین کنید. توسعه دهنده بر حسب نیاز خود ممکن است از API های مختلفی استفاده کند. توضیحات بیشتر در مباحث آتی ارائه می گردد.

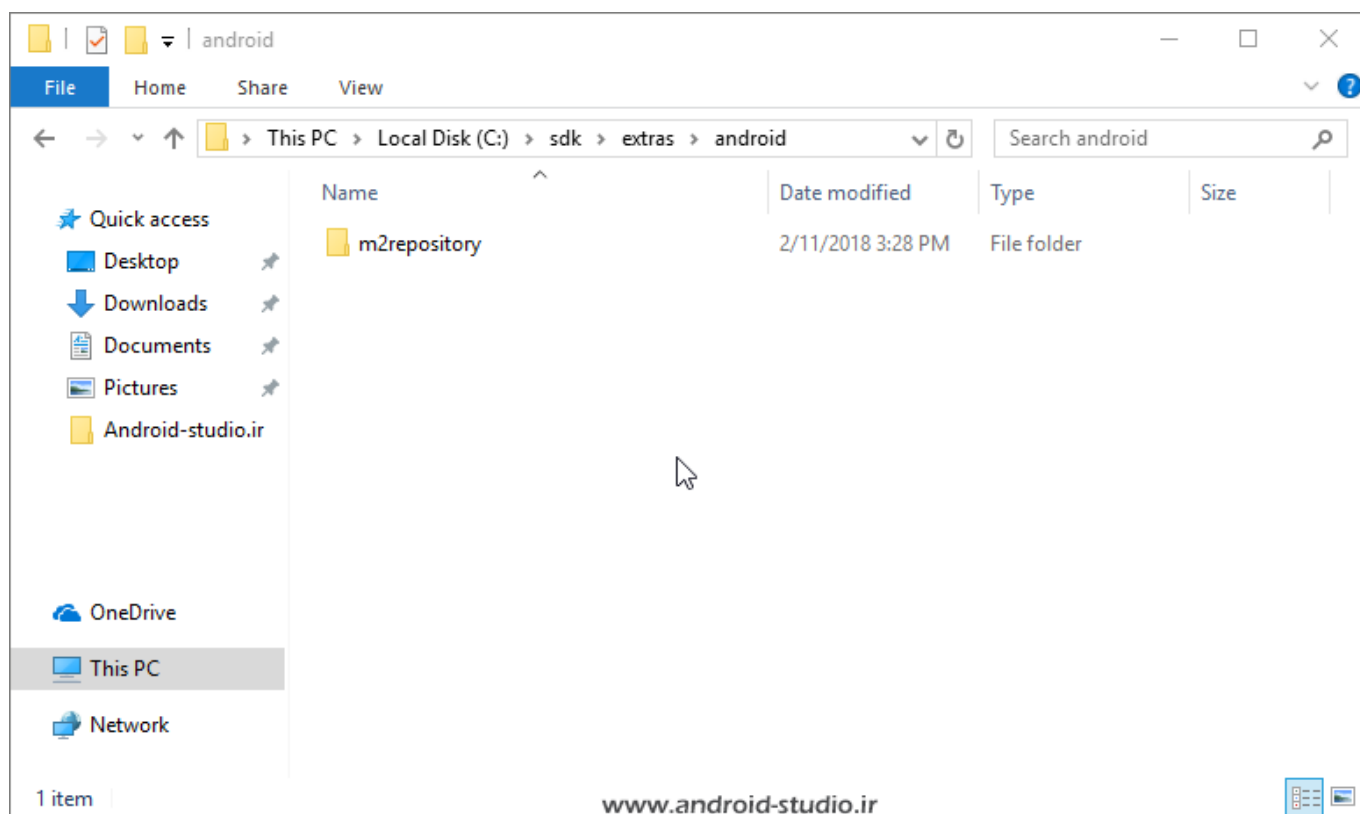
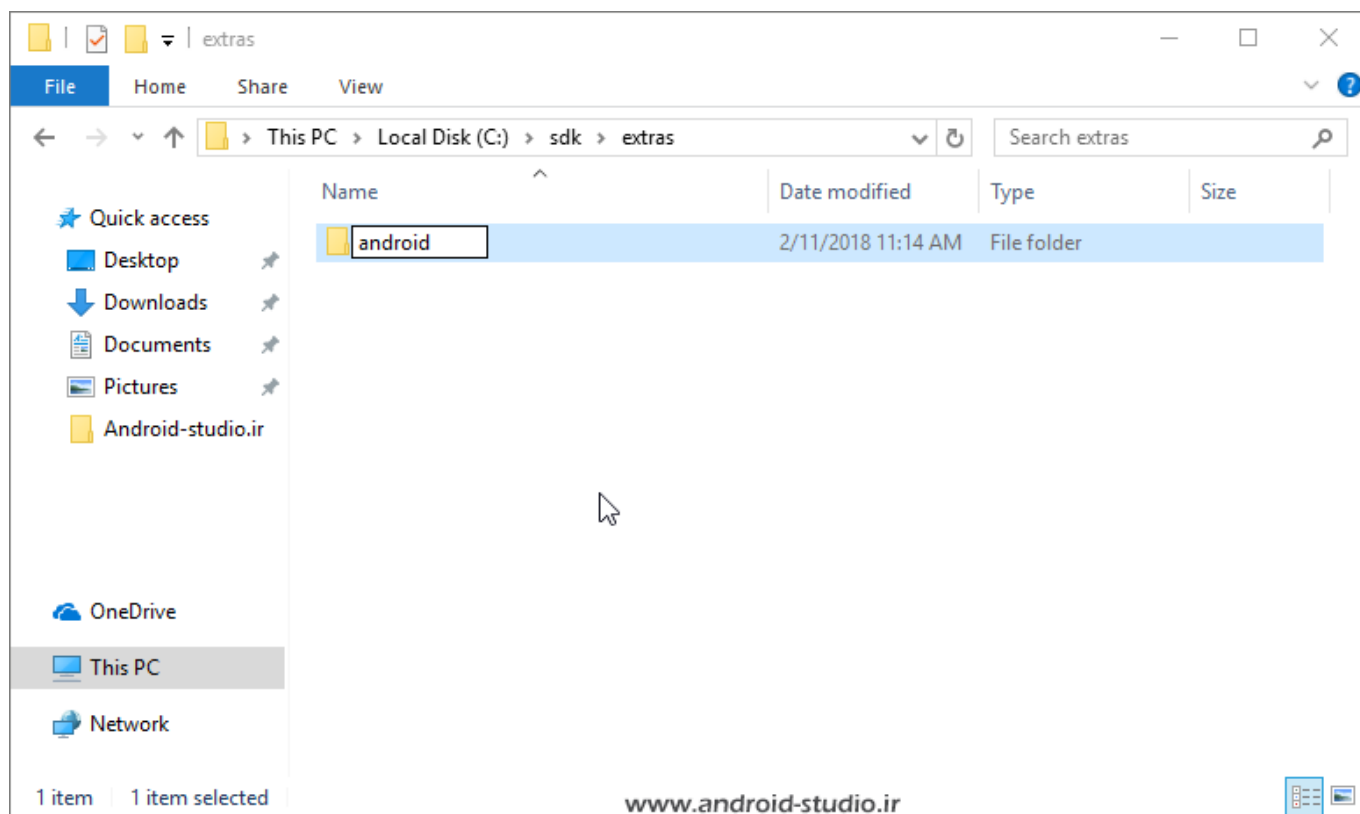
نکته: اینکه نام فایل تعدادی از API ها با android و تعدادی با platform شروع شده در ماهیت آن تفاوتی ایجاد نمی کند و صرفا یک نامگذاری از طرف اندروید است که من ترجیح دادم فایلها را بدون تغییر نام در اختیار شما بگذارم.



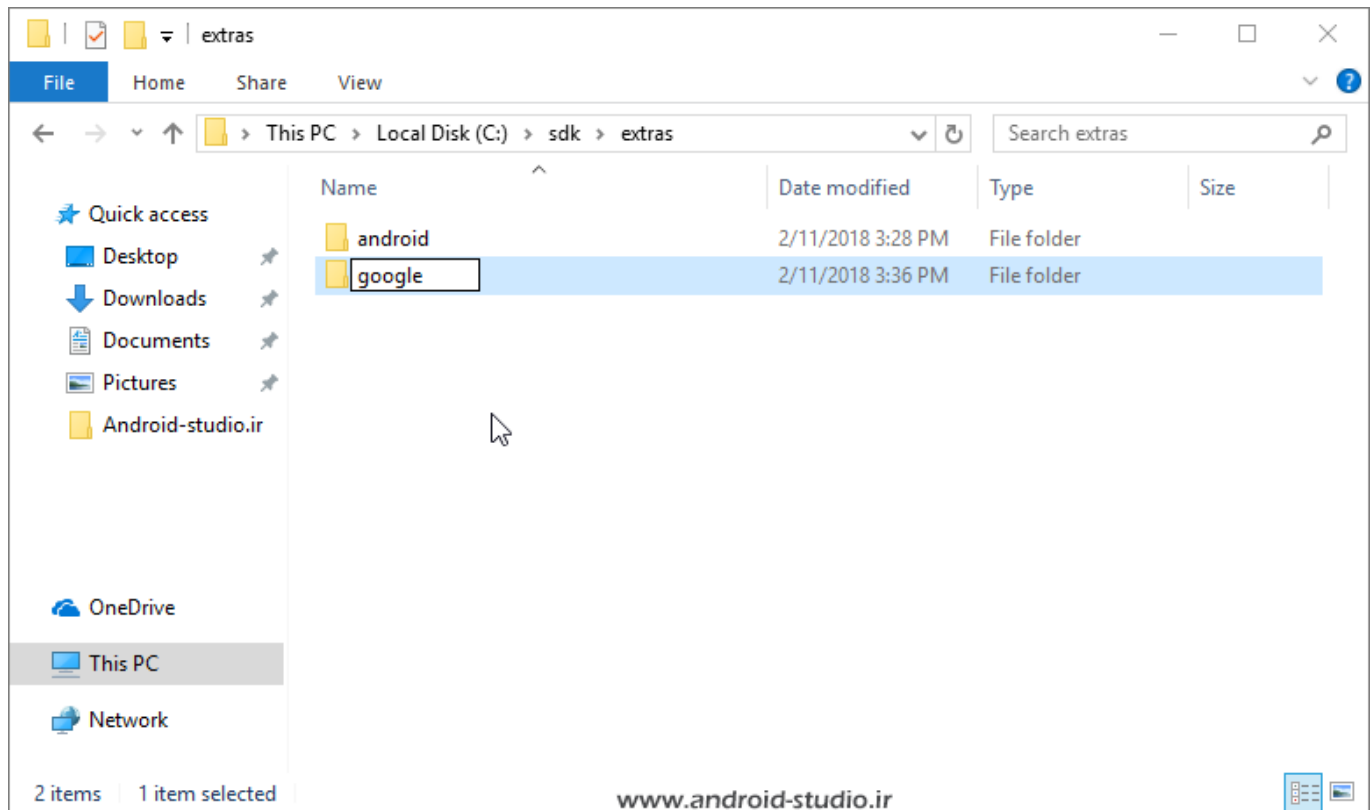
✓ Extras: تعداد دیگری از ابزار را معرفی می کنم که همگی درون فولدر extras قرار می گیرند:



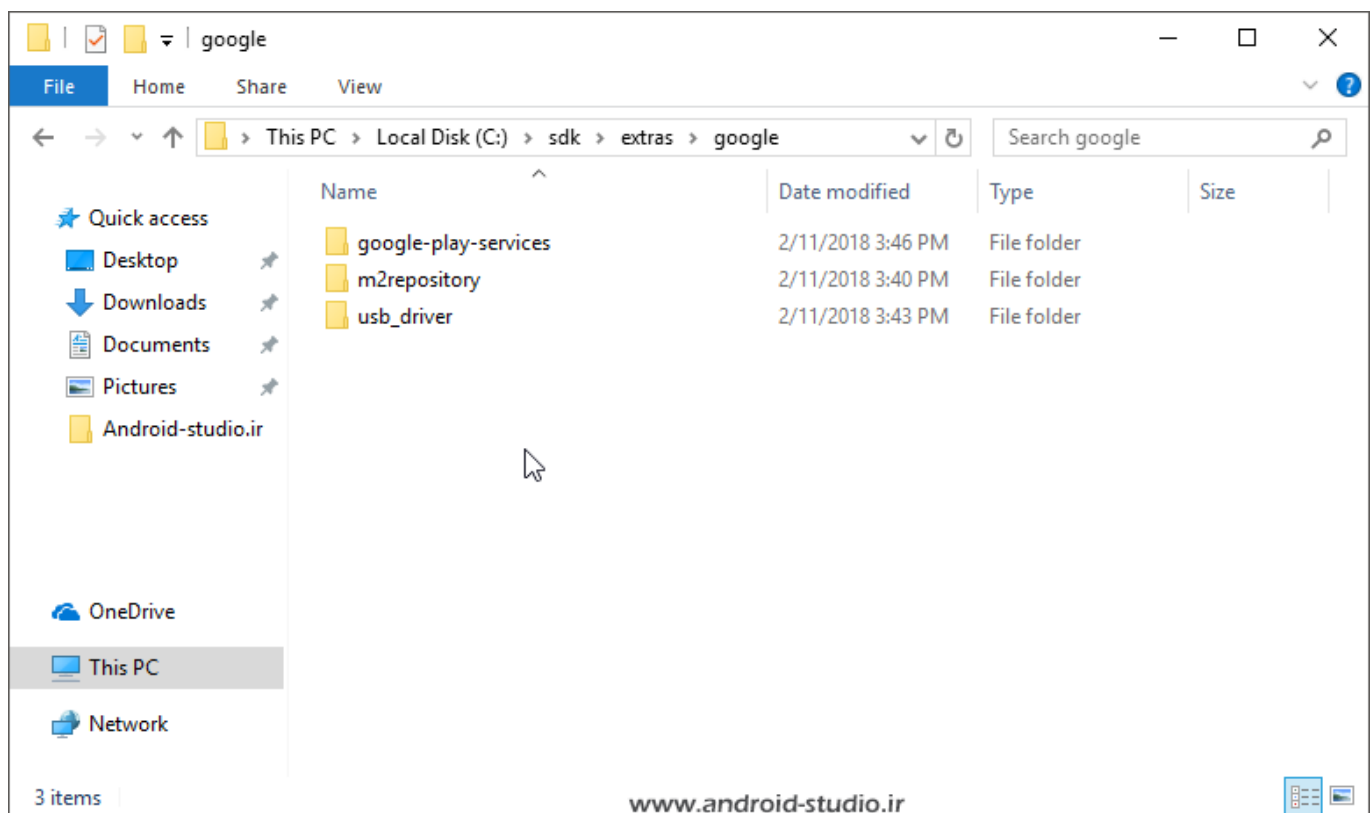
داخل extras یک فولدر دیگر با نام android ایجاد کرده، سپس خروجی android_m2repository_r47.zip را به این فولدر منتقل می کنم:



مجدد داخل extras یک فولدر دیگر با نام google ایجاد می کنم:

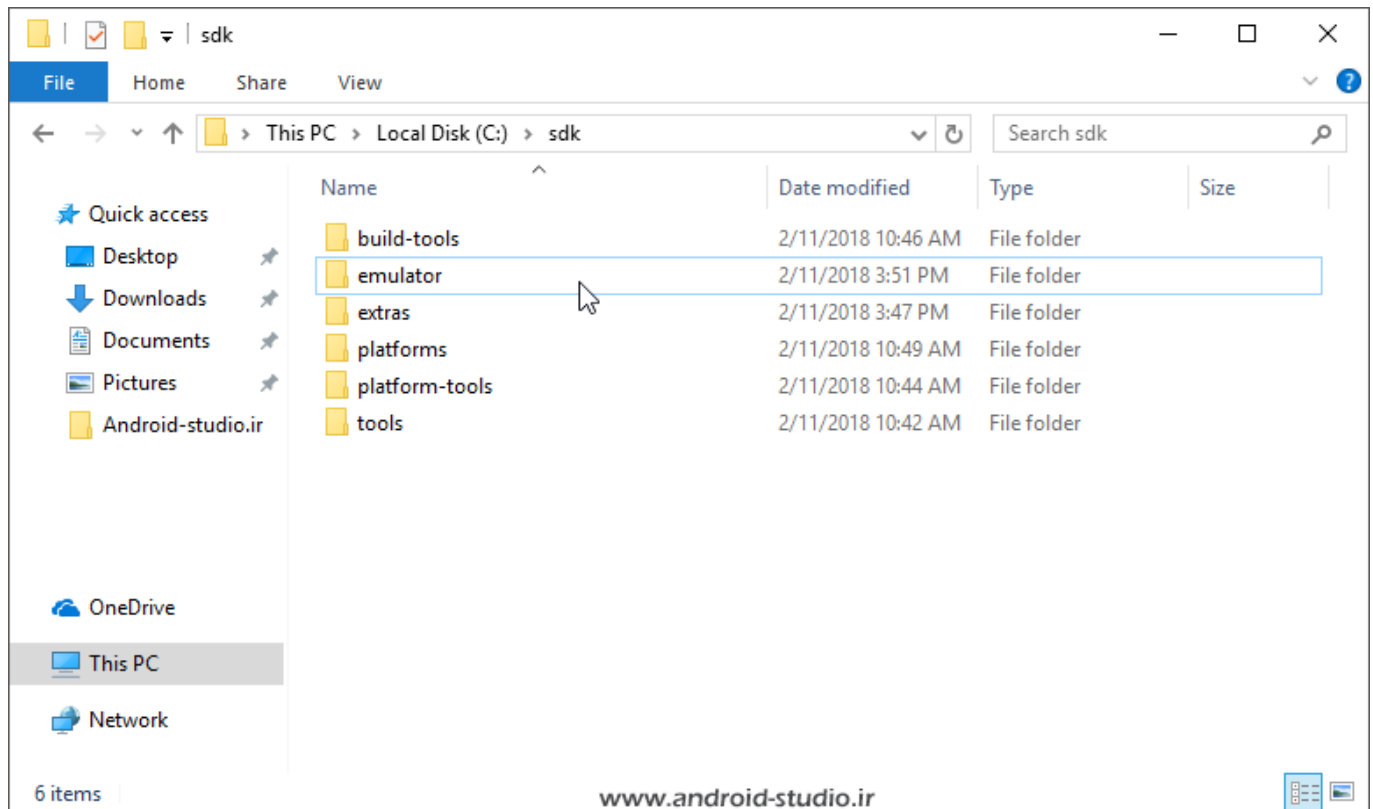


سپس ابزارهای `usb_driver_r11-windows.zip`، `google_m2repository_58.zip` و `google_play_services_44.zip` را به این فولدر منتقل می‌کنم:





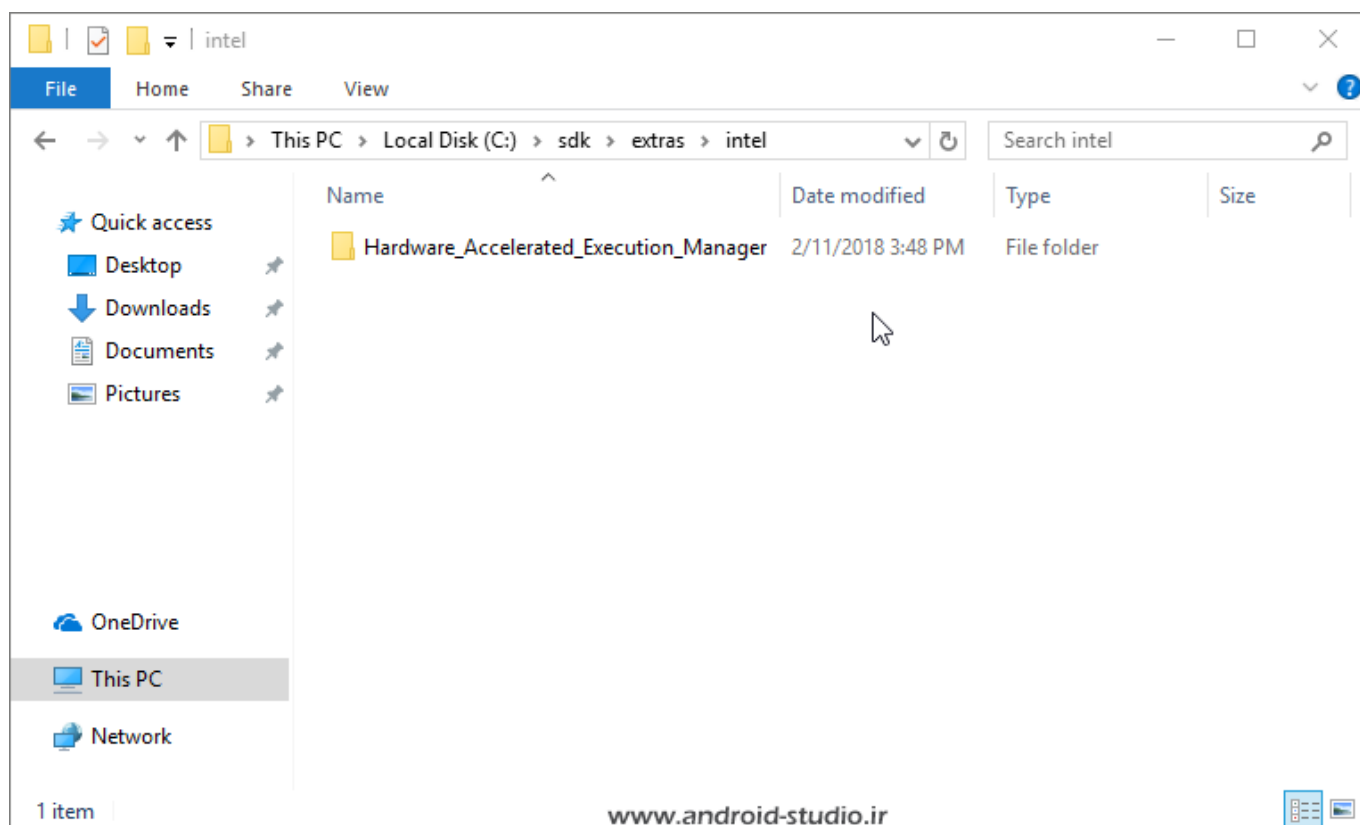
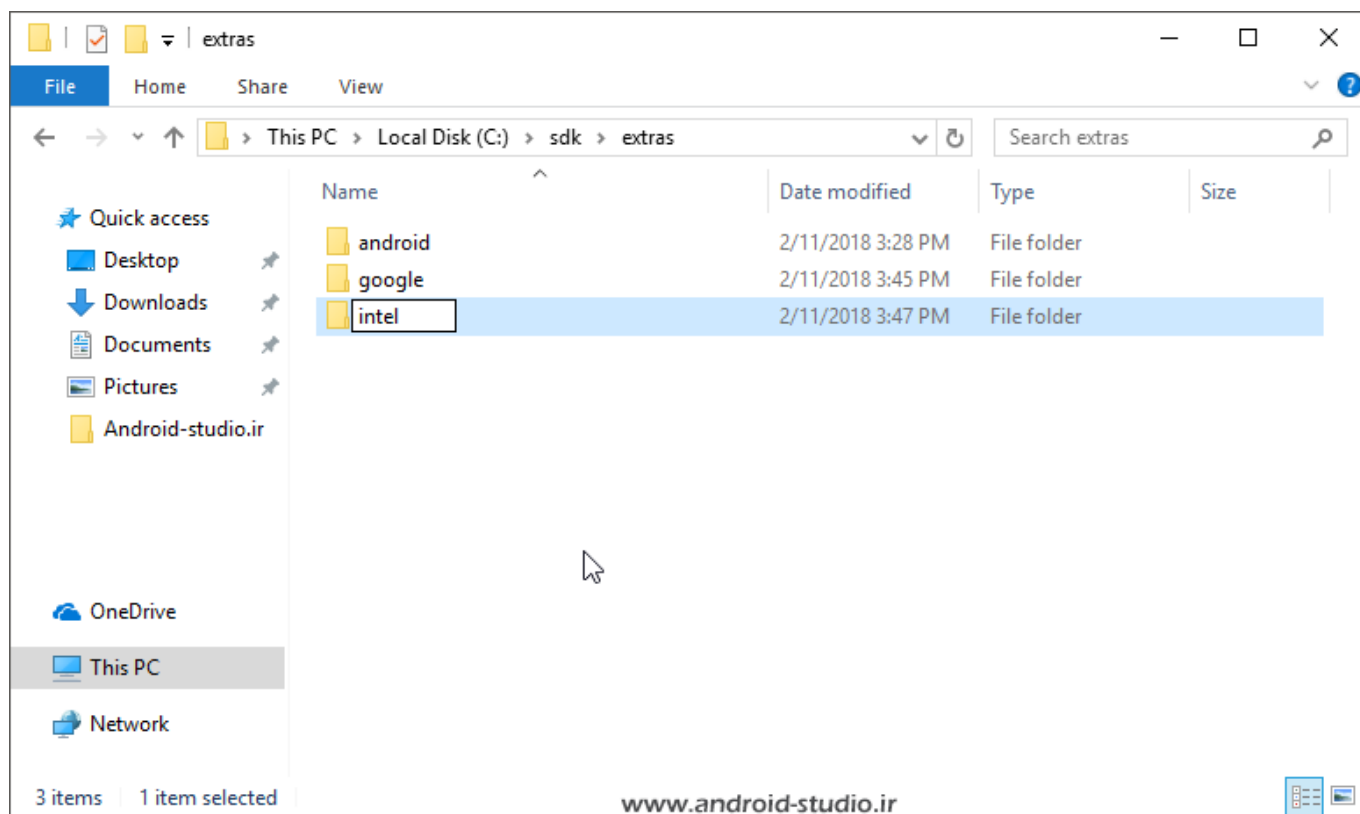
✓ Emulator: امولاتور یا شبیه سازها، یک محیط شبیه سازی شده اندرویدی است که روی رایانه اجرا شده و توسعه دهنده را از تست و خطایابی پروژه های خود روی دیوایس های حقیقی مانند موبایل و تبلت بی نیاز می کند. شبیه ساز های زیادی برای اندروید ساخته شده اند. از جمله AVD که به همراه اندروید استودیو نصب می شود. اگر مایل به استفاده از شبیه ساز پیش فرض اندروید استودیو یعنی AVD هستید، لازم است emulator-windows-27.0.5.zip را از حالت فشرده خارج کرده، به sdk منتقل کنید:

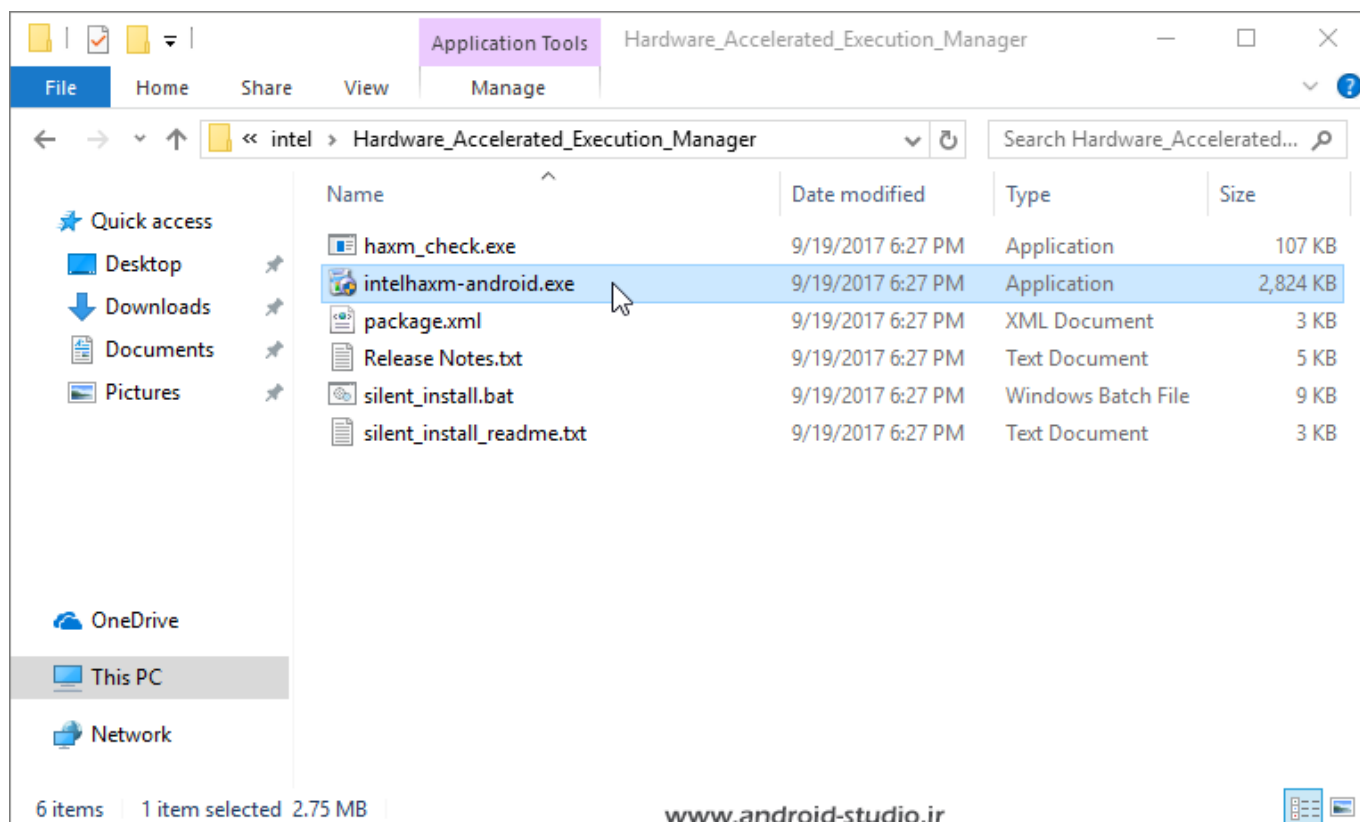


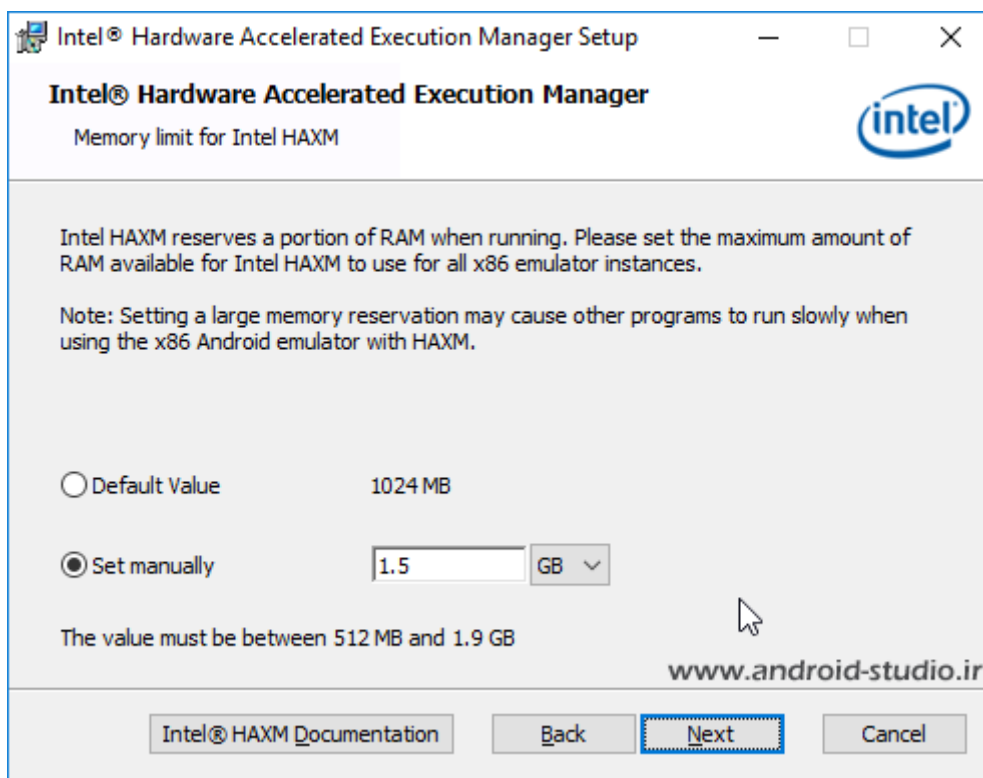
اگر پردازنده رایانه شما Intel است و قصد استفاده از شبیه ساز (AVD) Android Virtual Device را دارید، برای افزایش سرعت شبیه ساز لازم است

Intel Hardware Accelerated Execution Manager (Intel® HAXM)

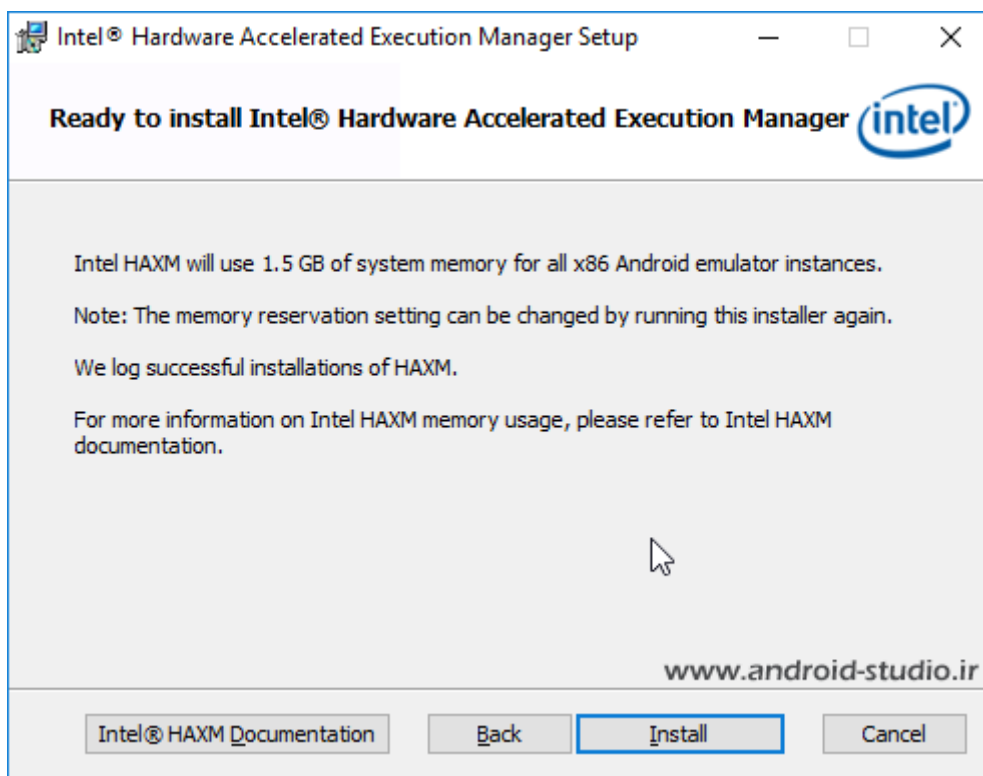
را نیز نصب کنید. برخلاف موارد پیشین، این ابزار علاوه بر قرارگیری در فولدر مخصوص به خود، دارای فایل Setup بوده که باید اجرا و نصب گردد. داخل extras یک فولدر با نام intel ایجاد کرده و خروجی فایل Hardware_Accelerated_Execution_Manager_6.2.1.zip را به این فولدر منتقل می کنیم:

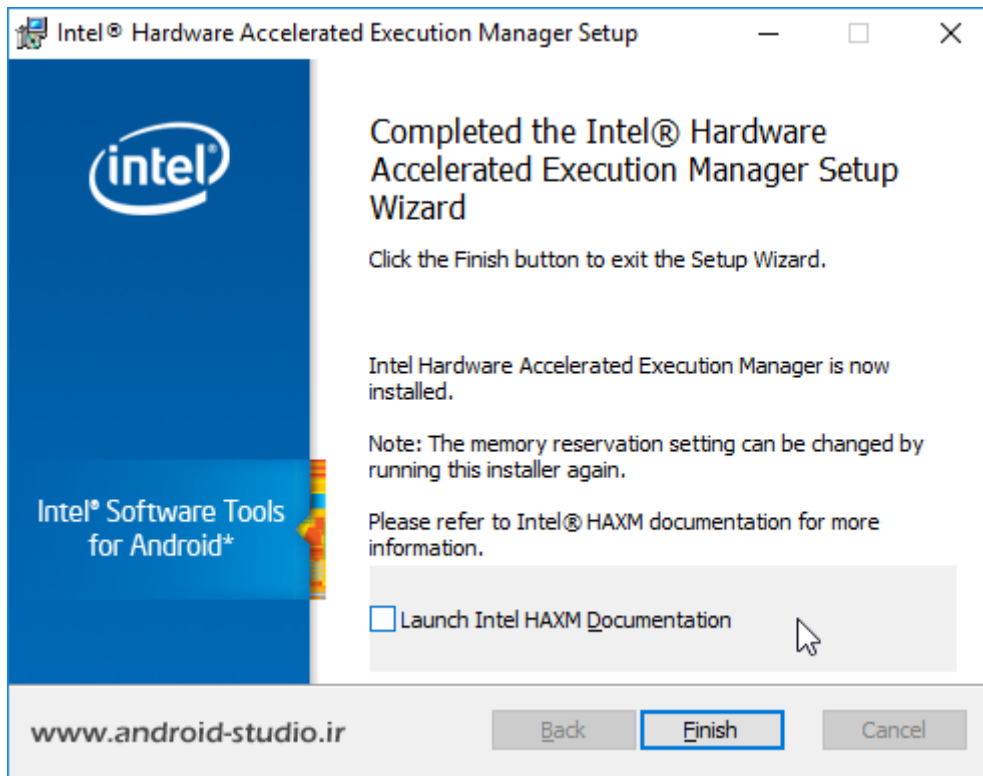






در این مرحله میزانی از حافظه RAM رایانه که مایلید در هنگام اجرای شبیه ساز AVD به آن اختصاص داده شود را تعیین کنید.





نکته: با اینکه شبیه ساز AVD نسبت به گذشته تغییراتی را به همراه داشته و گفته شده در نسخه های جدید از سرعت بیشتری برخوردار است، با اینحال اکثر توسعه دهندگان از شبیه سازهای دیگری مانند Genymotion استفاده می کنند که در مباحث آتی معرفی خواهد شد. بنابراین اگر فعلا قصد استفاده از AVD را ندارید، نیازی به نصب Emulator و HAXM نیست.

خب! تا اینجا کار موفق شدم یک SDK با حداقل ابزار ضروری را آماده و تکمیل کنم. بعدا در هر زمان که لازم بود ابزاری را به SDK اضافه یا حذف یا بروزرسانی کنم، چه بصورت آفلاین و مانند مراحل بالا چه بصورت آنلاین و از طریق SDK Manager اندروید استودیو، امکانپذیر خواهد بود.

احتمالا از من می پرسید چرا این فولدر sdk را به صورت کامل شده و در قالب یک فایل فشرده برای دانلود روی سایت قرار ندادم تا زحمت شما کم شود؟

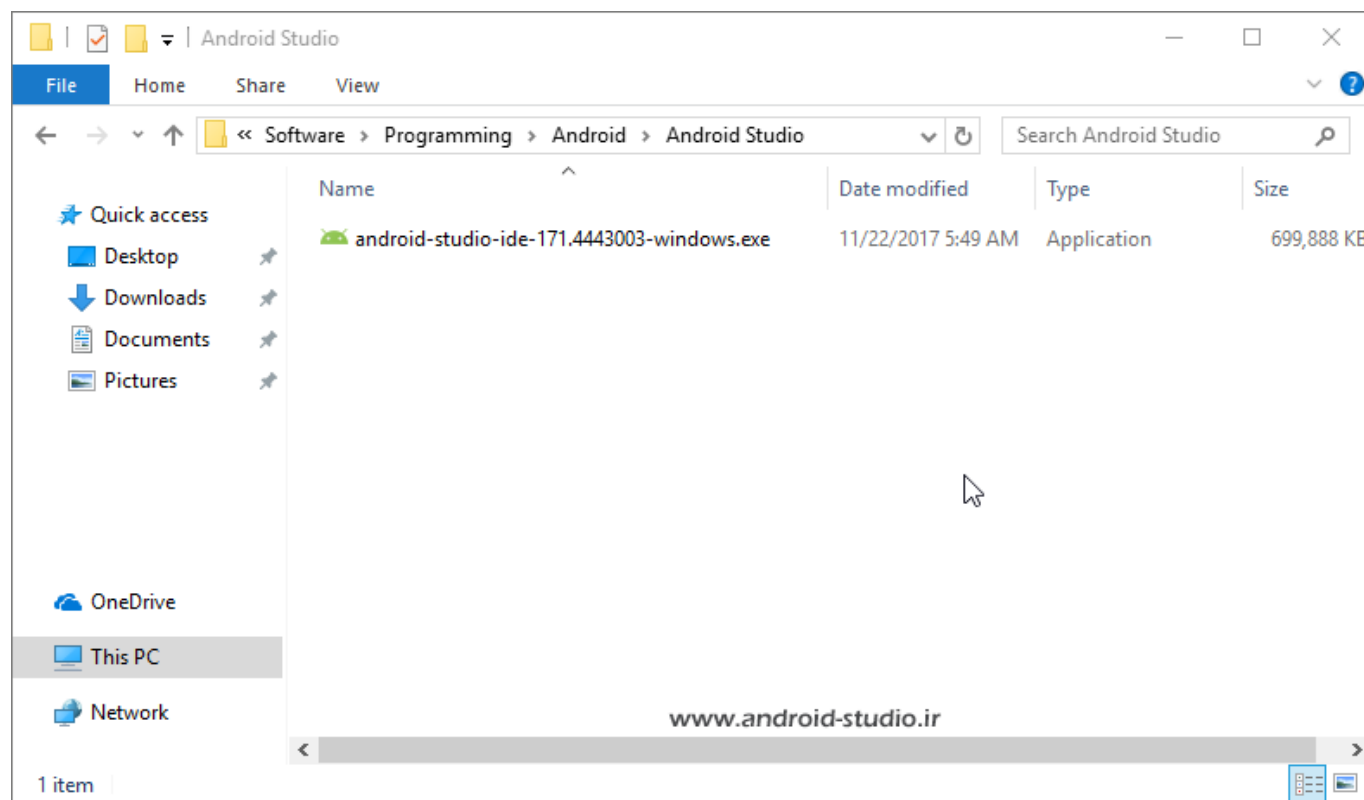
در پاسخ تاکید میکنم هدفم آشنایی شما با محتوای داخل SDK بود تا در مدیریت آفلاین ابزارها (اضافه و حذف ابزار یا بروزرسانی) مشکلی نداشته باشید. حالا اگر قصد داشتید روی یک رایانه دیگر هم از اندروید استودیو استفاده کنید، کافیست همین فولدر sdk کامل شده را به سیستم دیگر منتقل کرده و اندروید استودیو را نصب کنید و نیازی نیست مراحل بالا را تک به تک انجام دهید.

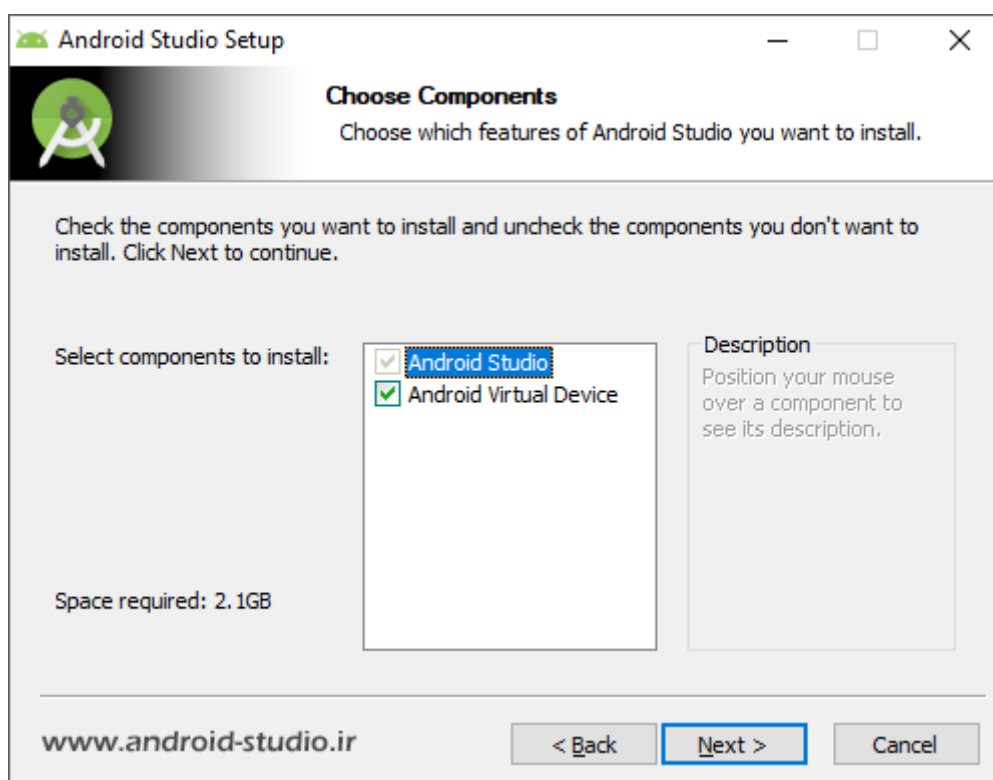


حالا باید اندروید استودیو را نصب کنیم. قبل از شروع مراحل نصب، ارتباط رایانه خود را کاملاً با اینترنت قطع کنید در غیر اینصورت بعد از معرفی مسیر sdk به اندروید استودیو با پیغام بروزرسانی ابزارهای sdk مواجه خواهید شد.

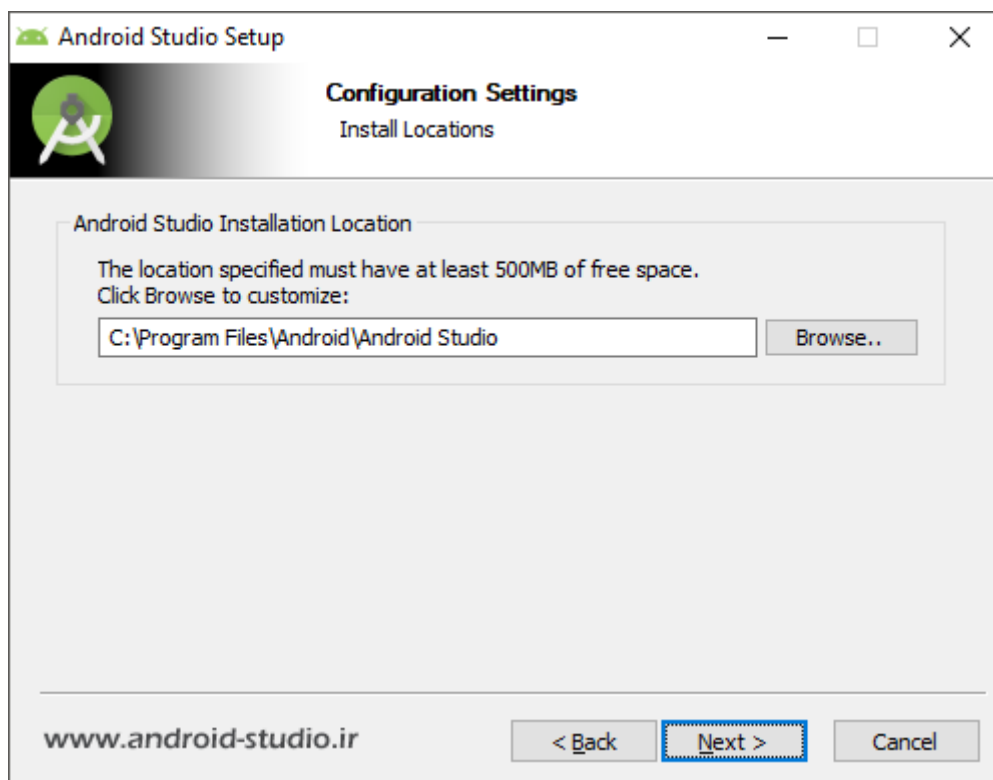
سیستم عامل من ۶۴ بیتی است بنابراین نسخه ۶۴ بیتی اندروید استودیو را نصب می‌کنم.

تذکر: اندروید استودیو نسخه ۳۲ بیتی به صورت پرتابل (Portable) و در قالب یک فایل zip منتشر شده و نیاز به نصب ندارد. بنابراین اگر شما از نسخه ۳۲ بیتی استفاده می‌کنید مراحل مربوط به نصب را نادیده بگیرید.

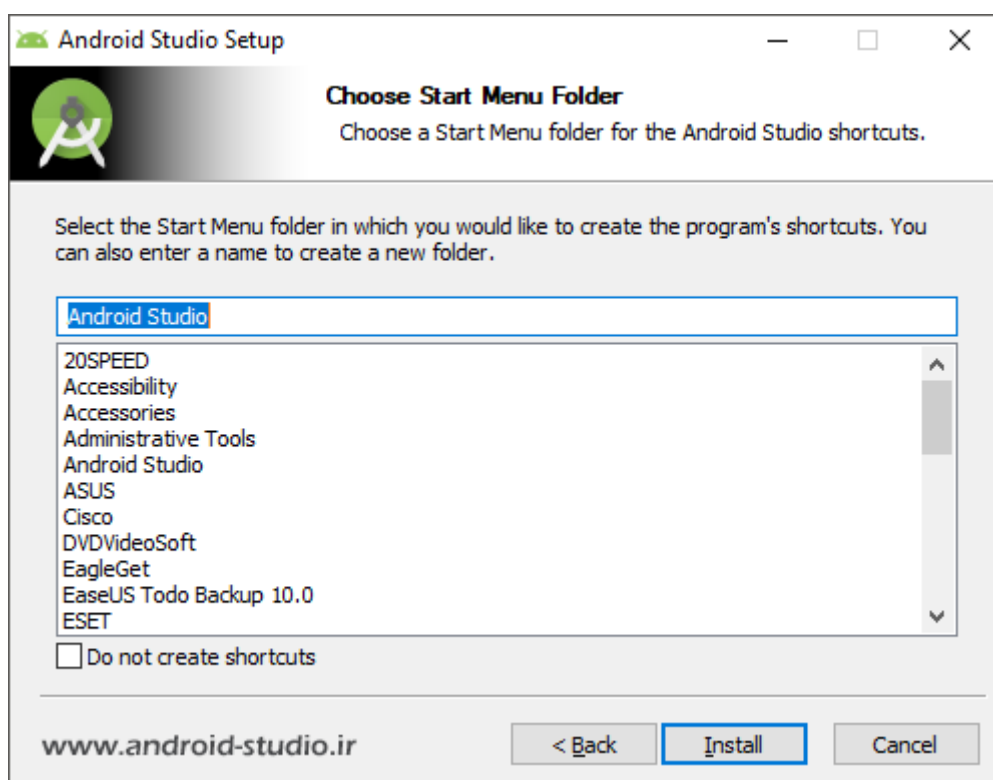


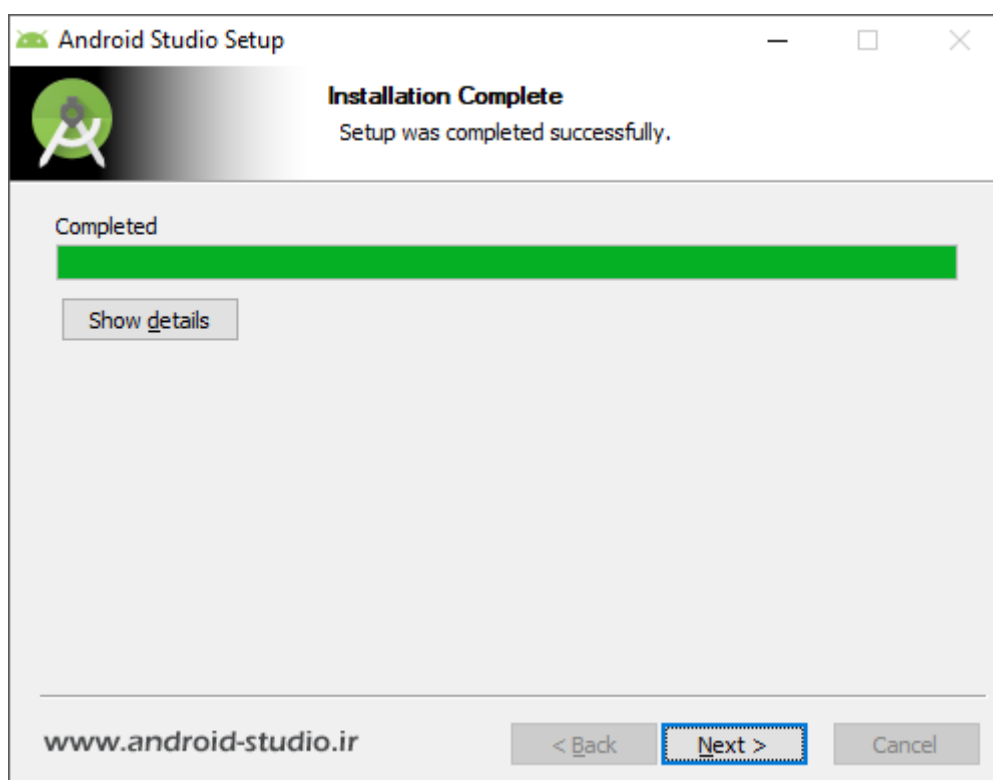
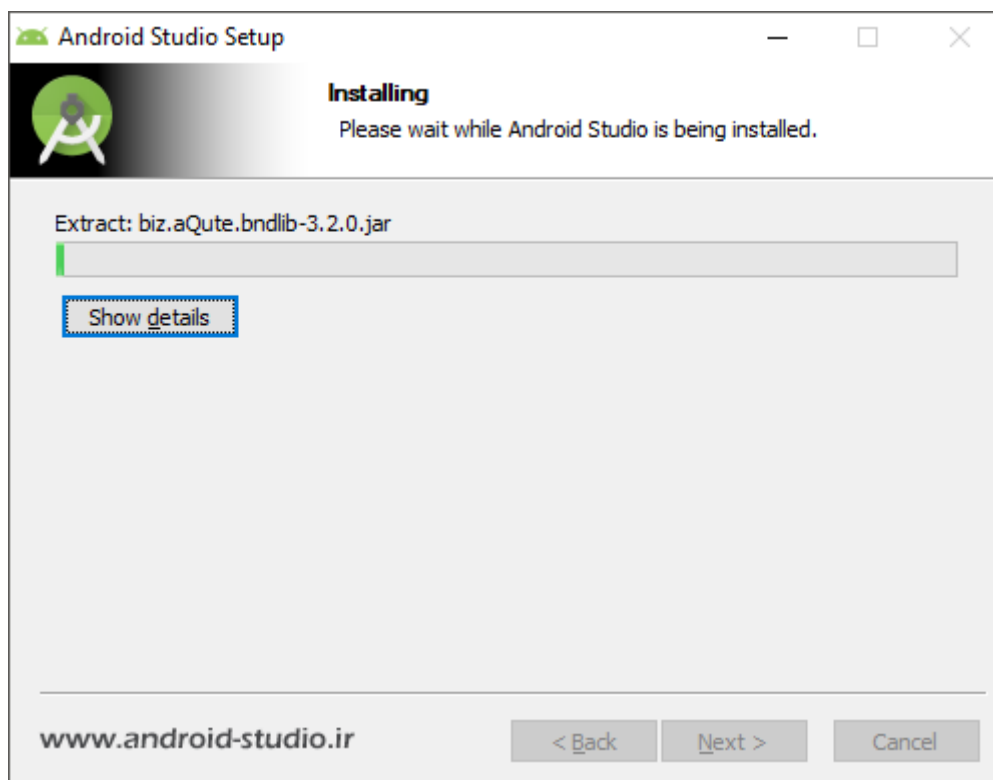


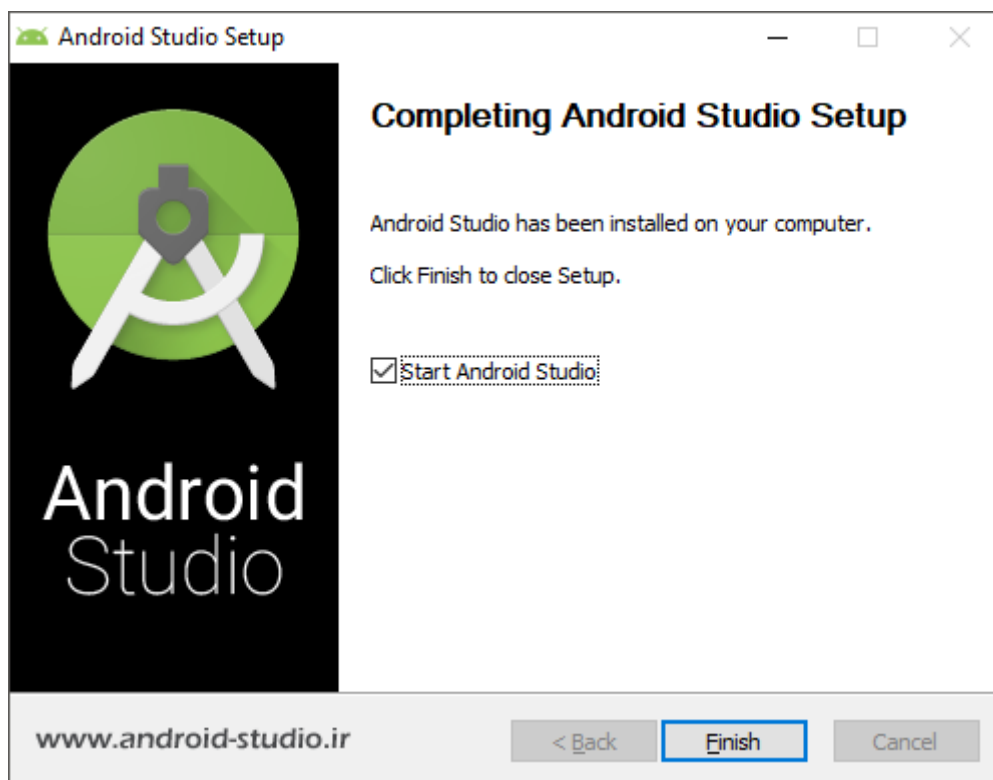
در این مرحله دو گزینه برای نصب تعیین شده که مورد دوم مربوط به AVD است. من این مورد را هم علامت می‌زنم تا نصب شود.



این مرحله مربوط به تعیین محل نصب اندروید استودیو است که بدون تغییر به مرحله بعد می روم.



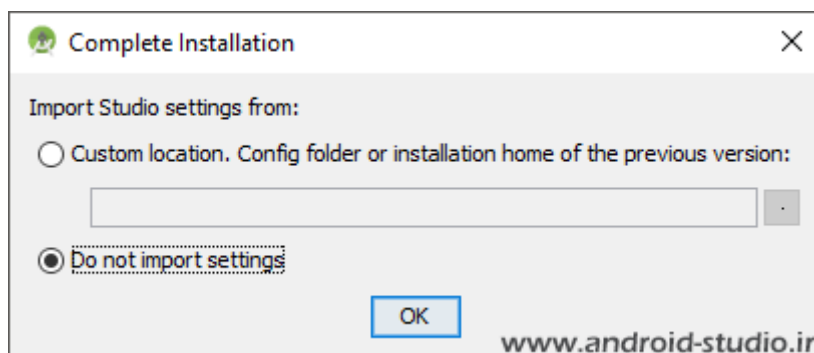




نصب به اتمام رسید. با زدن Finish اندروید استودیو اجرا می شود:

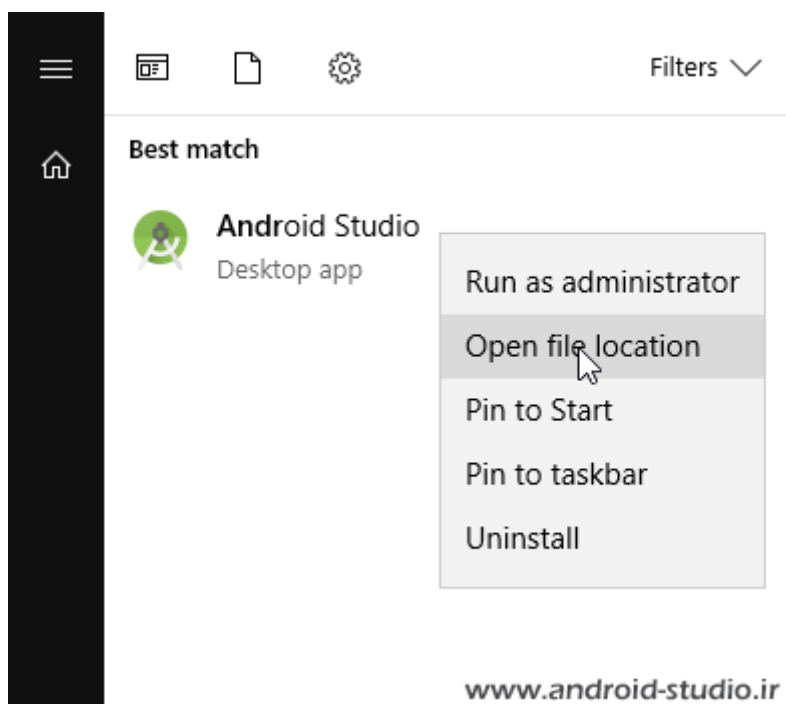


اندروید استودیو در حال لود شدن است و بسته به سخت افزار، چند لحظه زمان می برد. هنگام لودینگ عبارت Powered by IntelliJ Platform را مشاهده می کنید. کمپانی گوگل محیط توسعه (IDE) کمپانی JetBrains را برای اندروید استودیو انتخاب کرده و در واقع با انجام تغییرات و اضافه کردن پلاگین(ها) به IntelliJ IDEA که محیط توسعه جاوا می باشد، آنرا برای توسعه دهندگان اندروید آماده و منتشر کرده است.

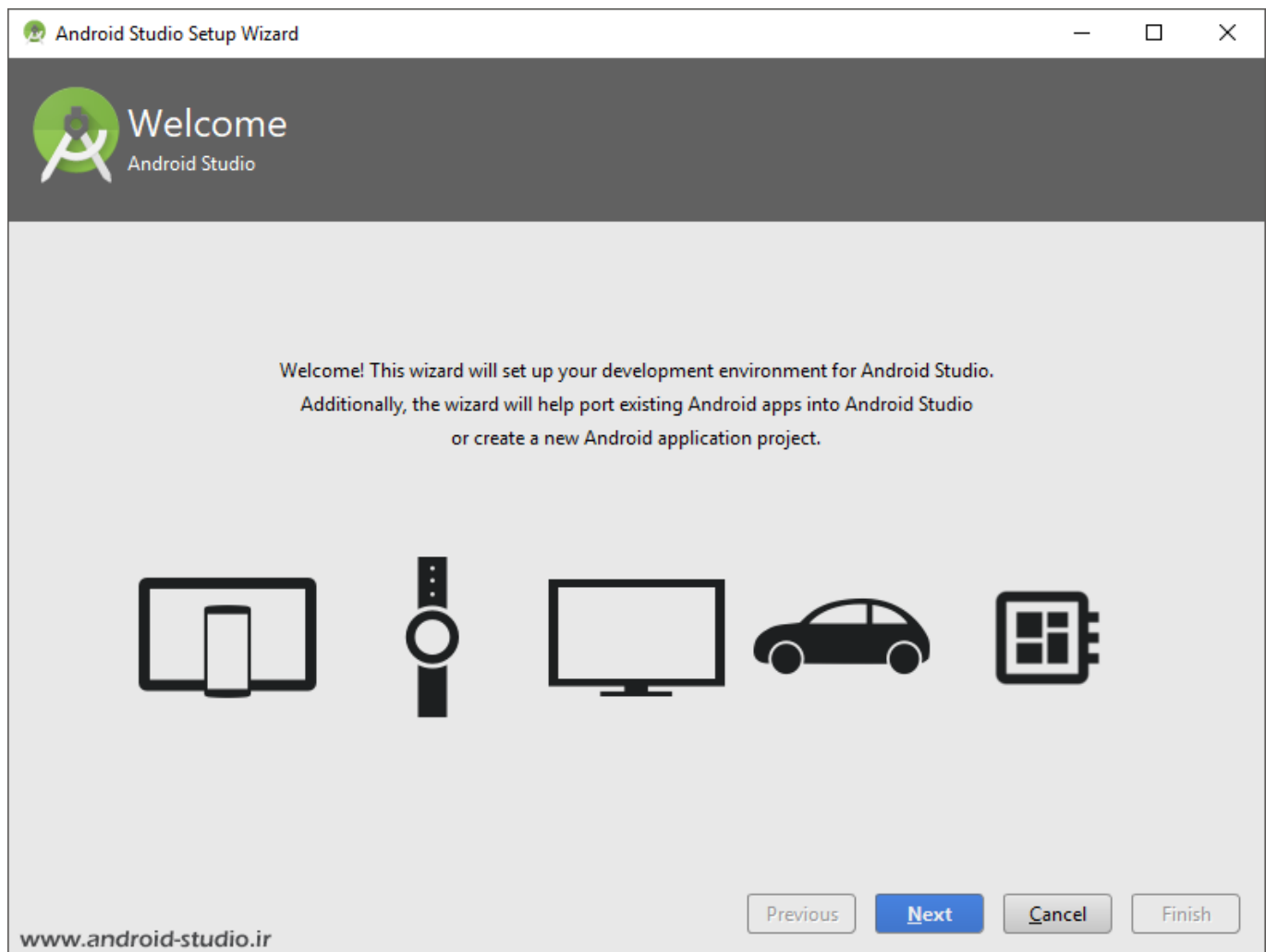
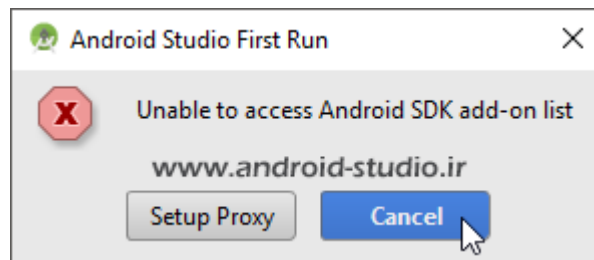


ابتدا پیغامی ظاهر می شود که امکان ایمپورت کردن تنظیمات از نسخه های قبل را در اختیار می گذارد. منظور تنظیمات اصلی اندروید استودیو است (مانند تم محیط توسعه و...). من قصد بازگردانی تنظیمات را ندارم.

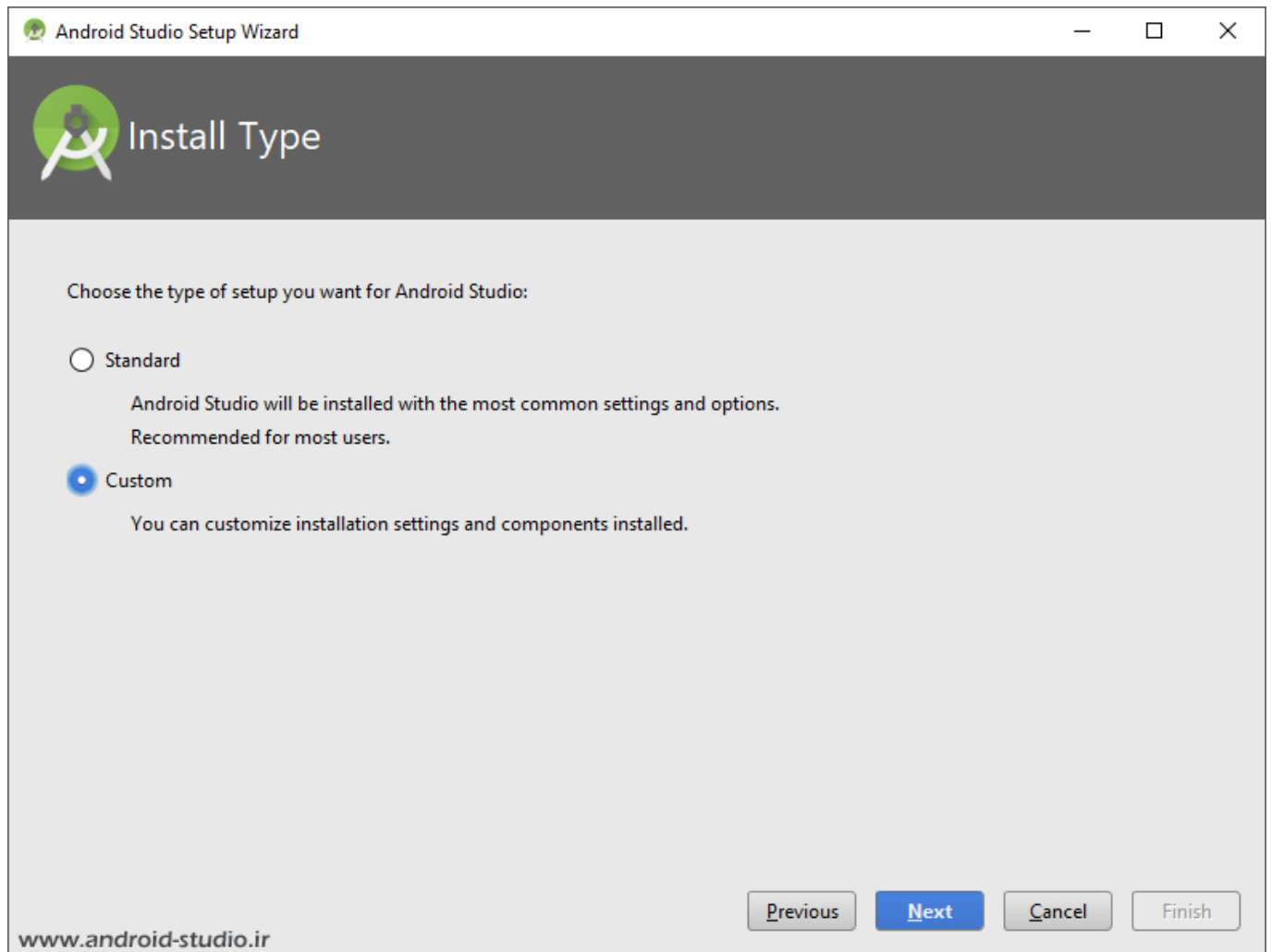
قبل از اینکه به ادامه بحث پردازش در خصوص آیکون نرم افزار نکته ای را یادآوری می کنم. آیکون اندروید استودیو به طور خودکار روی دسکتاپ اضافه نمی شود و اگر مایلید آیکون را روی دسکتاپ داشته باشید، در منوی استارت پس از جستجوی Android Studio روی آن راست کلیک کرده، با انتخاب Open file location پنجره ای باز می شود. آیکون را کپی کرده و به دسکتاپ منتقل می کنیم:

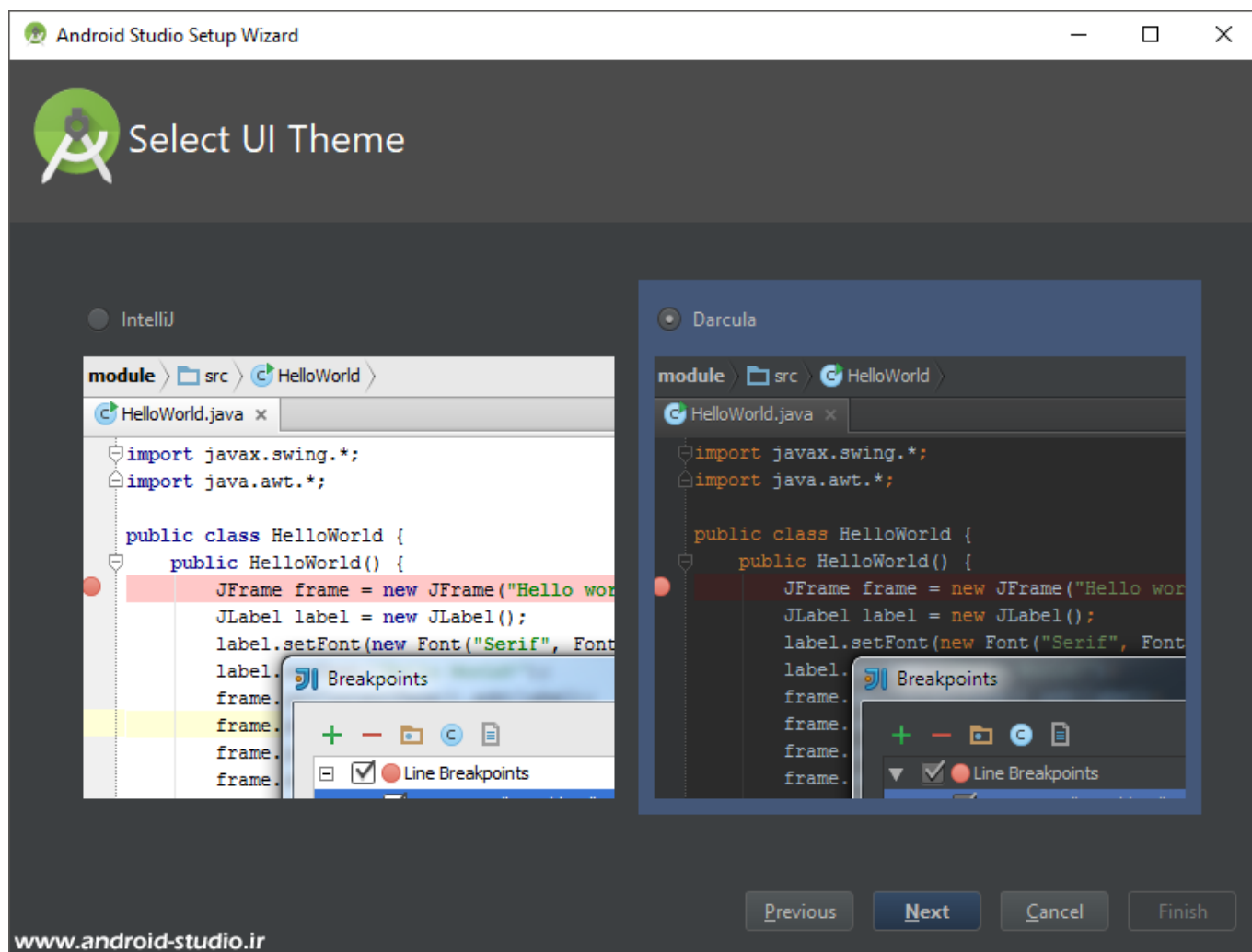


البته اضافه کردن به Taskbar توسط Pin to taskbar هم راه حل دیگر برای دسترسی سریعتر است. بعد از لود شدن اندروید استودیوخطاری مبنی بر عدم دسترسی به اینترنت دریافت و آنرا Cancel می کنم:

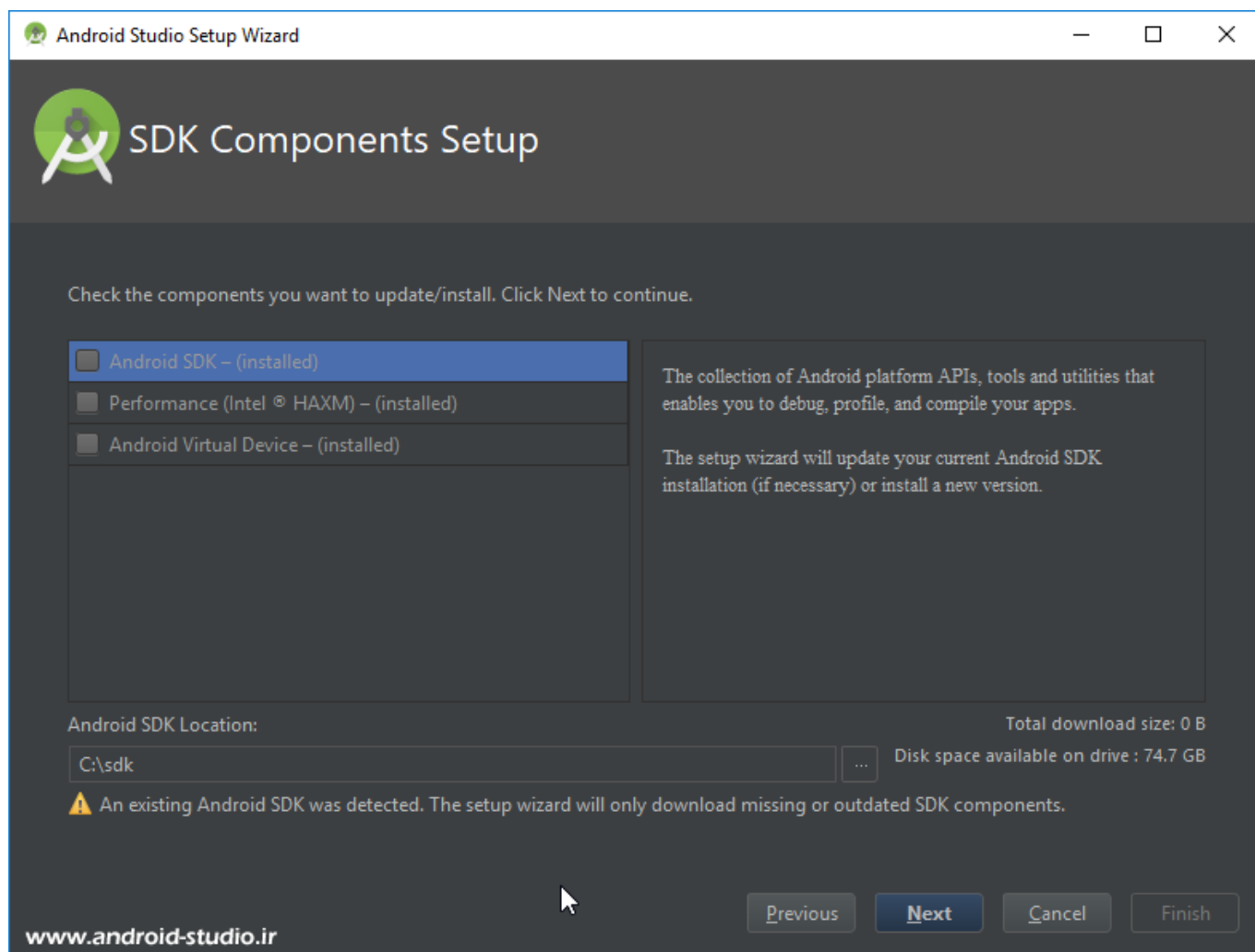


در مرحله بعد، گزینه Custom را انتخاب کرده تا تنظیمات مدنظر خود را اعمال کنم:

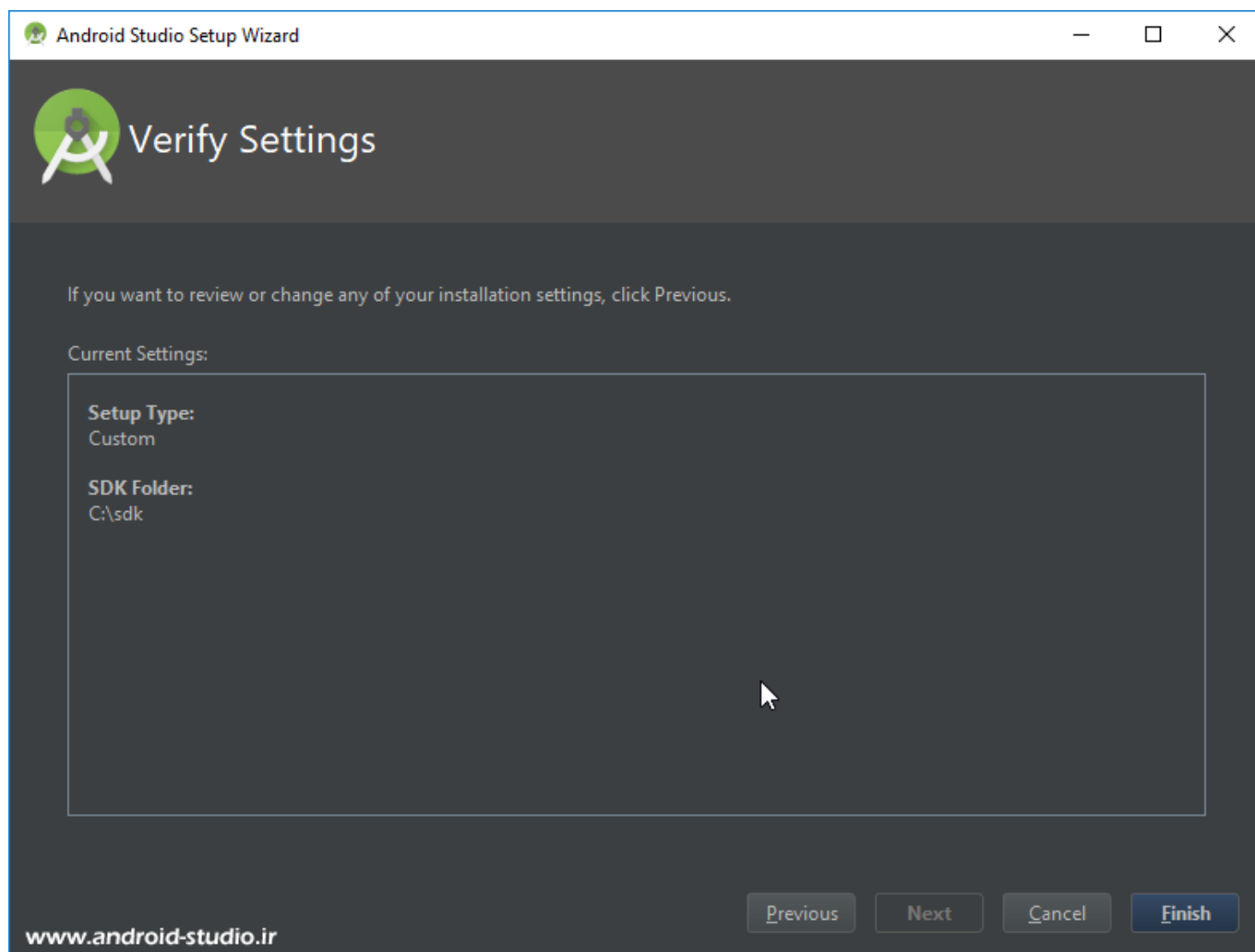


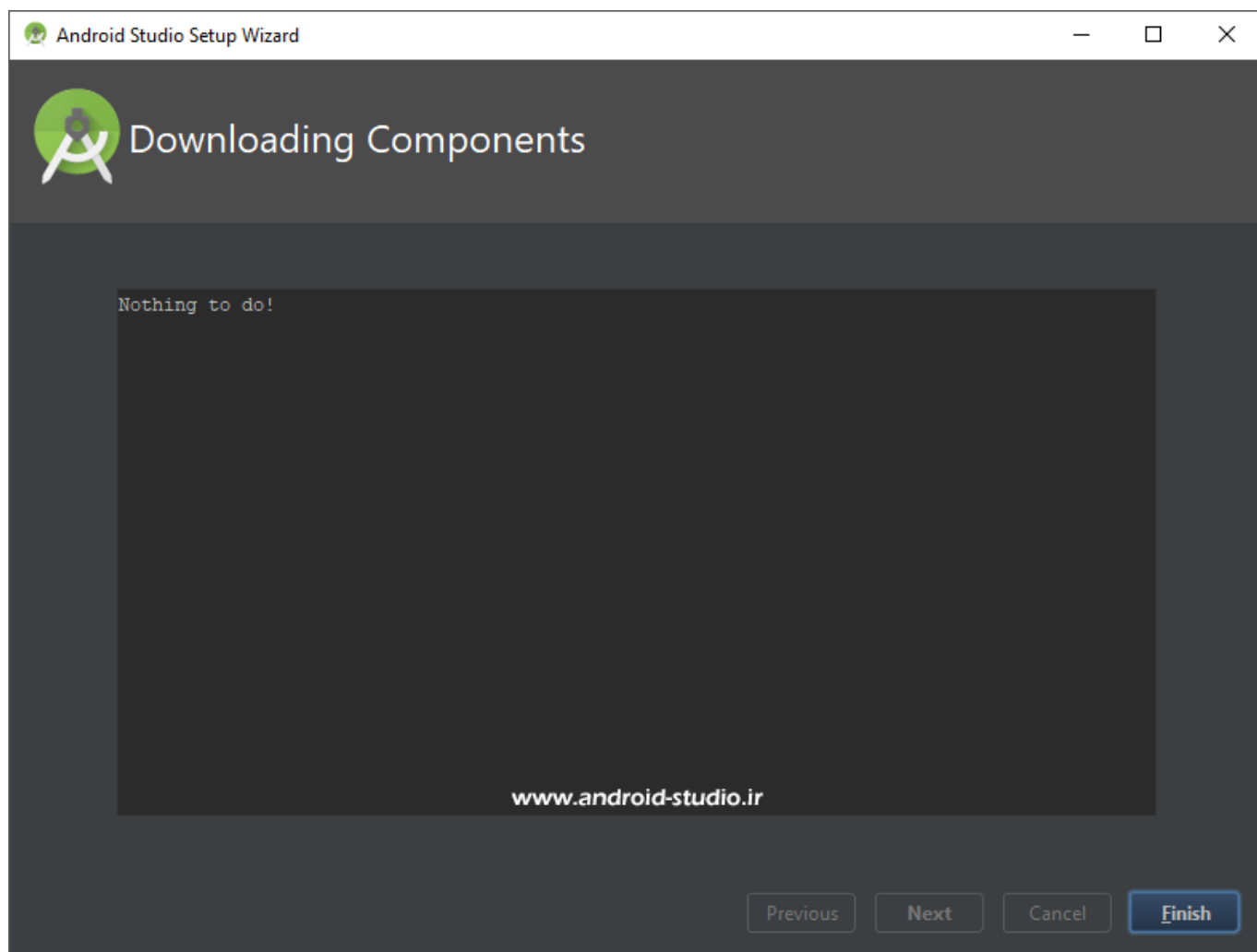


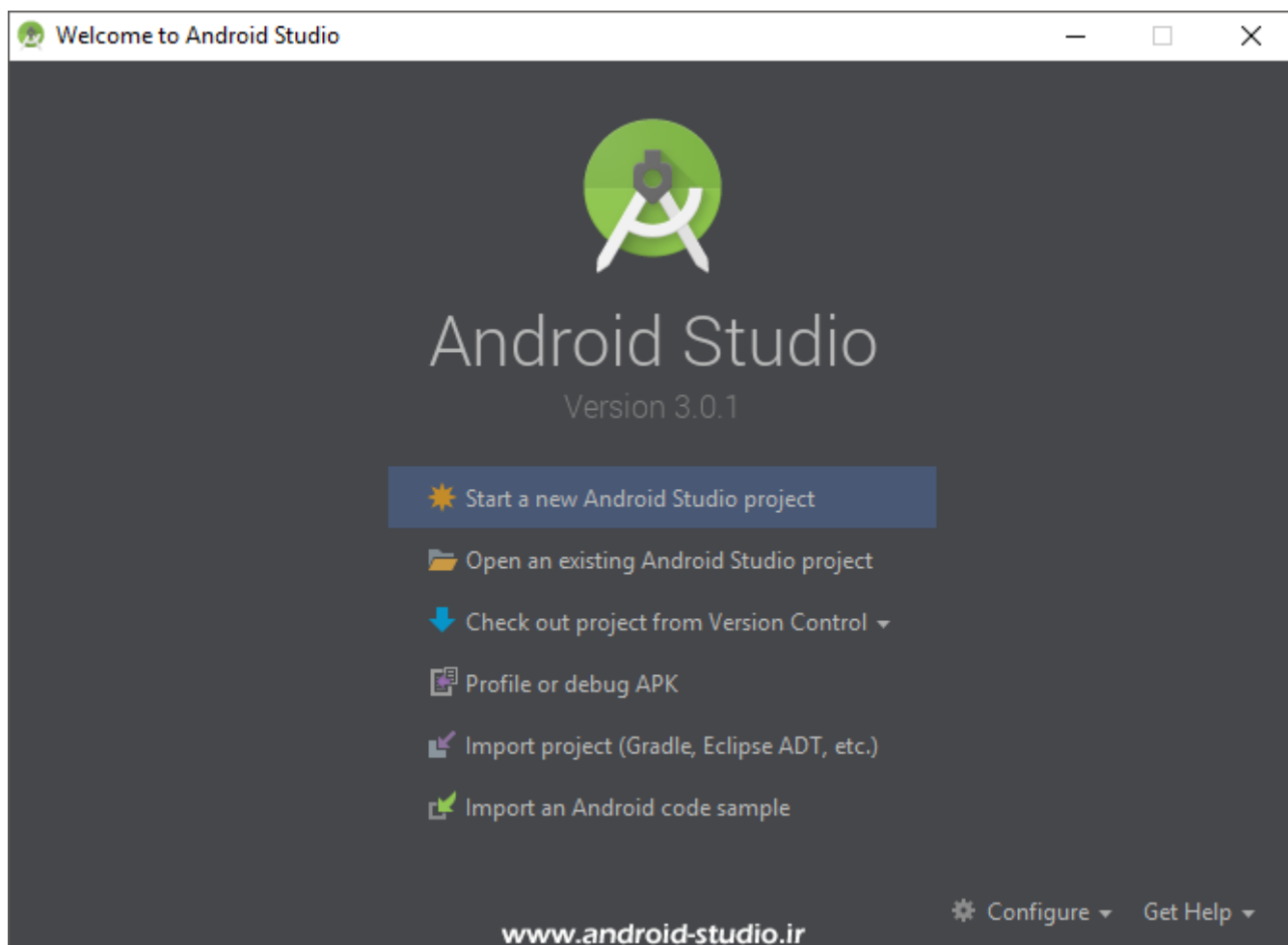
در این مرحله امکان انتخاب تم محیط برنامه وجود دارد. من تم تاریک را انتخاب می کنم تا هنگام کار، فشار کمتری به چشم وارد شود.



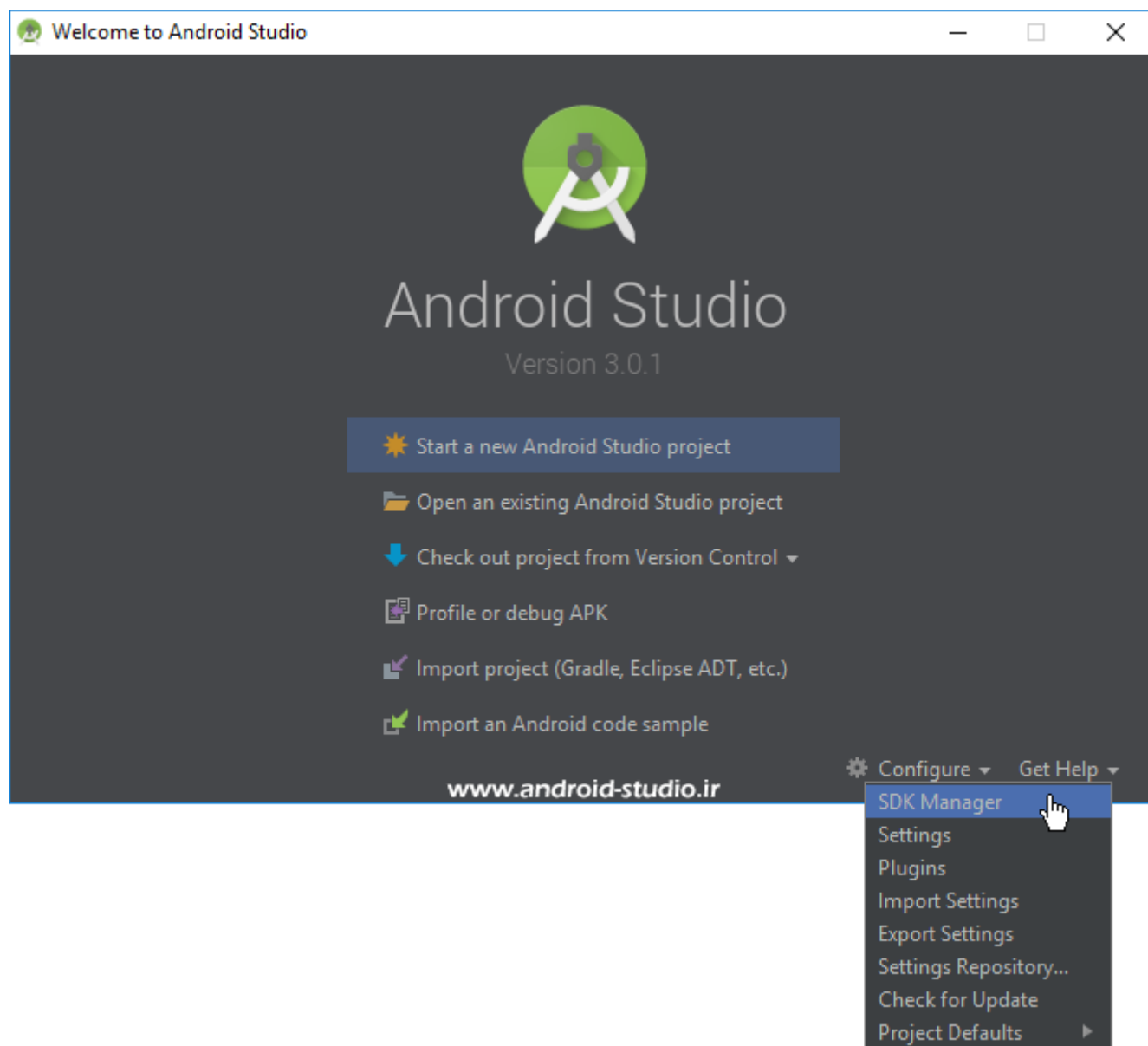
در این مرحله و در قسمت Android SDK Location مسیری که SDK را نصب کرده بودم انتخاب می کنم. اگر SDK به درستی نصب شده و مسیر هم به درستی معرفی شده باشد پیغامی دریافت می شود با این مضمون که Android SDK شناسایی شده و فقط ابزار (کامپوننت های) نصب نشده یا تاریخ گذشته دانلود خواهند شد که البته قبلا اتصال به اینترنت را قطع کرده ام.



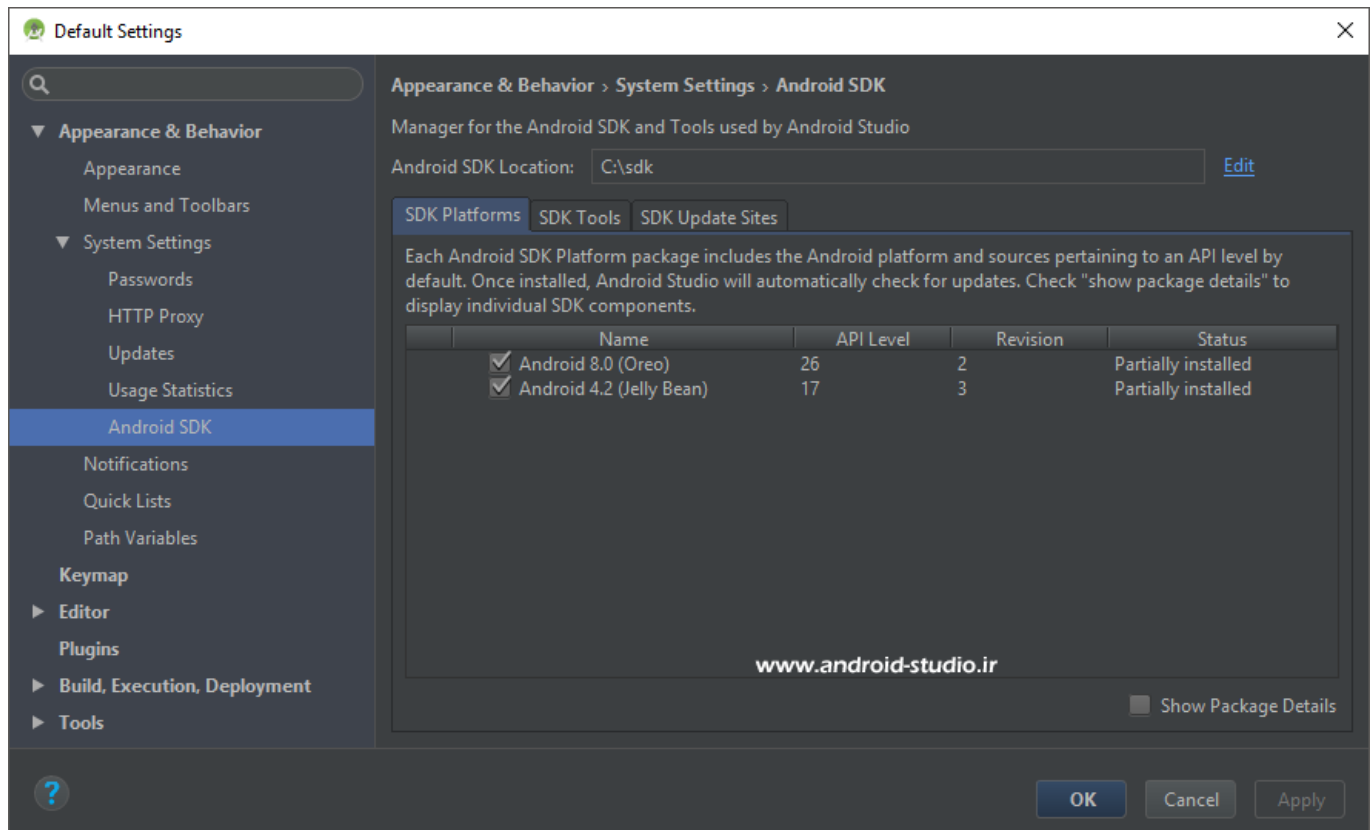




به صفحه ورودی رسیدیم. چند گزینه در اختیار داریم شامل ساخت پروژه جدید، ایمپورت پروژه های قبلی از اندروید استودیو یا Eclipse و... . فعلا با اینها کاری ندارم و سراغ SDK Manager می روم:



البته SDK Manager در محیط اصلی اندروید استودیو (محیط پروژه) نیز در دسترس است اما در این آموزش قصد ساخت پروژه ندارم و از همین پنجره خوش آمد گویی به قسمت مدیریت Android SDK می روم.



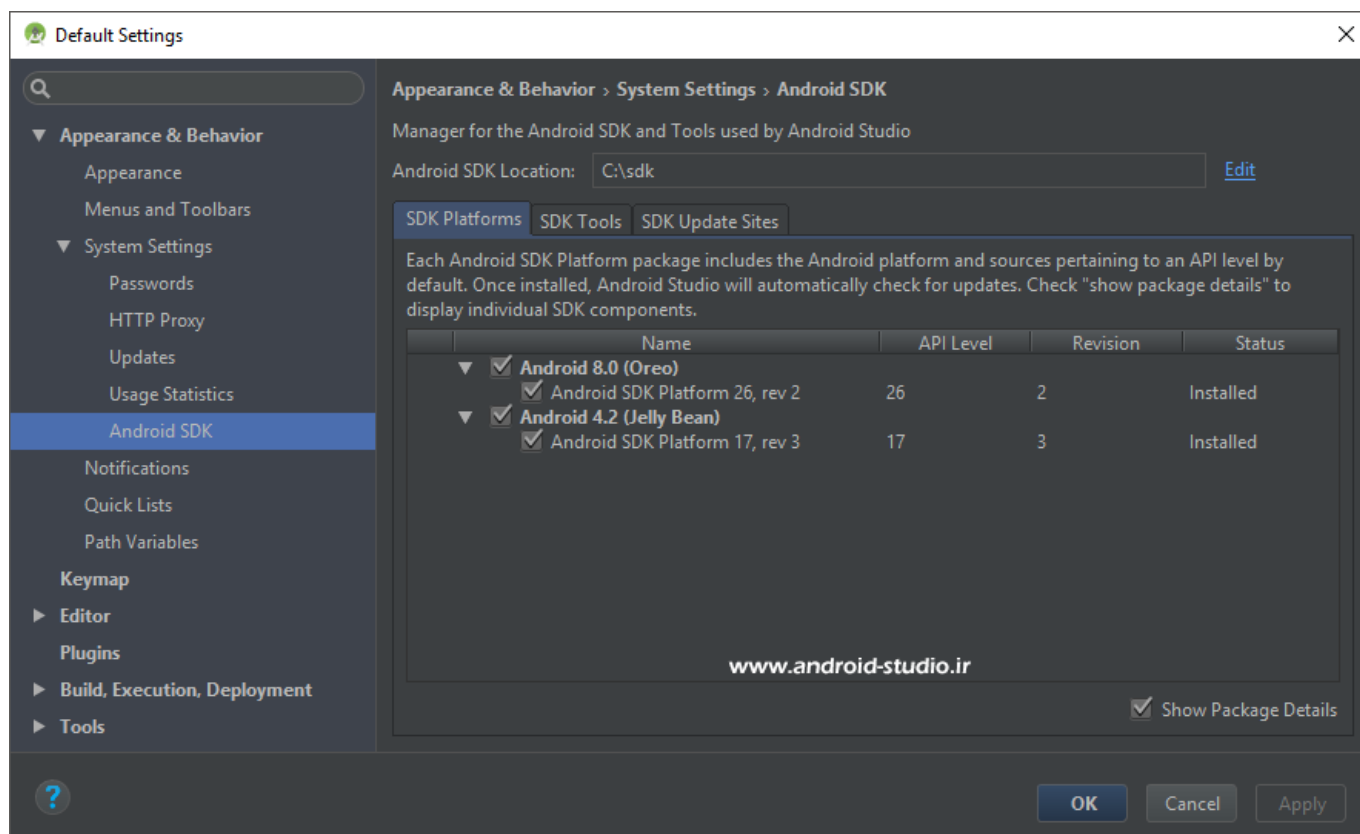
مشاهده می کنید ابزاری که داخل sdk قرار دادیم اینجا لیست شده است.

platform (API) ها در تب SDK Platforms و سایر ابزار در تب SDK Tools قرار گرفته اند.

در قسمت قبل من API های ۱۷ و ۲۶ را در فولدر platforms قرار دادم که هر دو در این لیست به همراه جزئیات نمایش داده شده است. ستون Revision همان شماره نسخه ای است که قبلا توضیح دادم. (مانند عدد ۰۲ در platform-26_r02.zip)

در ستون Status برای هر دو API عبارت Partially installed قید شده است. برای هر API بجز Platform، ابزار دیگری نیز منتشر می شود که عموماً برای ما کاربردی ندارد (مگر در موارد خاص). بنابراین چون برای این دو API فقط Platform را نصب کرده ایم، اندروید استودیو اعلام می کند ابزار این API به صورت کامل نصب نشده اند.

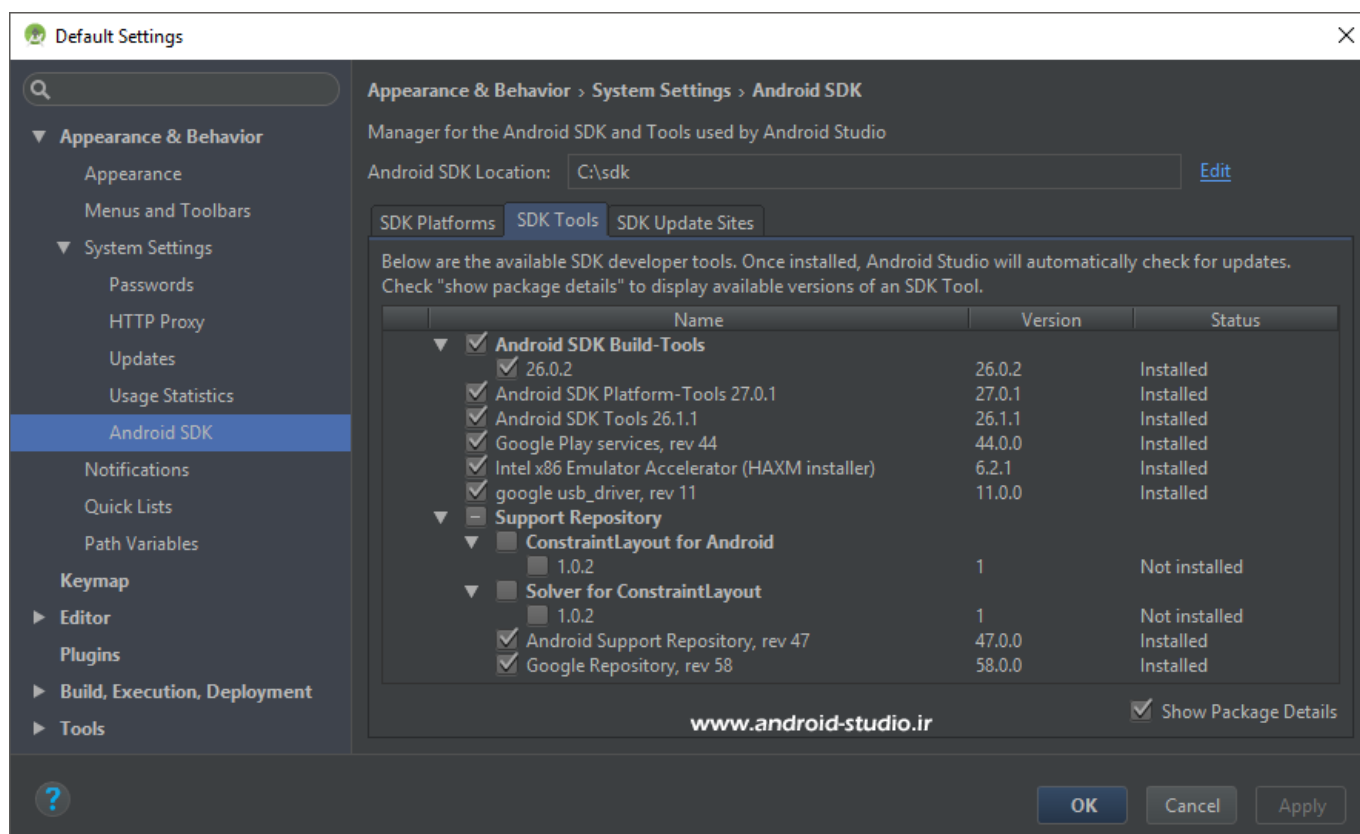
گزینه Show Package Details را علامت می زنم تا جزئیات ابزار و کامپوننت ها نیز نمایش داده شود:



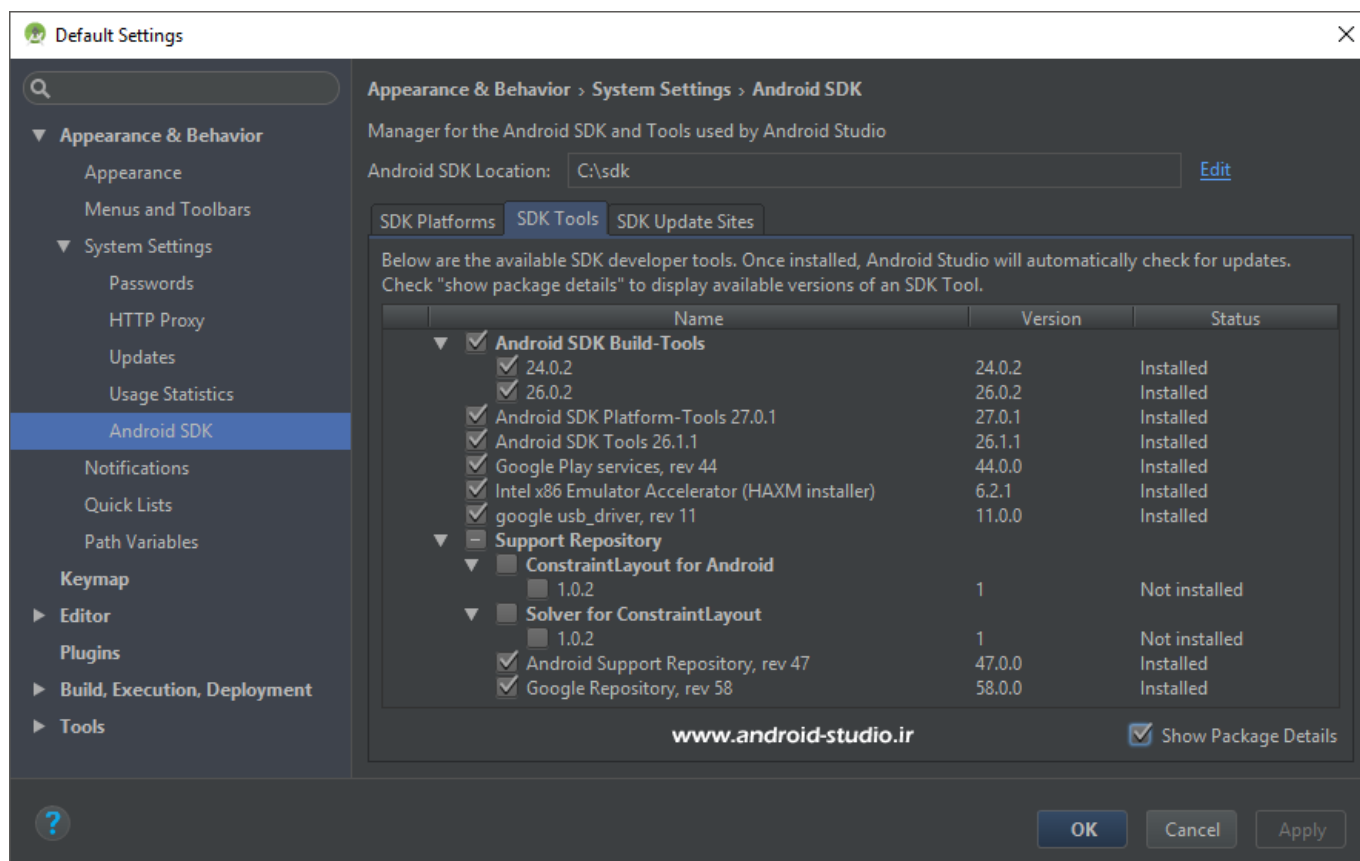
ملاحظه می کنید با فعال شدن این گزینه، Platform به عنوان یک زیرمجموعه از هر API نشان داده شده که به دلیل عدم اتصال به اینترنت یا اتصال با آی پی ایران، سایر ابزاری که برای دانلود در دسترس هستند نمایش داده نمی شود.



تب SDK Tools هم به همین صورت است:



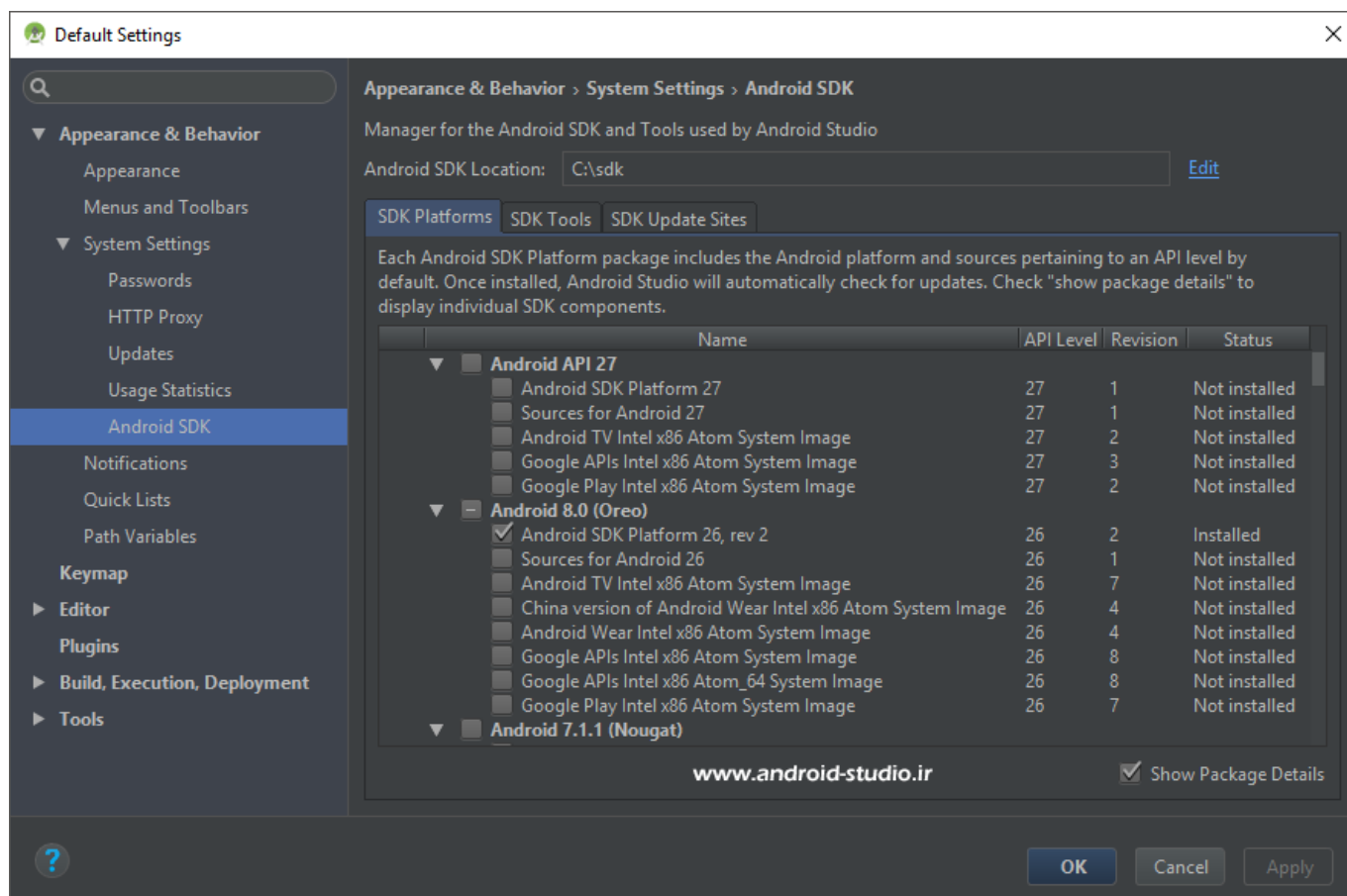
با انتخاب Show Package Details جزئیات بیشتری نمایش داده می شود. به عنوان مثال نسخه ۲۶.۰.۲ از Build Tools به صورت زیر مجموعه قرار گرفته که نشان می دهد می توان نسخه های دیگری نیز به Build Tools اضافه کرد. به عنوان مثال نسخه ۲۴.۰.۲ از Build Tools را به صورت آفلاین به فولدر build-tools در sdk اضافه می کنم. متأسفانه صفحه Android SDK گزینه Refresh ندارد و برای مشاهده تغییرات ناچارا باید پنجره را بسته و مجدد SDK Manager را باز کرد:

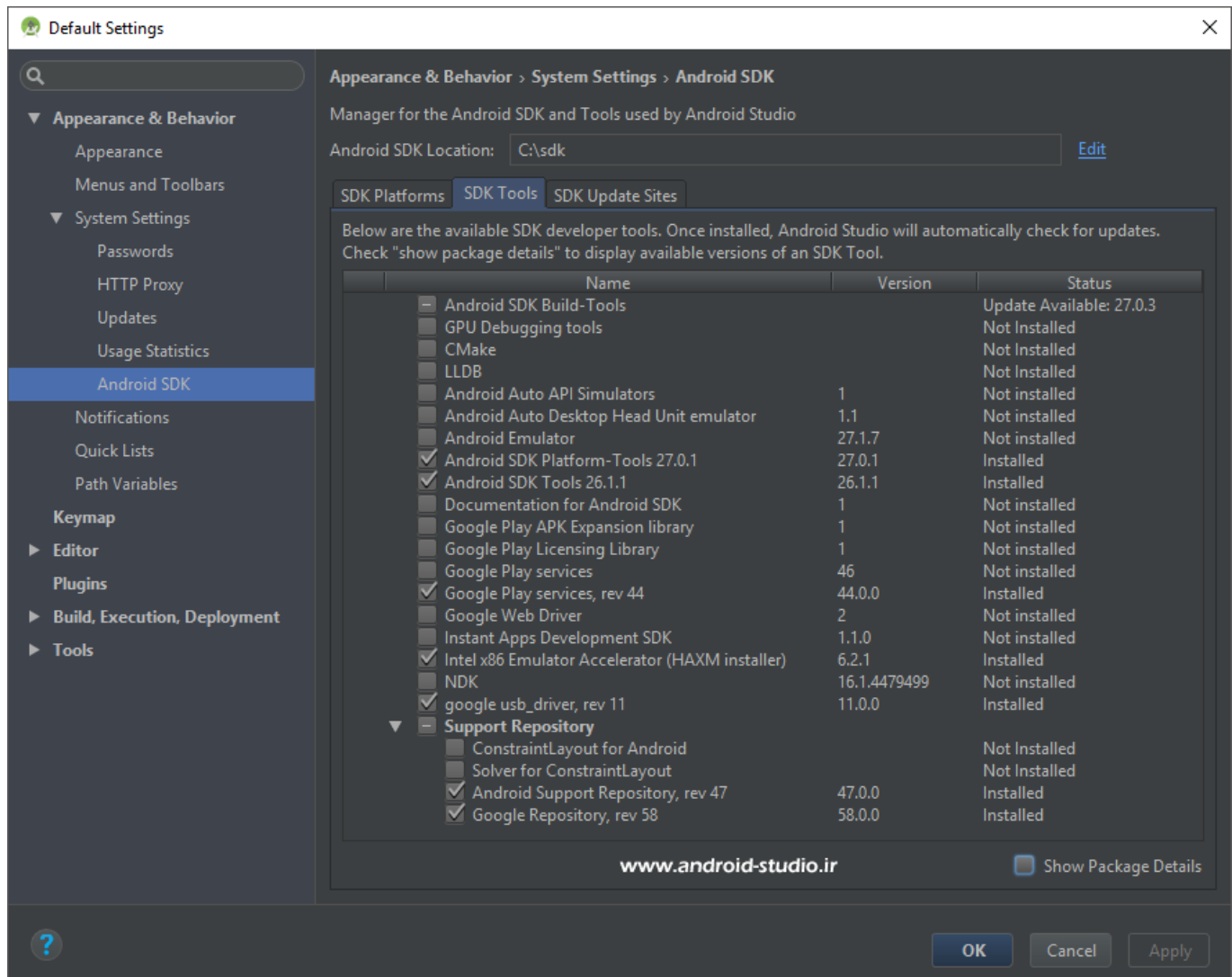


ملاحظه می کنید نسخه ۲۴.۰.۲ نیز به زیرمجموعه Build Tools اضافه شد. ابزار Platform-Tools، SDK Tools، Google Play Service، HAXM، USB Driver، Support Repository دارای زیر مجموعه نیستند که قبلا اشاره کردم از این ابزار فقط یک نسخه می تواند در SDK موجود باشد و در صورت نیاز به تغییر نسخه، می بایست فولدر مربوط به ورژن موجود ابتدا کامل حذف و ورژن جدید جایگزین شود.

دو مورد مربوط به ConstraintLayout هست که مربوط به Layout جدید معرفی شده در اندروید می باشد. نیازی به نصب این دو مورد نیست و با ساخت اولین پروژه اندرویدی، این دو نیز خودکار و به صورت آفلاین نصب خواهد شد.

اتصال رایانه به اینترنت را برقرار می کنم. قبل از باز کردن مجدد SDK Manager لازم است IP را تغییر دهم تا اندروید استودیو بتواند اطلاعات را از سرور فراخوانی کند. ممکن است با استفاده از ابزار تغییر آی پی رایگان و عمومی، مشکل برطرف نشود. در صورت نیاز به اطلاعات بیشتر صفحه [پرسش های رایج](#) را مطالعه کنید.





ملاحظه می کنید بعد از اتصال به اینترنت و تغییر IP، سایر ابزاری که قبلاً نصب نکرده ایم نیز در لیست نشان داده می شود. در حالت عادی تنها چیزی که شما باید در تب SDK Platforms خود داشته باشید، SDK Platform آخرین نسخه اندروید است؛ یعنی Android SDK Platform 27 در تصویر بالا (و احتمالاً SDK Platform ورژن MinSDK پروژه که در مباحث آینده اشاره می شود). مابقی موارد را فقط در صورتی نصب کنید که می دانید به آن نیاز دارید.

توضیحات تکمیلی:

گوگل برای هر نسخه (API) از اندروید ابزار مربوط به آن را نیز در اختیار توسعه دهنده قرار می دهد که شامل SDK Platform، Sources، System image، Google APIs می شود.



SDK Platform برای کامپایل کردن اپلیکیشن استفاده می شود. لازم است توسعه دهنده همواره آخرین API را نصب داشته باشد تا اپلیکیشنی که می سازد با آخرین نسخه اندروید موجود در بازار سازگاری داشته باشد.

با یک مثال ساده توضیح می دهیم: در زمان تهیه این آموزش، آخرین API منتشر شده اندروید نسخه ۲۷ است. حالا اگر آخرین API موجود در SDK من نسخه ۲۳ باشد، اپلیکیشن روی API 23 کامپایل شده. یعنی اپلیکیشن من حداکثر روی دیوایس هایی قابل نصب است که Android 6 (Marshmallow) را داشته باشند و امکان نصب روی دیوایس های جدید مانند گوشی و تبلت که از API 27 بهره می برند میسر نخواهد بود. پس لازم است همزمان با معرفی نسخه های جدید اندروید، API مربوط به آن را دریافت و نصب کنیم. البته از زمان معرفی نسخه جدید اندروید تا ورود دیوایس های آن به بازار، چند ماهی طول می کشد اما شرکت های سازنده برای دیوایس های رده بالا عموماً بعد از یک بازه زمانی حدوداً دو ماهه آپدیت ارائه می دهند.

مورد بعدی Sources است که شامل سورس های API بوده و الزامی در نصب آن نیست (مگر مواردی که مایل باشید از سورس ها کمک بگیرید).

گزینه بعد مربوط به System image ها می شود. از سیستم ایمیج ها برای اجرای شبیه ساز (امولاتور) اندروید روی رایانه استفاده می شود. شبیه سازها توسعه دهندگان را از تست اپ خود روی دیوایس های حقیقی مانند گوشی و تبلت بی نیاز می کنند. فرض کنید قصد تست و خطایابی پروژه خود را دارید و سعی دارید اپلیکیشن خود را روی تعداد زیادی دیوایس و نسخه های مختلف اندروید تست کنید تا مطمئن شوید اکثر کاربران در استفاده از اپلیکیشن شما به مشکلی برخورد نخواهند کرد. اگر بخواهید اپ خود را روی اندروید های ۳، ۴، ۵، ۶ و ۷ تست کنید، می بایست ۵ گوشی موبایل با سیستم عامل اندروید در اختیار داشته باشید! اگر به این لیست، تبلت را هم اضافه کنیم می شود ۱۰ دیوایس! پس عملاً غیر ممکن است بتوانیم اپ را روی تمامی دیوایس های حقیقی تست و خطایابی کنیم. اینجا شبیه ساز یا Emulator به کمک ما می آید. شبیه سازهای متعددی توسط شرکتهای مختلف برای اندروید منتشر شده است. گوگل هم AVD (Android Virtual Device) به معنی دیوایس مجازی اندروید) را به صورت پیش فرض و رایگان درون اندروید استودیو تعبیه کرده که البته به دلایلی از جمله سنگینی بیش از اندازه آن، تعدادی زیادی از توسعه دهندگان اندروید از شبیه سازهای ثانویه مانند Genymotion استفاده می کنند.

نکته: System Image های موجود در لیست SDK Platforms فقط مربوط به شبیه ساز AVD اندروید استودیو است و اگر از سایر شبیه سازها استفاده می کنید نیازی به نصب این سیستم ایمیج ها نیست.



مورد آخر Google APIs می باشد که بازهم بجز مواردی که به آن نیاز دارید، دریافت و نصب آن الزامی نیست.

به طور خلاصه برای من که از شبیه ساز Genymotion استفاده می کنم، تنها داشتن Android SDK Platform برای API هایی که تمایل به استفاده دارم، کفایت می کند.

نکته: اگر قصد استفاده از شبیه ساز اندروید استودیو را دارید ممکن است لیست System image های مربوط به هر API برایتان گیج کننده باشد. به عنوان مثال Android Marshmallow تعداد ۱۰ سیستم ایمیج را شامل شده است:

Name	API Level	Revision	Status
▼ Android 6.0 (Marshmallow)			
Google APIs	23	1	Not installed
Android SDK Platform 23	23	3	Not installed
Sources for Android 23	23	1	Not installed
Android TV ARM EABI v7a System Image	23	3	Not installed
Android TV Intel x86 Atom System Image	23	9	Not installed
Android Wear ARM EABI v7a System Image	23	6	Not installed
Android Wear Intel x86 Atom System Image	23	6	Not installed
ARM EABI v7a System Image	23	6	Not installed
Intel x86 Atom System Image	23	9	Not installed
Intel x86 Atom_64 System Image	23	9	Not installed
Google APIs ARM EABI v7a System Image	23	20	Not installed
Google APIs Intel x86 Atom System Image	23	20	Not installed
Google APIs Intel x86 Atom_64 System Image	23	20	Not installed

به نامگذاری ها دقت کنید. دو مورد اول مربوط به Android TV (تلویزیون هوشمند اندرویدی) و دو مورد دوم Android Wear (ابزار پوشیدنی) هستند. پس واضح است ما با اینها کاری نداریم (مگر اینکه قصد داشته باشیم برای تلویزیون های هوشمند یا ساعت های هوشمند اپلیکیشن بسازیم). ۶ مورد بعد به دو دسته ۳ تایی تقسیم می شوند که ۳ مورد آخر مشابه ۳ مورد قبل هستند با این تفاوت که شامل Google APIs نیز می شوند. بدون در نظر گرفتن شرایط خاص، شاید بهتر باشد سیستم ایمیجی را دریافت کنید که شامل Google APIs هم باشد. حالا باز هم ۳ انتخاب داریم. اگر CPU (پردازنده) رایانه شما intel است گزینه Intel در غیر اینصورت اگر پردازنده AMD استفاده می کنید گزینه ARM EABI را می بایست نصب کنید. Intel هم دو گزینه Atom و Atom_64 دارد که برای اکثر اپلیکیشن ها سیستم ایمیج Atom کفایت می کند (Atom_64 برای موارد خاصی استفاده می شود که اپلیکیشن با معماری ۶۴ بیتی ساخته شده است).



به طور خلاصه اگر پردازنده رایانه شما برند اینتل است و قصد دارید اپلیکیشن خود را در اندرویدهای Google APIs Intel x86 System image، Marshmallow، Lollipop و Nougat تست کنید، باید Google APIs Intel x86 System image را برای API های ۲۱، ۲۲، ۲۳، ۲۴ و ۲۵ دریافت کنید.

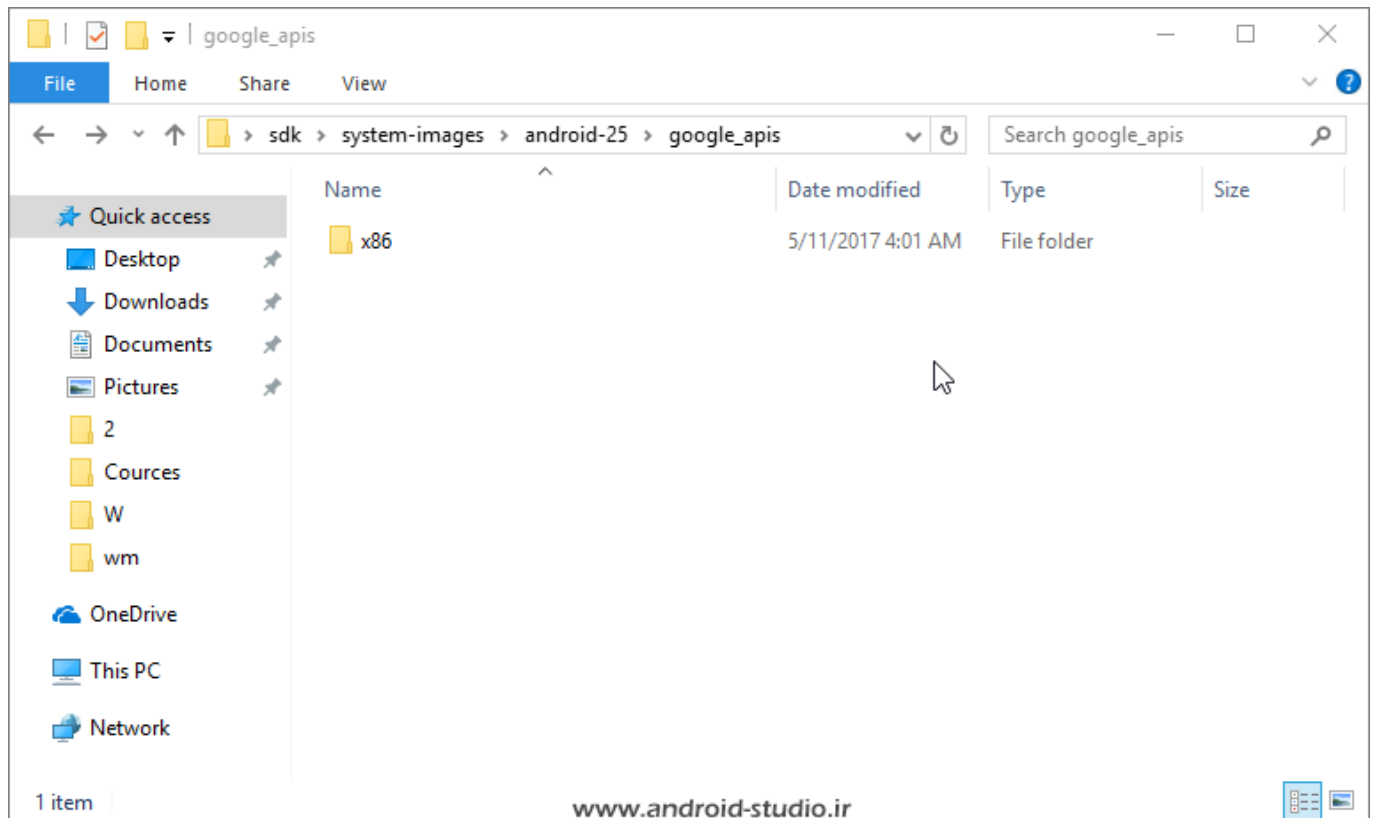
مجدد تاکید می کنم در حال حاضر استفاده از شبیه سازهایی مانند Genymotion راه حل بهتری نسبت به شبیه ساز AVD اندروید استودیو است. با این حال وظیفه دانستم توضیحات لازم را ارائه دهم.

نکته: به دلیل حجم و تعداد بالای System Image های شبیه ساز داخلی اندروید استودیو و همچنین توصیه به استفاده از سایر شبیه سازها، لینک دانلود این سیستم ایمیج ها روی سایت ما قرار نمی گیرد. این سیستم ایمیج ها حجمی حدود ۲ گیگابایت دارند در حالی که سیستم ایمیج های شبیه ساز جنی موشن حدود ۳۰۰ الی ۴۰۰ مگابایت حجم دارد.

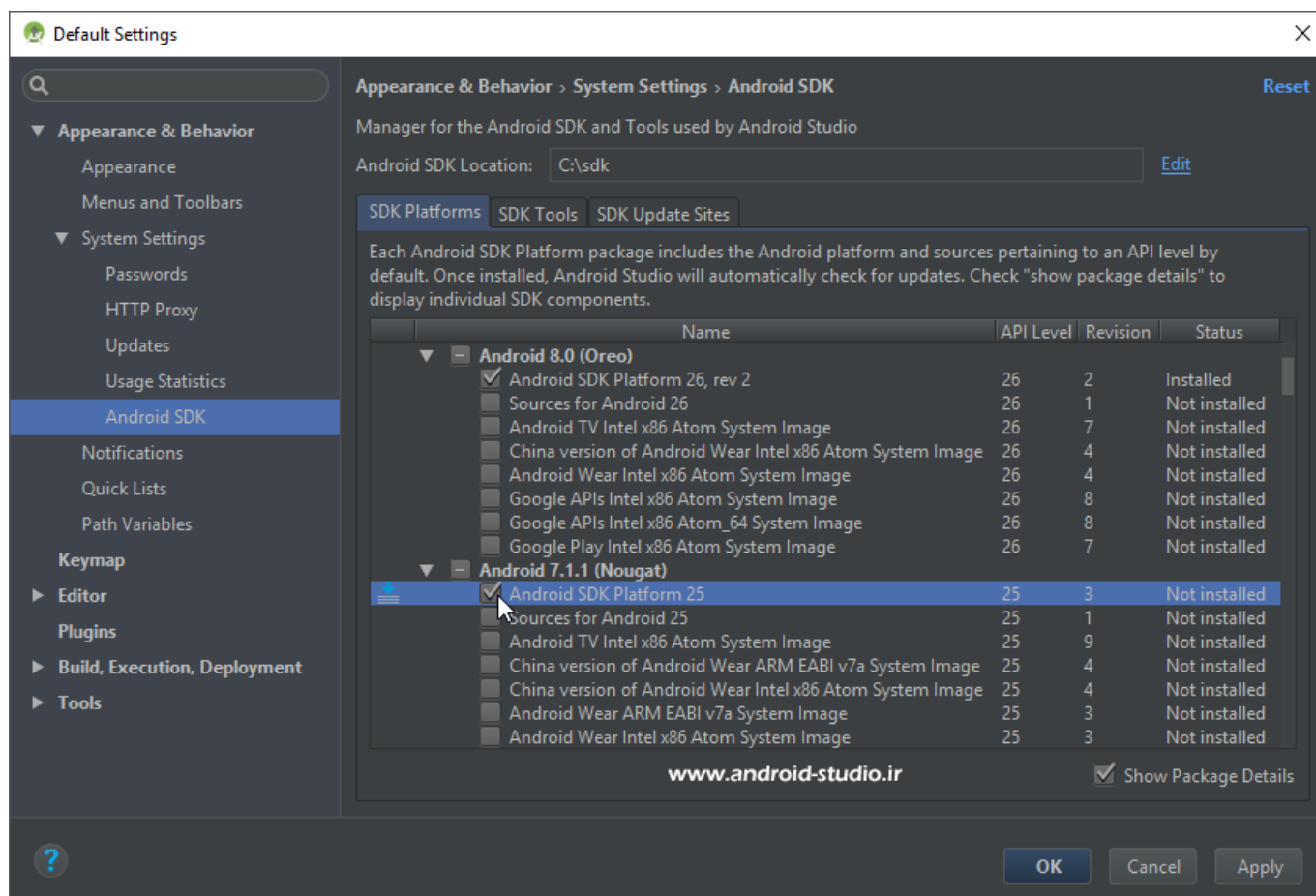
علاوه بر سیستم ایمیج، لینک دانلود ابزار Sources نیز به دلیل ضروری نبودن آن، روی وب سایت ما قرار نمی گیرد. با اینحال اگر مایل به نصب آفلاین این دو مورد هستید، بعد از دانلود فایل مربوطه از سایر وب سایتها (یا روشی که در ادامه برای پیدا کردن لینک دانلود ابزار معرفی می کنم)، خروجی فایل فشرده را در مسیرهای زیر قرار دهید:

✓ Sources: یک فولدر با نام sources در sdk ایجاد کرده و خروجی فایل فشرده مربوط به Source را به داخل آن منتقل کنید.

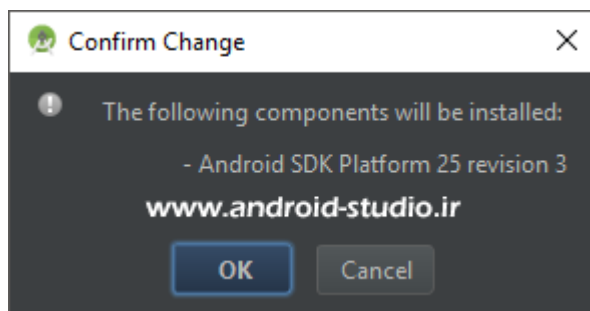
✓ System Image: ابتدا یک فولدر با نام system-images به sdk اضافه کنید. سپس درون این فولدر، فولدر جدیدی با نام android-xx ایجاد کنید که xx عدد مربوط به API است. به عنوان مثال برای API 25 فولدر باید به صورت android-25 نامگذاری شود. داخل android-25 فولدر دیگری با نام google_apis بسازید و خروجی فایل سیستم ایمیج (مانند x86-25_r08.zip) را به این فولدر منتقل کنید. به آدرس بار تصویر زیر دقت کنید:

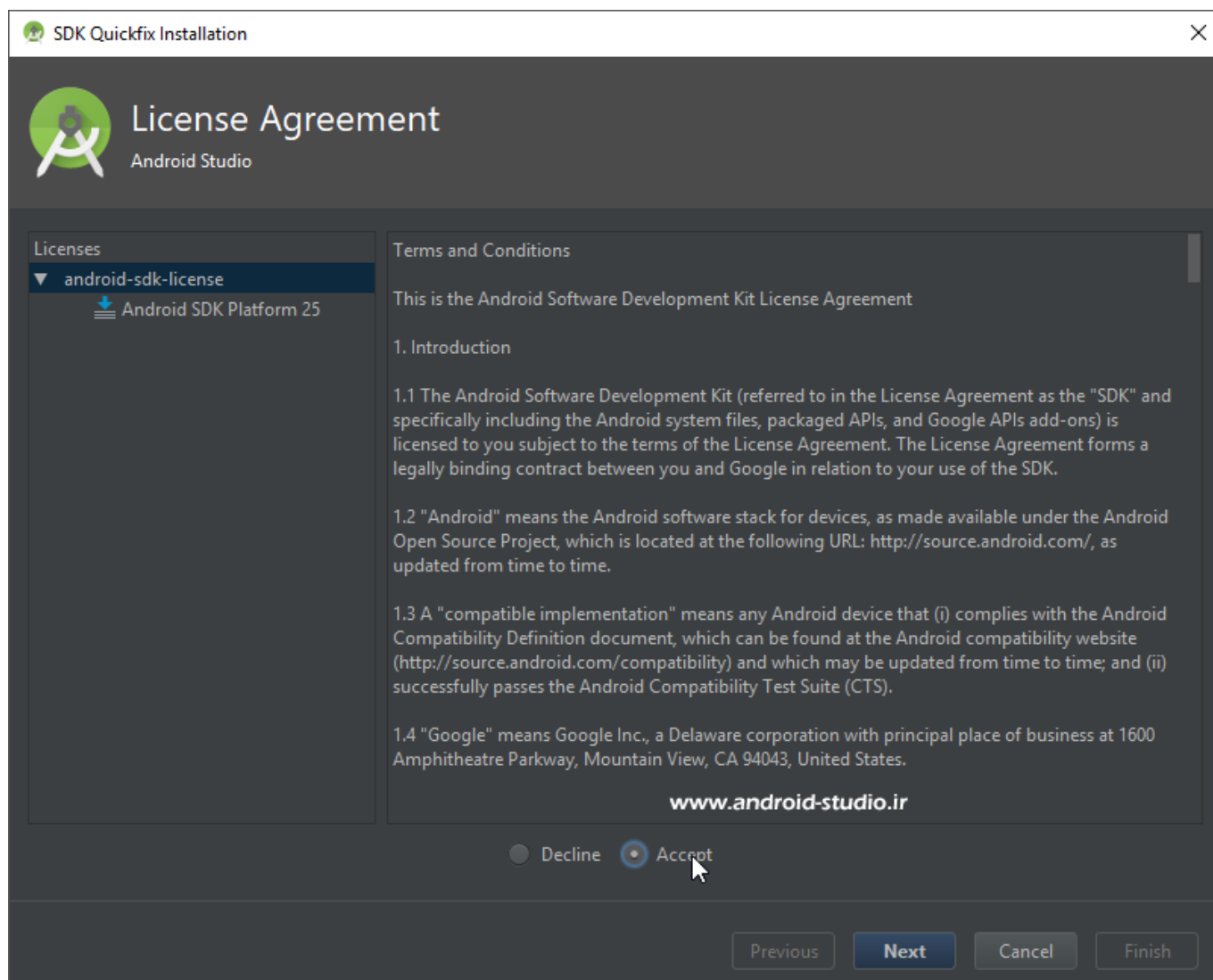


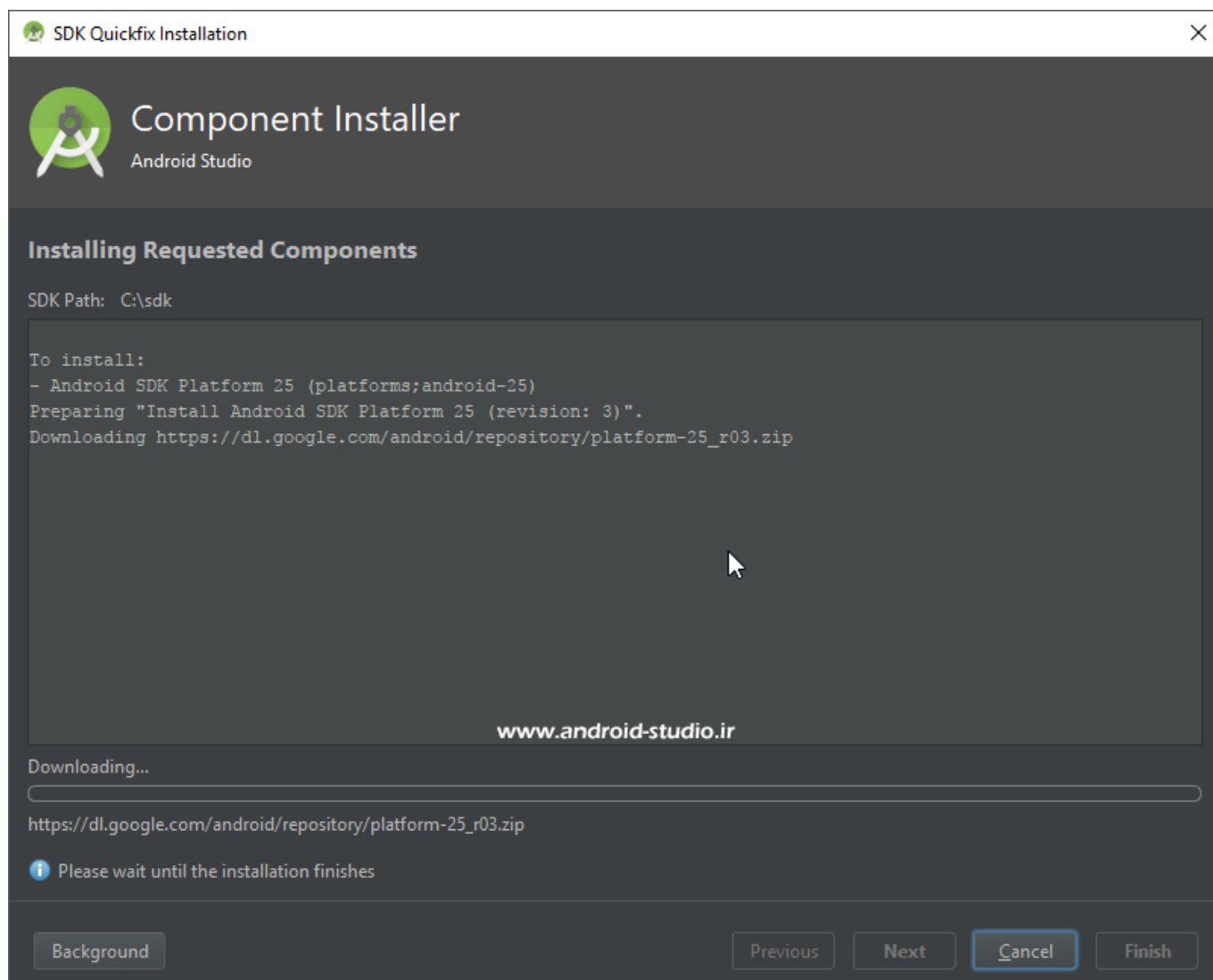
به SDK Manager برمی‌گردم. تا اینجا با نصب آفلاین ابزار آشنا شدیم. نصب آنلاین ابزار ساده است و همه مراحل به طور خودکار انجام می‌شود. برای مثال می‌خواهم SDK Platform 25 را آنلاین نصب کنم:



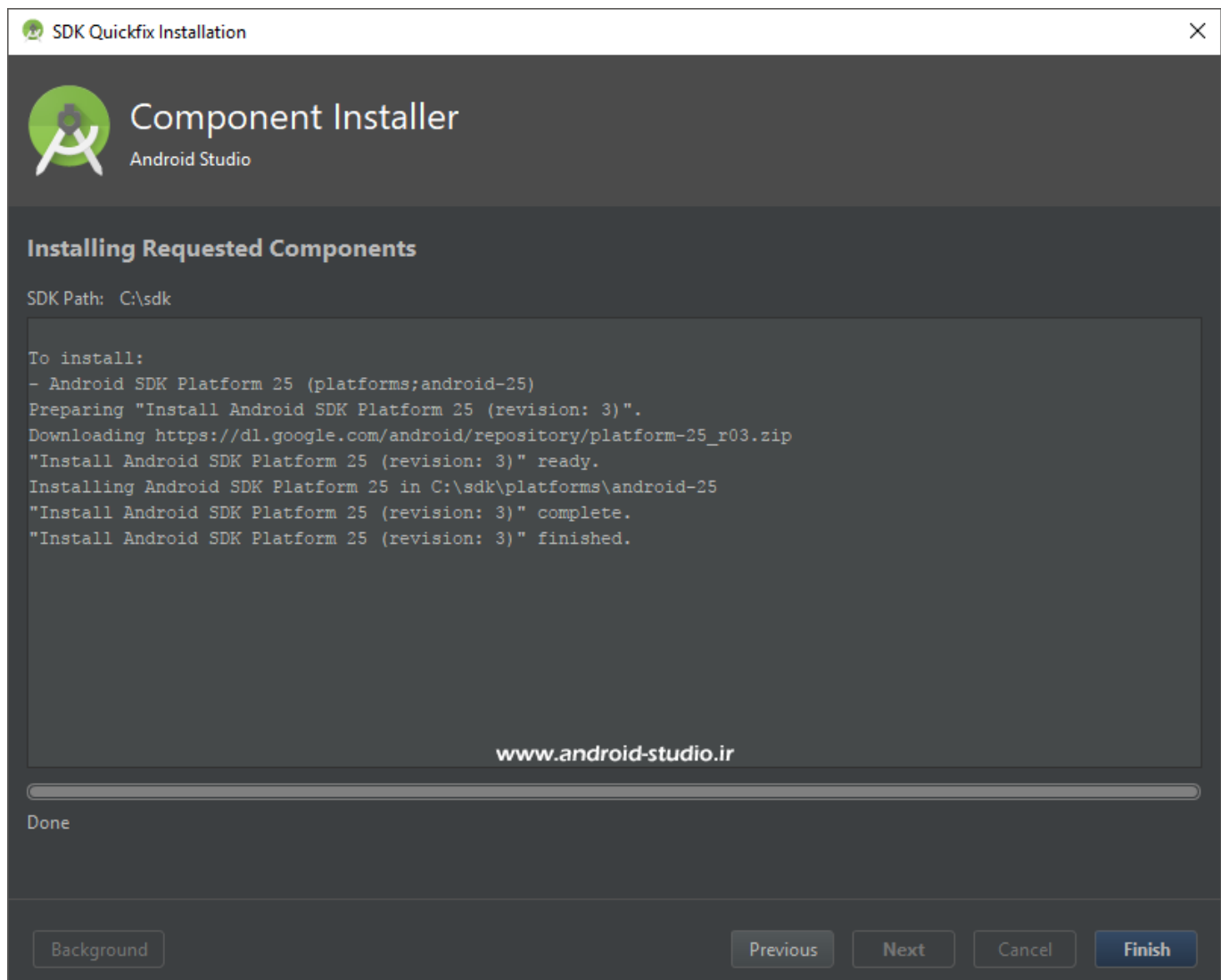
با انتخاب گزینه مدنظر، علامت دانلود در سمت چپ ظاهر می شود. روی گزینه Apply کلیک می کنم:

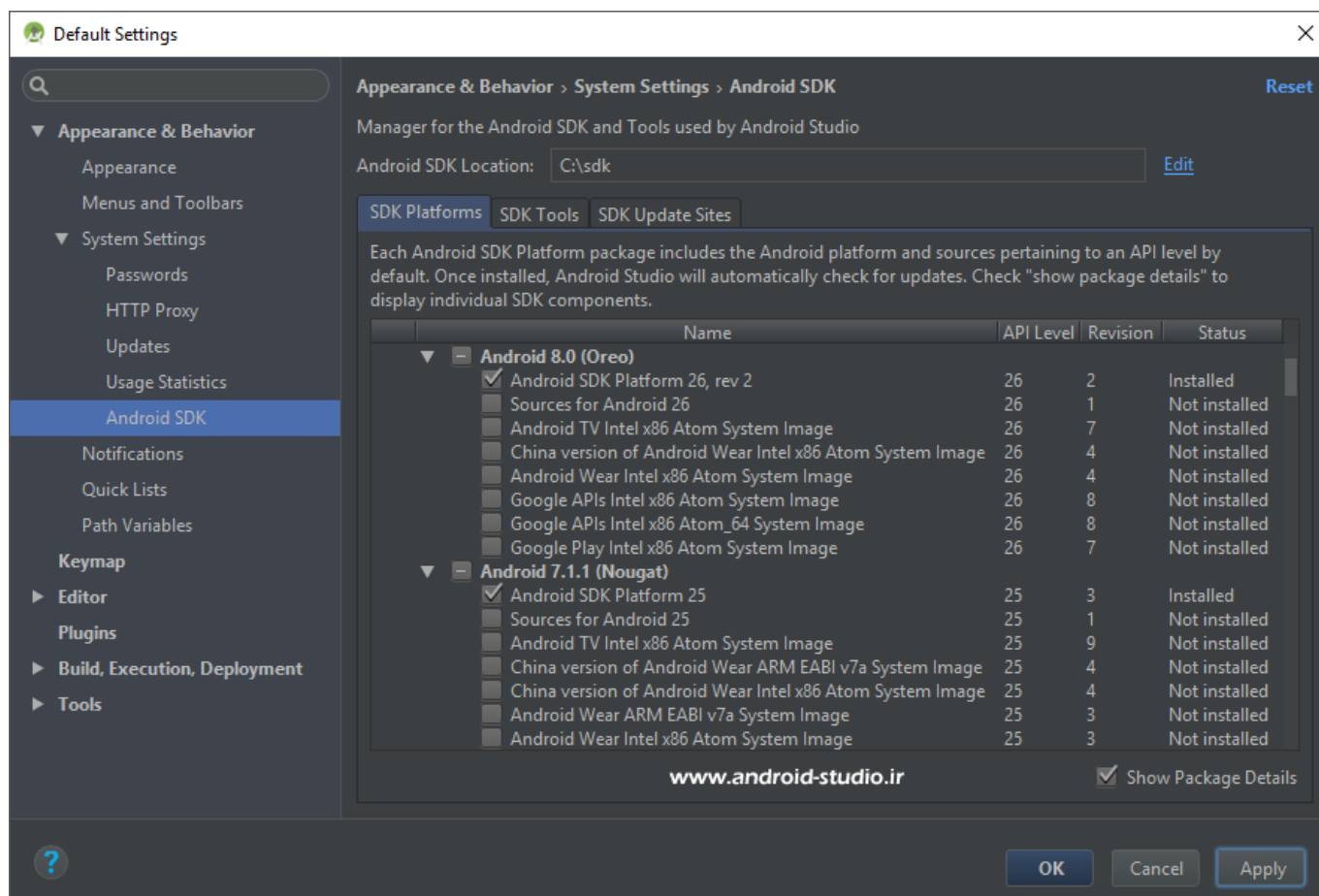
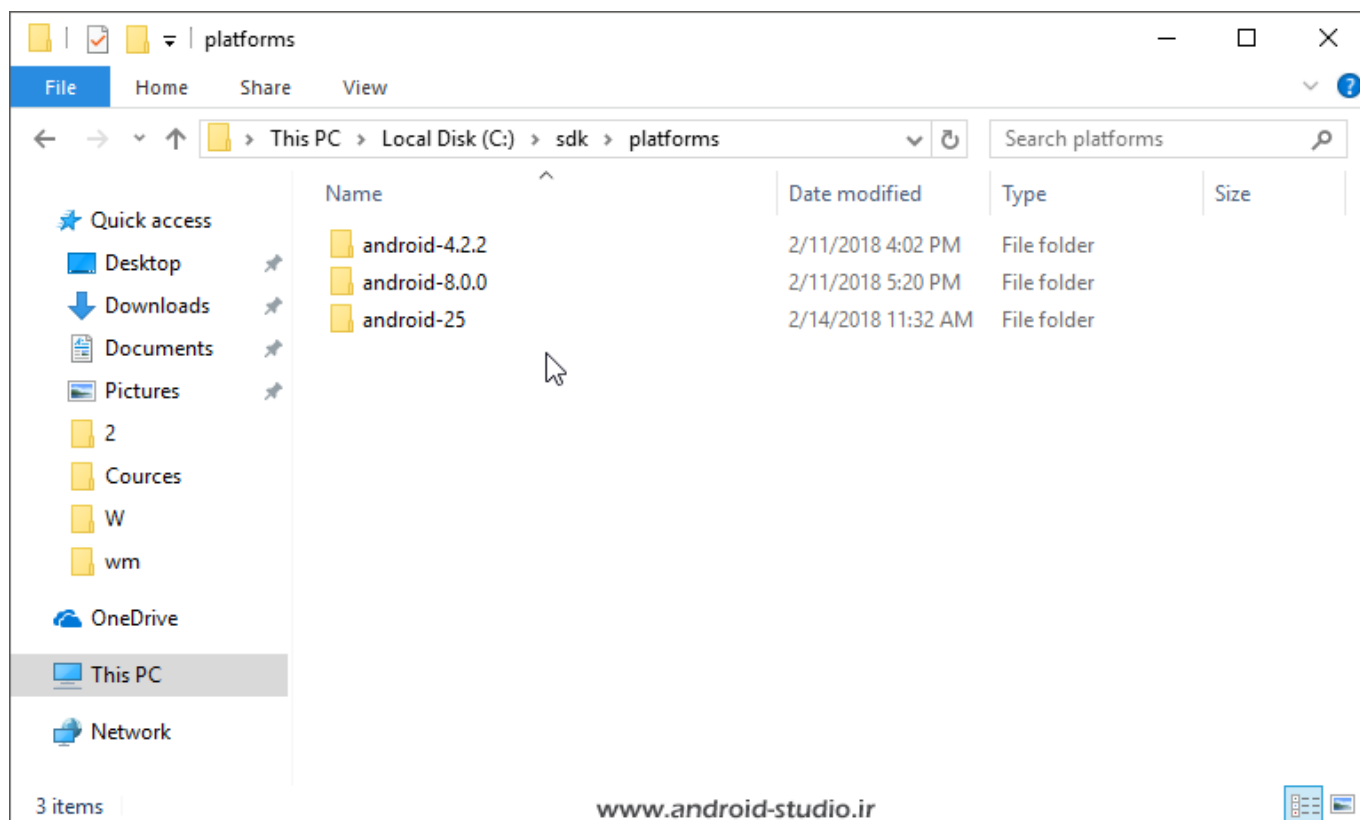






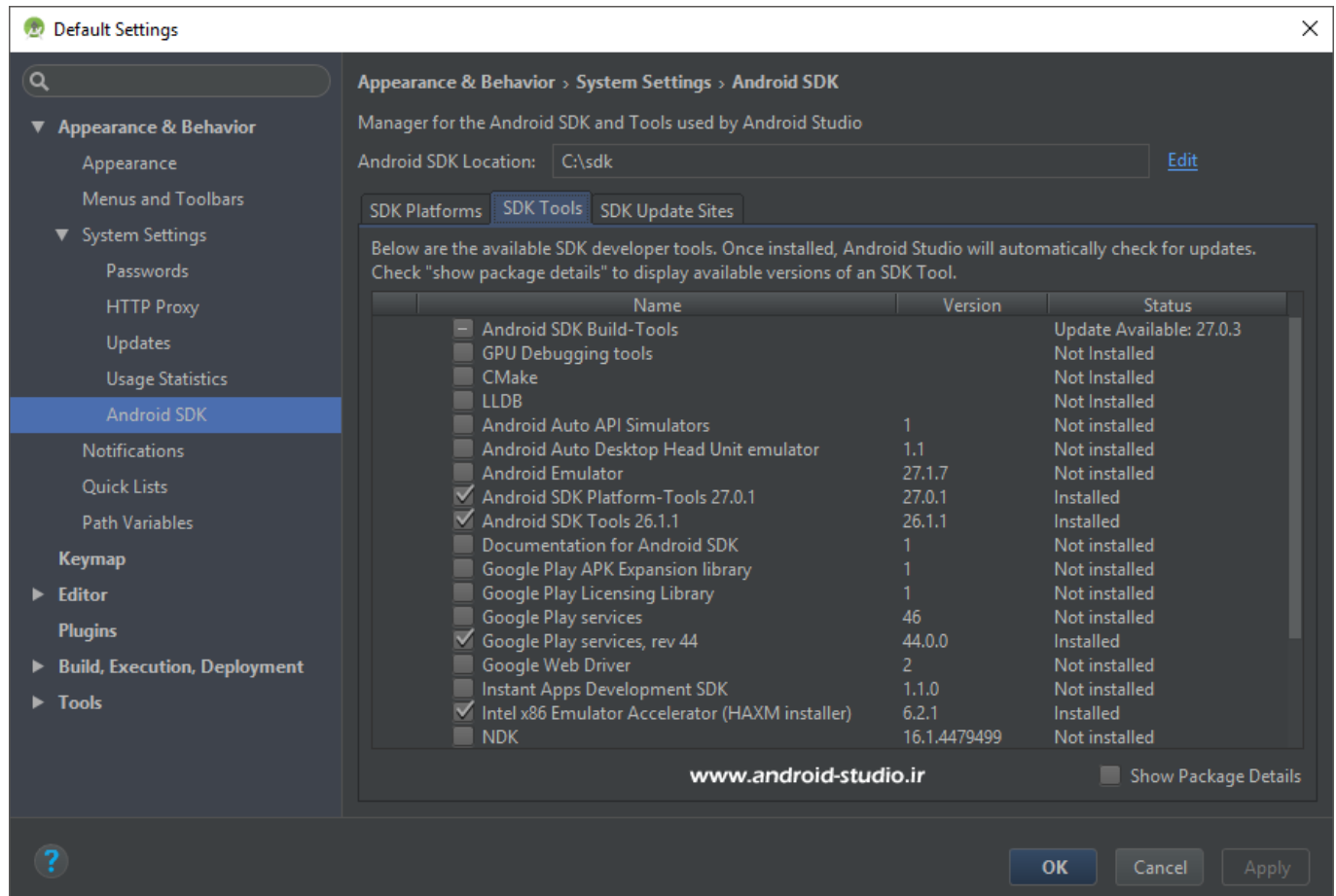
ملاحظه می کنید فایل platform-25_r3.zip در حال دانلود است. بعد از اتمام دانلود، پیغامی مبنی بر اتمام عملیات نصب نمایش داده می شود که علاوه بر آن محل نصب را نیز اعلام می کند:



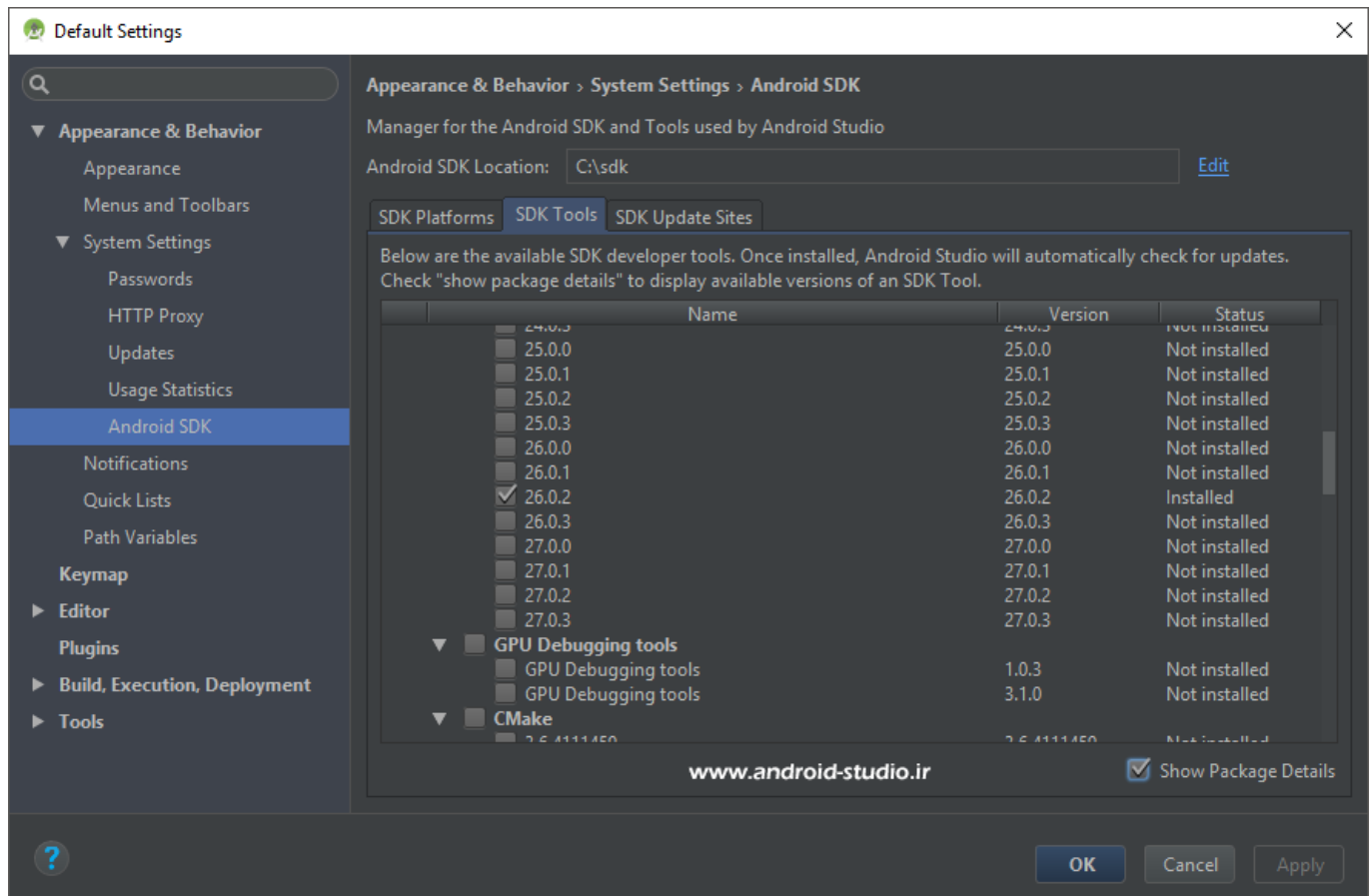




عملیاتی که آنلاین صورت می گیرد نیاز به بسته و باز کردن مجدد SDK Manager ندارد و پس از پایان دانلود، لیست بروز می شود. در تصویر بالا Platform 25 در ستون Status برچسب Installed خورده است.



در تب Sdk Tools نیز با بروزرسانی آنلاین لیست ابزار، گزینه های جدیدی مانند CMake و NDK اضافه شده که فعلا نیاز نداریم. برای SDK Build-Tools نسخه جدیدتر از آنچه قبلا آفلاین نصب کردم منتشر شده که با انتخاب Show Package Details لیست کامل نسخه ها را نشان می دهد:



جهت حذف ابزار از طریق SDK Manager کافیست تیک گزینه ای که قبلا نصب شده را بردارید. علامت ضربدر در سمت چپ نمایش داده می شود که با Apply کردن، حذف انجام می شود.

نکته: به تصویر صفحه ۵۷ دقت کنید. در حین دریافت فایل، لینک مستقیم آن نیز نمایش داده می شود. اگر به هر دلیلی دریافت و نصب خودکار با موفقیت انجام نشد و یا مایل به دانلود ابزار از وبسایت های ثانویه نبودید، کافیست لینک فایل که به پسوند zip. ختم شده را کپی و به صورت عادی دانلود کرده (البته با آی پی غیر از ایران) و به شیوه آفلاین نصب کنید. البته اگر این مراحل را به قصد دریافت لینک فایل جهت استفاده آفلاین انجام می دهید توجه داشته باشید باید مربوط به نصب باشد نه بروزرسانی. هنگامی که ابزار را در SDK Manager بروزرسانی می کنیم فایل دانه فقط شامل بخشی از آن ابزار است که می بایست در نسخه موجود جایگزین شود. بنابراین ابتدا باید ابزار مدنظر را از SDK حذف کرده، سپس اقدام به نصب نسخه جدید کنیم تا لینک نسخه کامل را در اختیار ما قرار دهد.

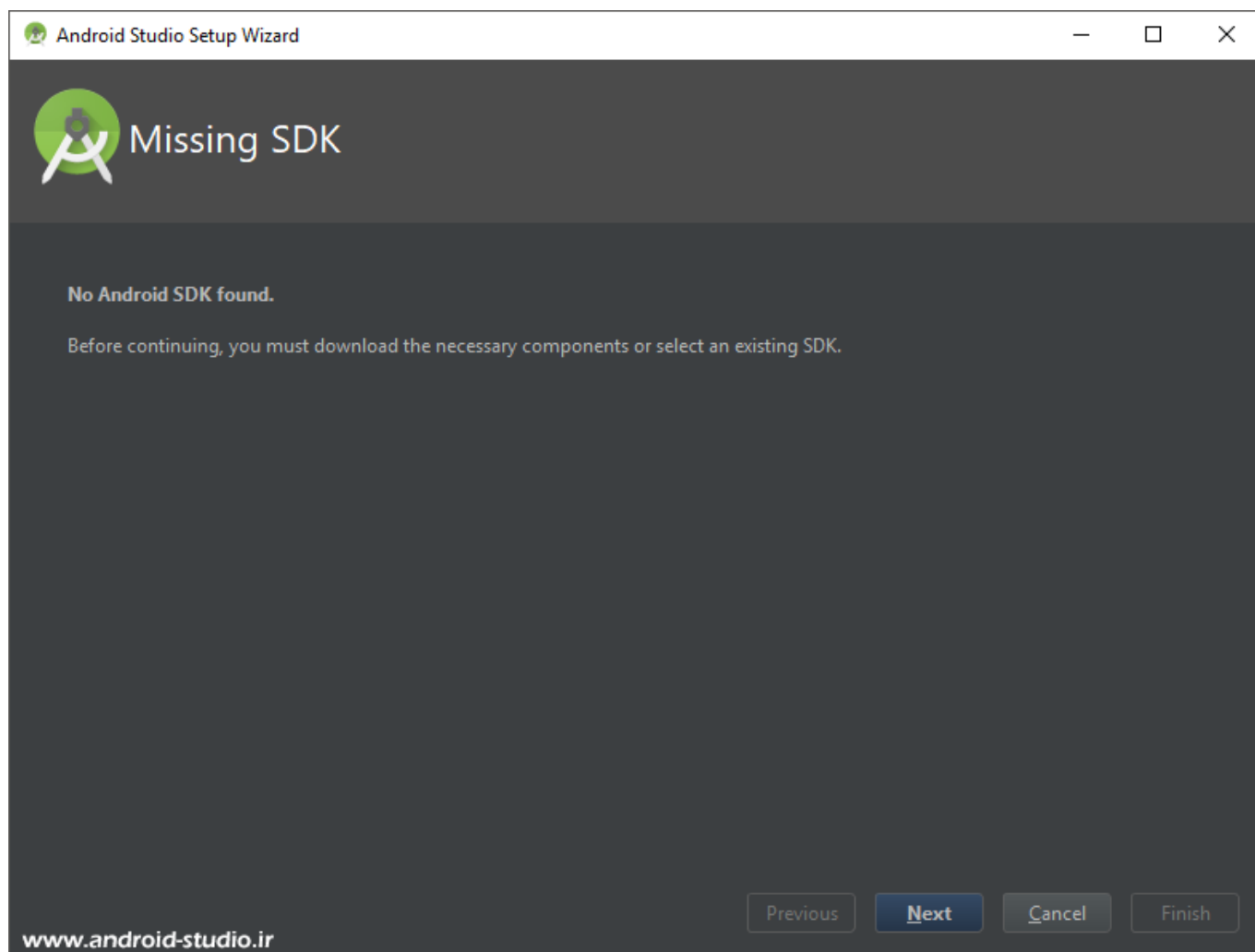


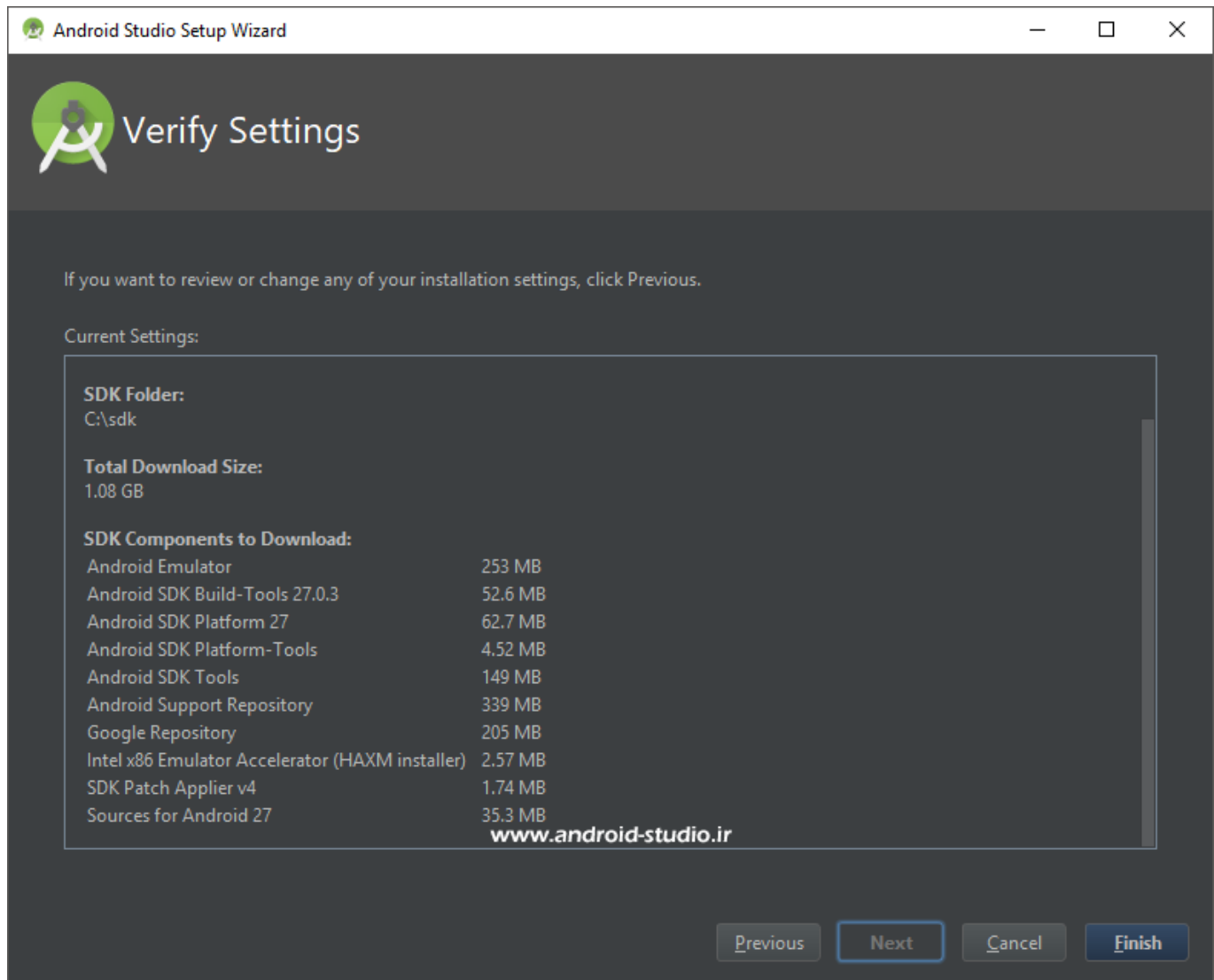
توضیحات مربوط به ابزار:

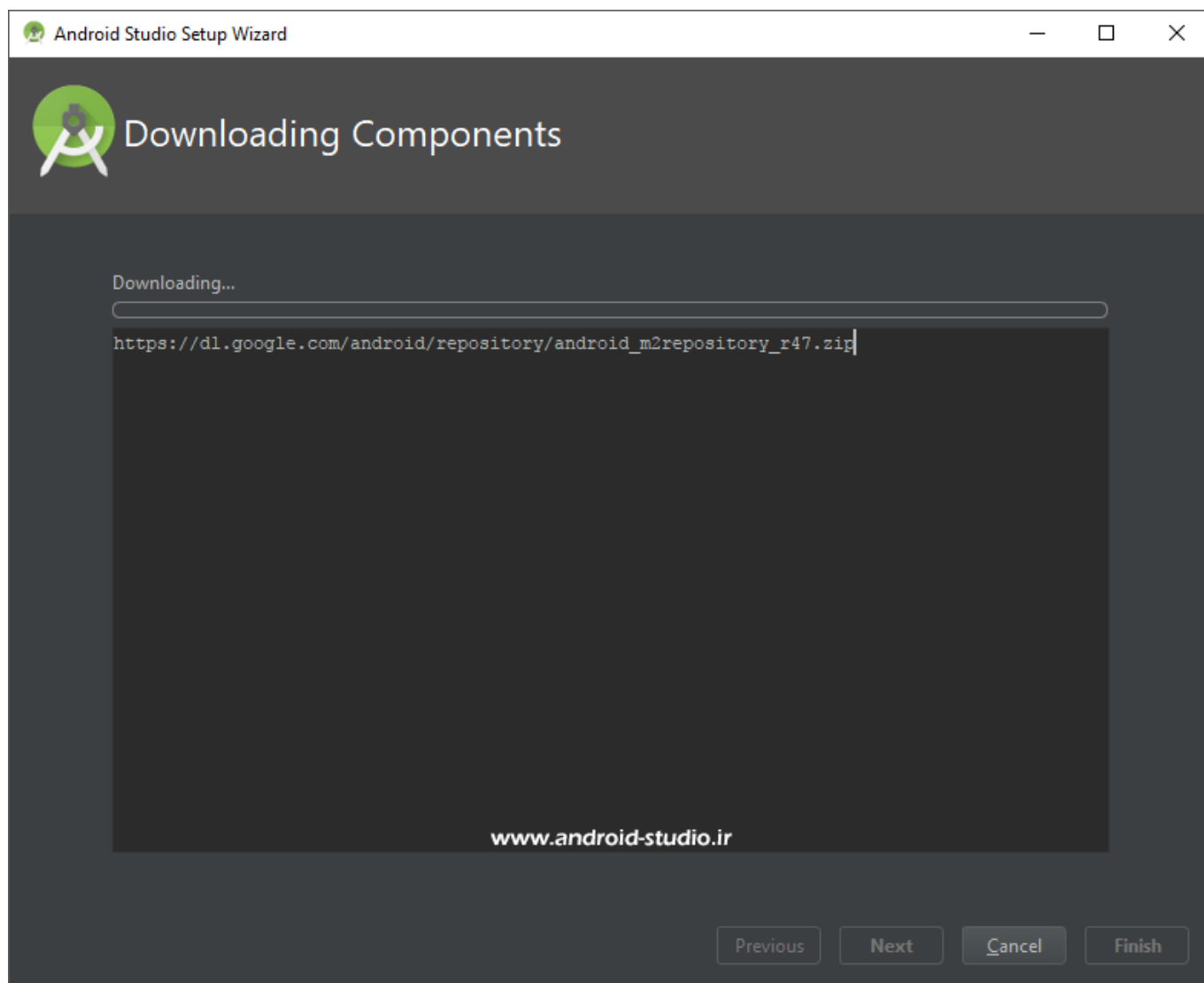
- ❖ **Google Repository و Android Support Repository** : مجموعه کتابخانه هایی هستند که در حین توسعه اپلیکیشن توسط اندروید استودیو استفاده می شوند.
- ❖ **Google USB Driver** : درایور دیوایس های اندرویدی NEXUS است (گوشی یا تبلت هایی که با برند نکسوس، به سفارش گوگل و توسط کمپانی هایی مانند الجی، سامسونگ و ... تولید می شوند). به وسیله این درایور می توانید دیوایس نکسوس را به رایانه متصل کرده و پروژه اندرویدی خود را روی آن اجرا، تست و خطایابی (دیباگ) کنید. برای سایر گوشی ها و تبلت های اندروید لازم است درایور کمپانی مربوطه را از وب سایت آن دانلود و نصب کنید. به عنوان مثال درایور شرکت الجی که قبلا آنرا دانلود کرده ام LGMobileDriver_WHQL_Ver_4.2.0.exe نام دارد (البته در اکثر موارد دیوایس های غیر نکسوس نیز با Google USB Driver سازگار بوده و نیازی به دانلود و نصب درایور اختصاصی شرکت سازنده نیست و بهتر است ابتدا با USB Driver تست کنید).
- ❖ **Build-Tools** : همانطور که از نامش پیداست برای بیلد شدن پروژه استفاده می شود.
- ❖ **Platform-Tools** : شامل مجموعه ای از ابزار مختلف مورد نیاز Platform ها.
- ❖ **SDK-Tools** : شامل ابزار ضروری برای ساخت پروژه اندرویدی.

« نصب آنلاین SDK

در قسمت قبل، ابتدا sdk را به صورت آفلاین آماده کرده و سپس اندروید استودیو را نصب کردیم. اما در نصب آنلاین نیازی به ساخت دستی sdk و ابزار درون آن نیست. لذا ابتدا توسط ابزار تغییر IP که سرعت مناسبی داشته باشد، آی پی ایران را تغییر داده و سپس مراحل نصب اندروید استودیو را انجام می دهیم. تا مرحله تعیین محل نصب SDK با روش قبل تفاوتی ندارد. در این مرحله کافیسیت یک فولدر خالی مانند C:\sdk ایجاد کرده و مسیر را به اندروید استودیو معرفی کنیم. در مرحله بعد برنامه اعلام می کند که مسیر معرفی شده شامل ابزار SDK نیست و موارد ضروری باید دانلود شود:







به این ترتیب عملیات دانلود آغاز شده و پس از دانلود حدود ۱.۰۸ گیگابایت (طبق تصویر) SDK کامل شده و اندروید استودیو آماده اجراست.

با ارائه انتقادات و پیشنهادات خود، ما را در ارائه آموزش های بهتر یاری فرمائید.

www.android-studio.ir