CASIO®

DT-X400 Series

Quick Start Guide

This document is a guide book for DT-X400 application developers.

Ver. 1.06

Cautions

- No part of this document may be produced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without the express written permission of CASIO Computer Co., Ltd. in Tokyo Japan.
- Information in this document is subject to change without advance notice.
- CASIO Computer Co., Ltd. makes no representations or warranties with respect to the contents or use of this manual and specifically disclaims any express or implied warranties of merchantability or fitness for any particular purpose.

- Table of contents -

1. Overview

This manual is a guidebook for developers that outlines the "DT-X400 Basic Development Kit" and describes how to develop applications that control CASIO extended functions such as keys and barcode scanners.

2. Basic Development Kit

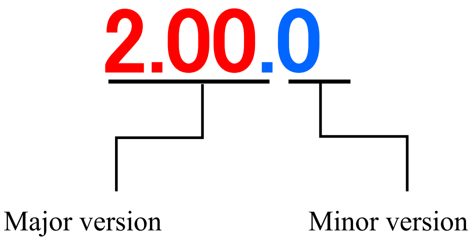
2.1 Structure

The following figure shows the structure of this kit.
xxxxxxxxxx in the following table means a software version.

Folder / File	Description
QuickStartGuide.lnk	Read this document first.
/MANUAL	This folder contains manuals.
QuickStartGuide.pdf	Quick start guide
SoftwareManual.pdf	Software manual
DeviceLibraryManual.pdf	Device library manual
HardwareManual.pdf	Hardware manual
/SOFTWARE	This folder contains software programs
/BDK	Basic development kit
/CasioAndroidAddons	Casio Addons for android
/bin	Program's folder
CASIOAndroidAddons0100020012.apk	CasioAndroidAddons (binary)
/manual	Manual's folder
CasioAndroidAddonsManual.pdf	Casio Android Addons manual
/DeviceLibrary	Device Library's folder
/javadoc	javadoc's folder
index.html	javadoc index file
etc.	Other files.
/bin	Program's folder
DeviceLibrary.aar	Device Library file (binary)
/samples	Sample file's folder
KeyLibrarySample.zip	KeyLibrary samples
SymbolScan.zip	ScannerLibrary samples
RangeScan.zip	
InverseScan.zip	
CenteringWindowScan.zip	
TriggerScan.zip	
ImageCapture.zip	
ImageCaptureEx.zip	
/TOOLS	Tools folder
/SecurityTools	SecurityTool folder
/manual	Manual's folder
SecurityToolManual.pdf	Security tool manual
/bin	bin
SecurityToolsxxxxxxxxxx.apk	Launcher
ActiveMenuxxxxxxxxxx.apk	Active menu
LoginSettingxxxxxxxxxx.apk	Login security
UserAccountEditorxxxxxxxxxx.apk	User account editor
SecurityServicexxxxxxxxxx.apk	Authentication limit
AuthenticateServicexxxxxxxxxx.apk	Service
SettingsBlockerxxxxxxxxxx.apk	Settings blocker
SecurityToolsFileCreator.xlsm	Job User Account Creator
/SupportTools	SupportTool folder
/manual	Manual's folder
KittingManual.pdf	Kitting manual
FLDroidManual.pdf	FLDroid manual
/bin	Program's folder
KitData.xls	Kitting scenario file
/OSUpdateService	OSUpdateService folder
/bin/OSUpdateService0100010004.apk	OS Update Service
/samples/OSUpdateSample.zip	OS Update Sample

2.2 Version

The version number of this development kit consists of a combination of major version and minor version, as shown below.



The major number represents the version of DeviceLibrary (i.e. DeviceLibrary.aar).
So, when a newer version of development kit is released, if it includes newer version of DeviceLibrary, its major version number is increased, and the minor version is returned to zero.
If changes other than DeviceLibrary (i.e. manuals, tools, etc.) are included, its minor version number is increased.

2.3 Notes

Notes for using this development kit are as follows.

There is a dependency between the version of Android Studio and the version of the Basic Development Kit. For details, refer to "3.1 Recommended environment (p.5)".

3. Development environment

3.1 Recommended environment

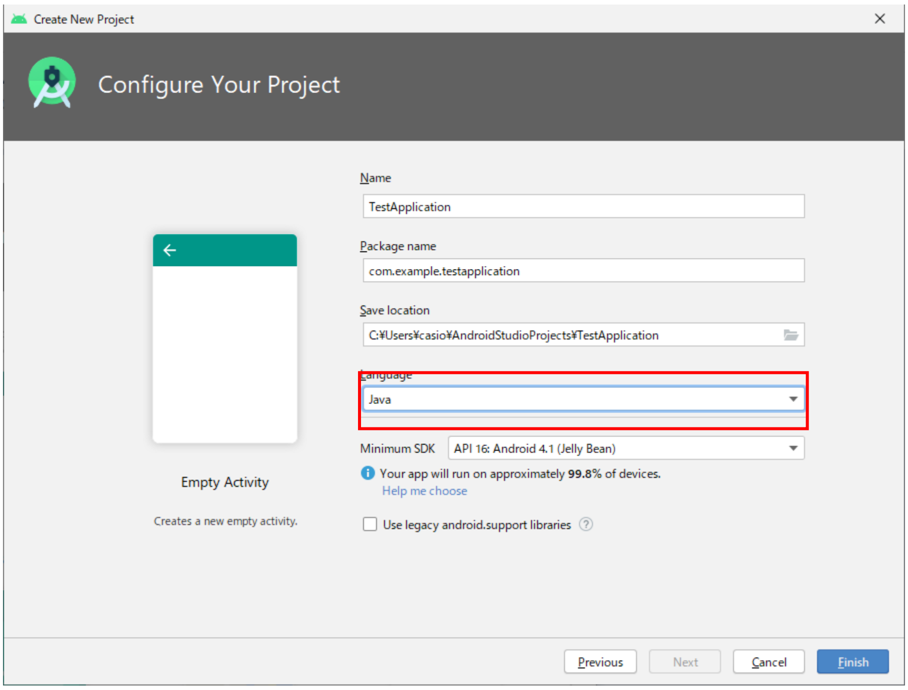
To develop the application for DT-X400, it needs the development environment of Android.

Development language	Development platform (Recommended)
Java	Android Studio 3.0 or above Android SDK (API level 26 - 27 above) Google USB driver JDK7 or above (Bundled with Android Studio)

Note!
The basic development kit was confirmed to work with the following Android Studio combination.
Use the following combination or above version.
However, there is a possibility that something wrong with work if use the latest unconfirmed Android Studio version.
If there is something wrong with work, try with the confirmed below combination.
All released versions of Android Studio are available at the following web site.
https://developer.android.com/studio/archive

Software	Version
Android Studio	4.1.1
Gradle version (Android Studio)	6.5
Android Plugin Version (Android Studio)	4.1.1
Development Kit	2.014.x

The default development language for Android Studio 3.4 or above is Kotlin.
Select Java in case of creating a new project.



Create New Project

Configure Your Project

Name: TestApplication

Package name: com.example.testapplication

Save location: C:¥Users¥casio¥AndroidStudioProjects¥TestApplication

Language: Java

Minimum SDK: API 16: Android 4.1 (Jelly Bean)

Your app will run on approximately **99.8%** of devices.
Help me choose

□ Use legacy android.support libraries

Empty Activity

Creates a new empty activity.

3.2 Required system

[Windows]

- Microsoft® Windows® 8/10 (64bit)
- 4GB RAM minimum, 8GB RAM recommended
- 2GB of available disk space minimum, 4GB Recommended (500MB for IDE + 1.5GB for Android SDK and emulator system image)
- 1280 x 800 minimum screen resolution

[Mac]

- Mac® OS X® 10.10 (Yosemite) or higher, up to 10.14 (macOS Mojave)
- 4GB RAM minimum, 8GB RAM recommended
- 2GB of available disk space minimum, 4GB Recommended (500MB for IDE + 1.5GB for Android SDK and emulator system image)
- 1280 x 800 minimum screen resolution

[Linux]

- GNOME or KDE desktop
- Tested on gLinuxbased on Debian (4.19.67-2rodete2)
- GNU C Library (glibc) 2.19 or later
- 4GB RAM minimum, 8GB RAM recommended
- 2GB of available disk space minimum, 4GB Recommended (500MB for IDE + 1.5GB for Android SDK and emulator system image)
- 1280 x 800 minimum screen resolution

Note!
The explanation after the next chapter is all for Windows (64bit). If you are using Mac or Linux as a development environment, you are responsible for building the environment at your own risk.
If using Windows (32bit), make sure Android Studio is working properly on the PC before installing this BDK. The final version of Android Studio for Windows (32bit) is 3.6.

3.3 Construction steps

The followings are the basic steps to construct the development environment.
Due to version upgrade of Android Studio, environment may not be constructed by the following procedure. In that case, refer to the procedure described on Google's website etc. and construct the environment.

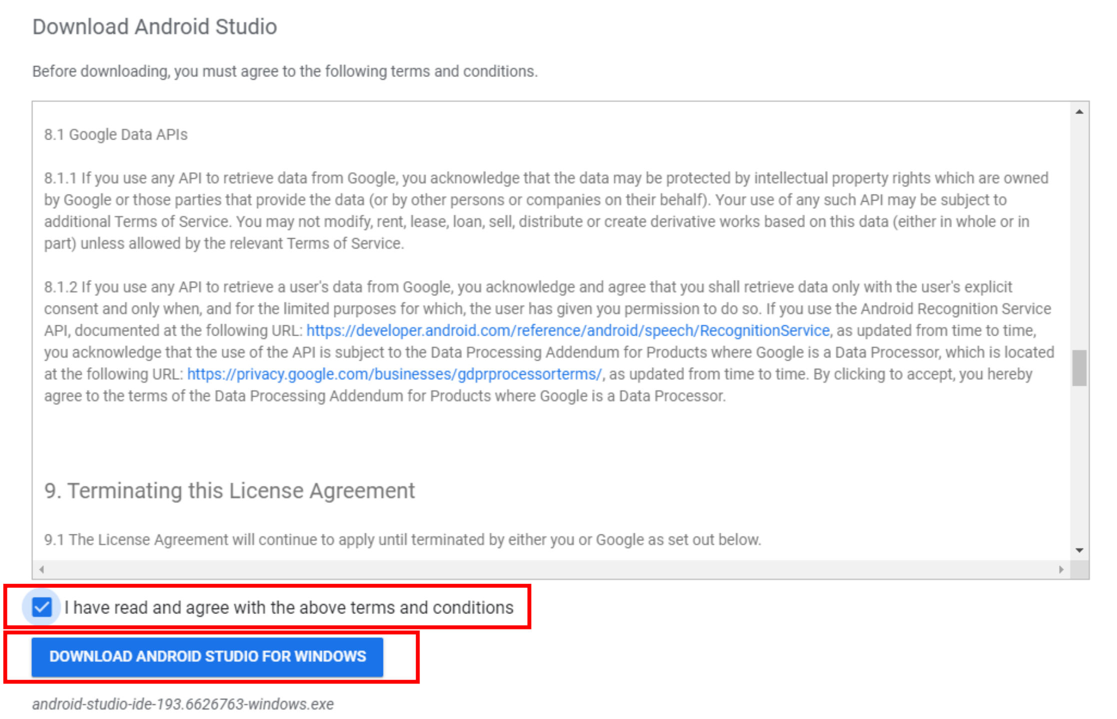
(1) Download and install Android Studio
(2) CASIO extended function control

3.3.1 Android Studio

Download and install

Android Studio can be downloaded from the following site of Google.
https://developer.android.com/studio/

Agree to the dialog shown below and, then start downloading.
Executing the downloaded file, follow the instructions to proceed with the installation.



Download Android Studio

Before downloading, you must agree to the following terms and conditions.

8.1 Google Data APIs

8.1.1 If you use any API to retrieve data from Google, you acknowledge that the data may be protected by intellectual property rights which are owned by Google or those parties that provide the data (or by other persons or companies on their behalf). Your use of any such API may be subject to additional Terms of Service. You may not modify, rent, lease, loan, sell, distribute or create derivative works based on this data (either in whole or in part) unless allowed by the relevant Terms of Service.

8.1.2 If you use any API to retrieve a user's data from Google, you acknowledge and agree that you shall retrieve data only with the user's explicit consent and only when, and for the limited purposes for which, the user has given you permission to do so. If you use the Android Recognition Service API, documented at the following URL: https://developer.android.com/reference/android/speech/RecognitionService, as updated from time to time, you acknowledge that the use of the API is subject to the Data Processing Addendum for Products where Google is a Data Processor, which is located at the following URL: https://privacy.google.com/businesses/gdprprocessorterms/, as updated from time to time. By clicking to accept, you hereby agree to the terms of the Data Processing Addendum for Products where Google is a Data Processor.

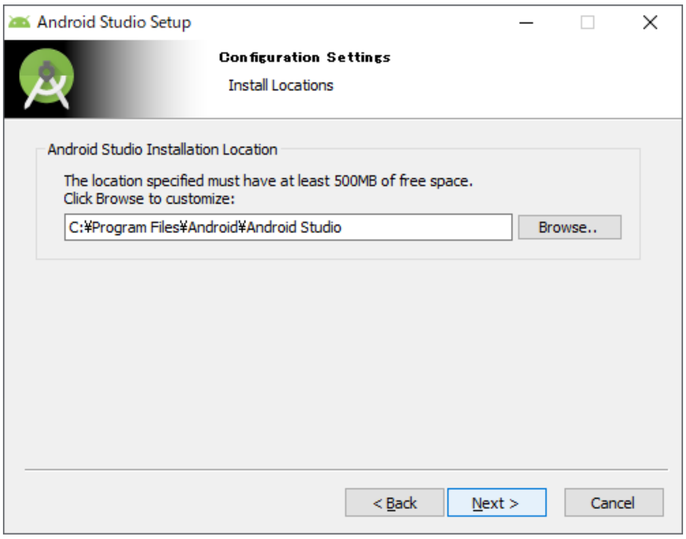
9. Terminating this License Agreement

9.1 The License Agreement will continue to apply until terminated by either you or Google as set out below.

☑ I have read and agree with the above terms and conditions

android-studio-ide-193.6626763-windows.exe

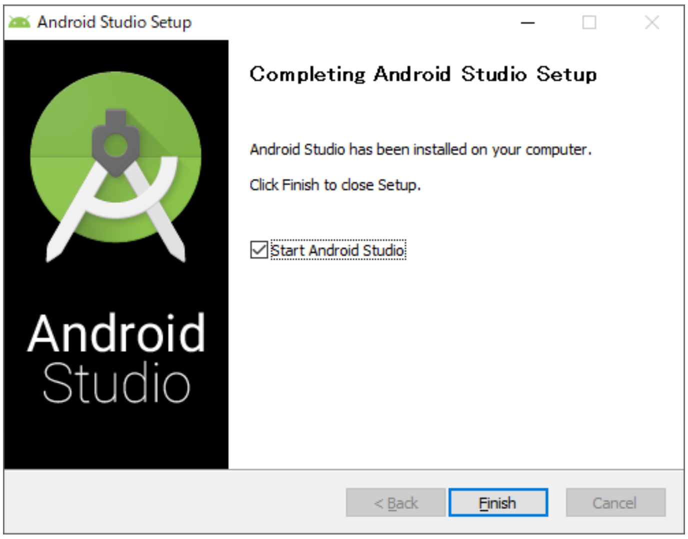
Prompted to specify the installation destination. Change it if necessary.



The location specified must have at least 500MB of free space.
Click Browse to customize:

C:¥Program Files¥Android¥Android Studio

When the below dialog is displayed, installation of Android Studio is completed.



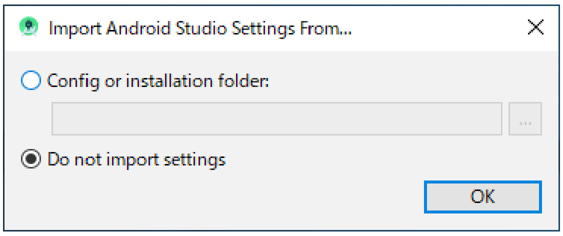
Completing Android Studio Setup

Android Studio has been installed on your computer.

Click Finish to close Setup.

☑ Start Android Studio

When start the installed Android Studio first time, select whether to use the previous setting or not. If do not want to use the previous setting, select "Do not import settings".

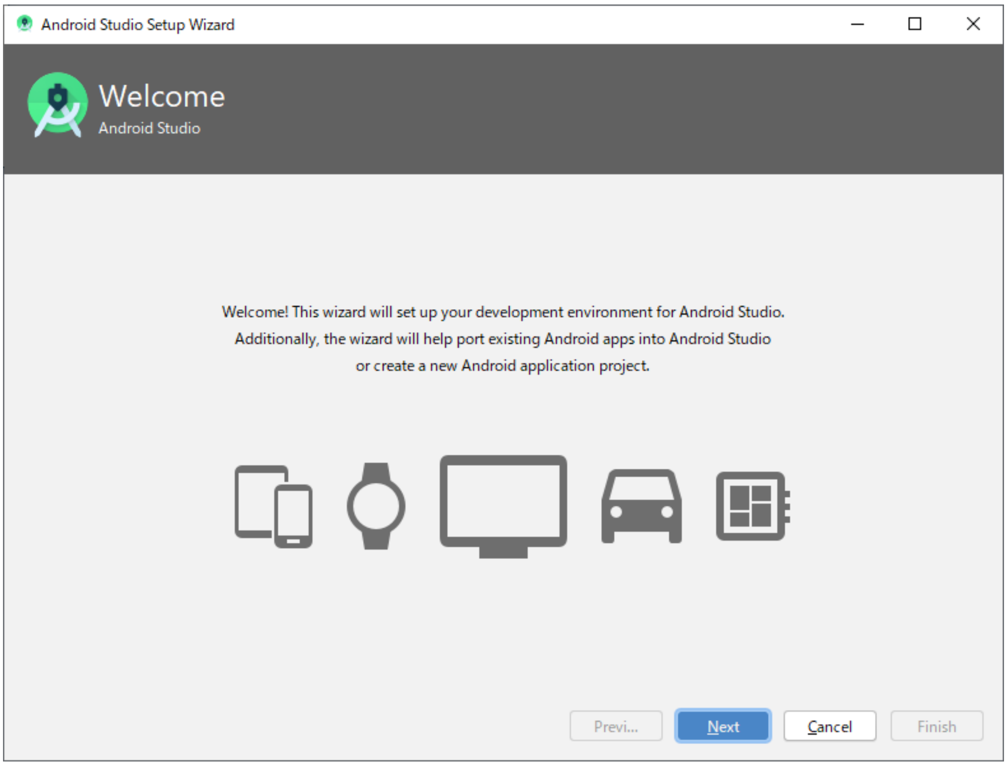


Import Android Studio Settings From...

○ Config or installation folder:

● Do not import settings

The Setup Wizard will start, so follow the instructions.



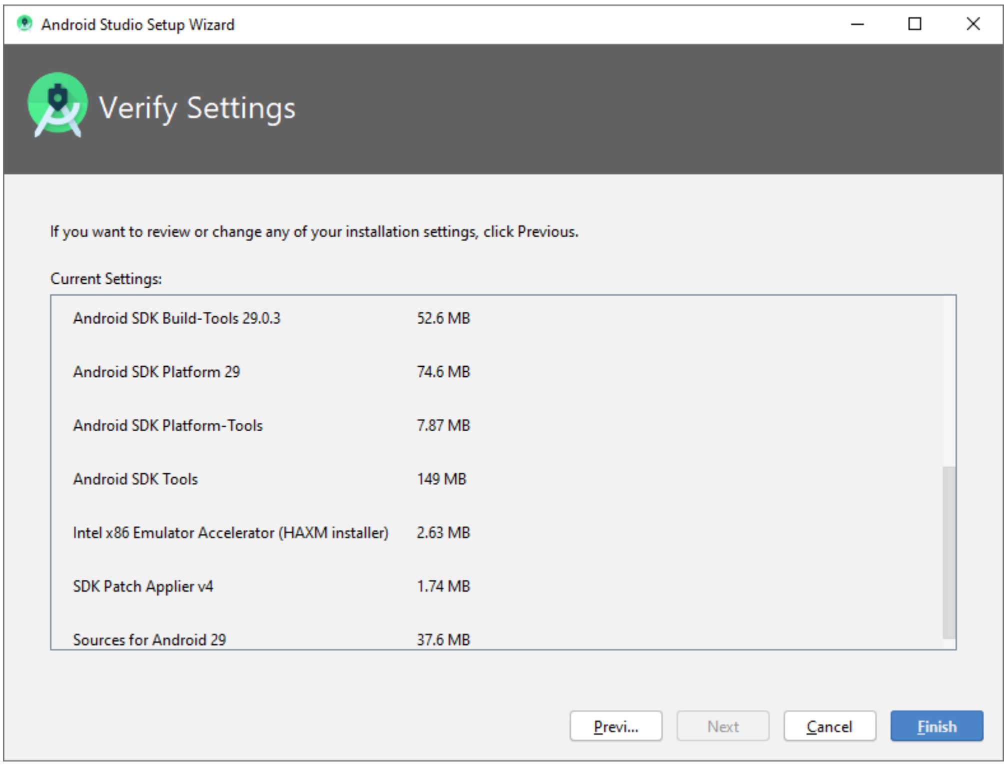
Android Studio Setup Wizard

Welcome

Android Studio

Welcome! This wizard will set up your development environment for Android Studio.
Additionally, the wizard will help port existing Android apps into Android Studio
or create a new Android application project.

If do not have any problem after confirming "Finish".

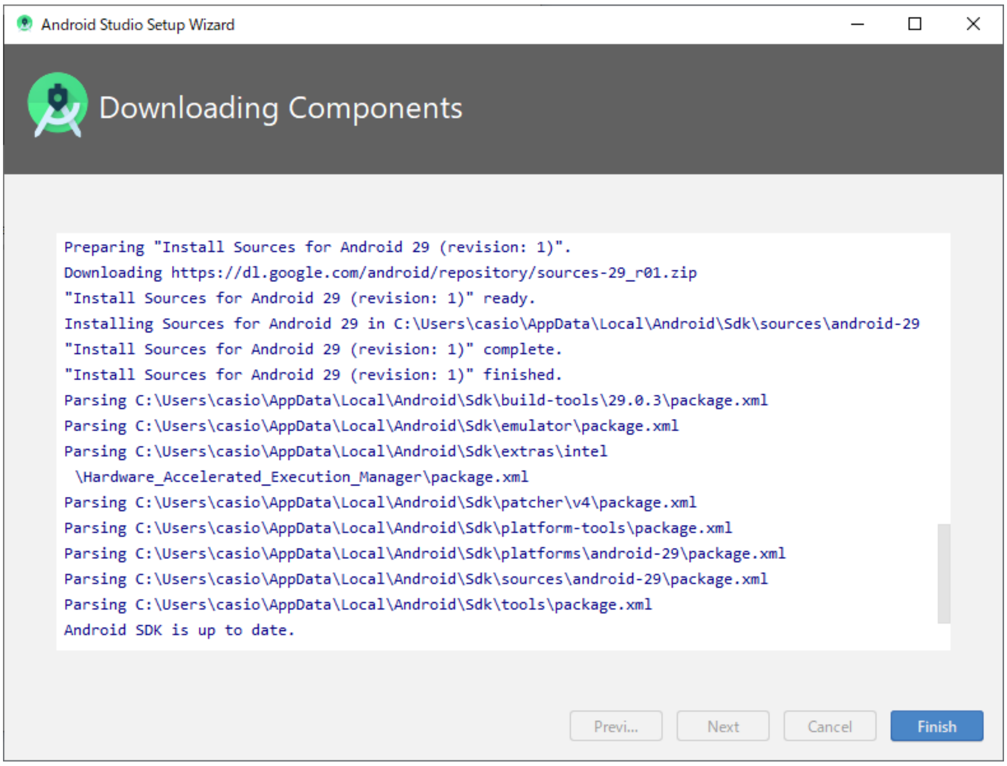


If you want to review or change any of your installation settings, click Previous.

Current Settings:

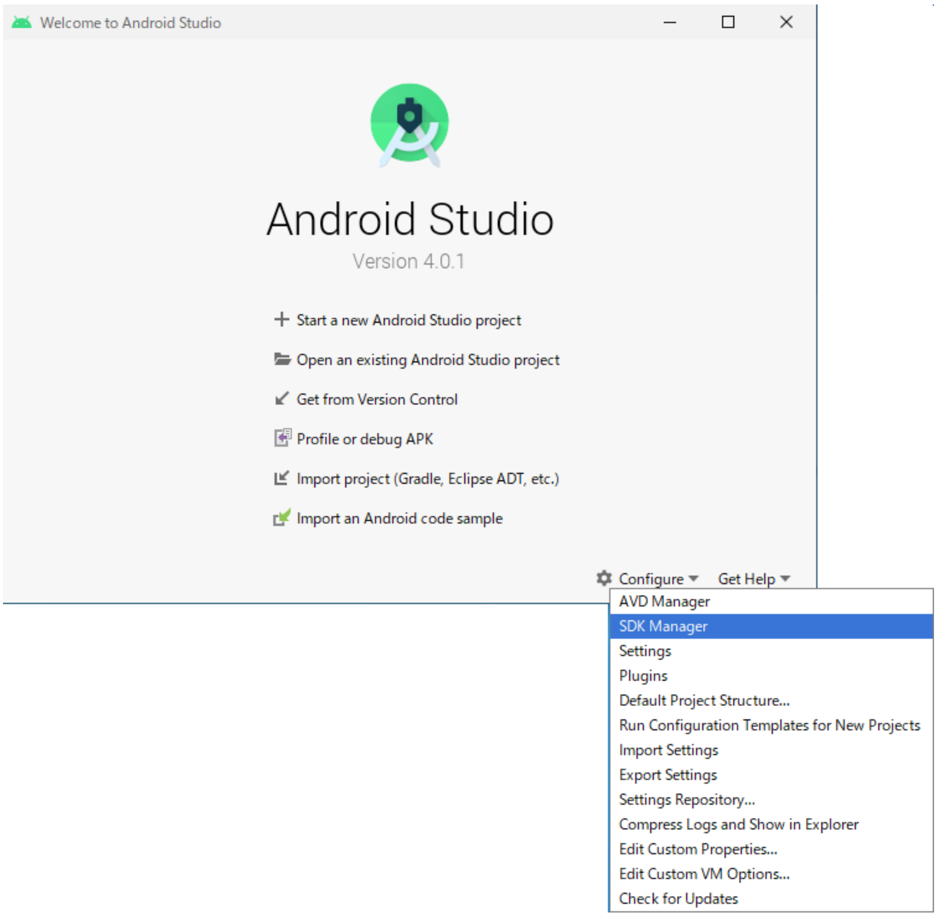
Android SDK Build-Tools 29.0.3	52.6 MB
Android SDK Platform 29	74.6 MB
Android SDK Platform-Tools	7.87 MB
Android SDK Tools	149 MB
Intel x86 Emulator Accelerator (HAXM installer)	2.63 MB
SDK Patch Applier v4	1.74 MB
Sources for Android 29	37.6 MB

Downloading Components begins. When completed, Android Studio will startup.

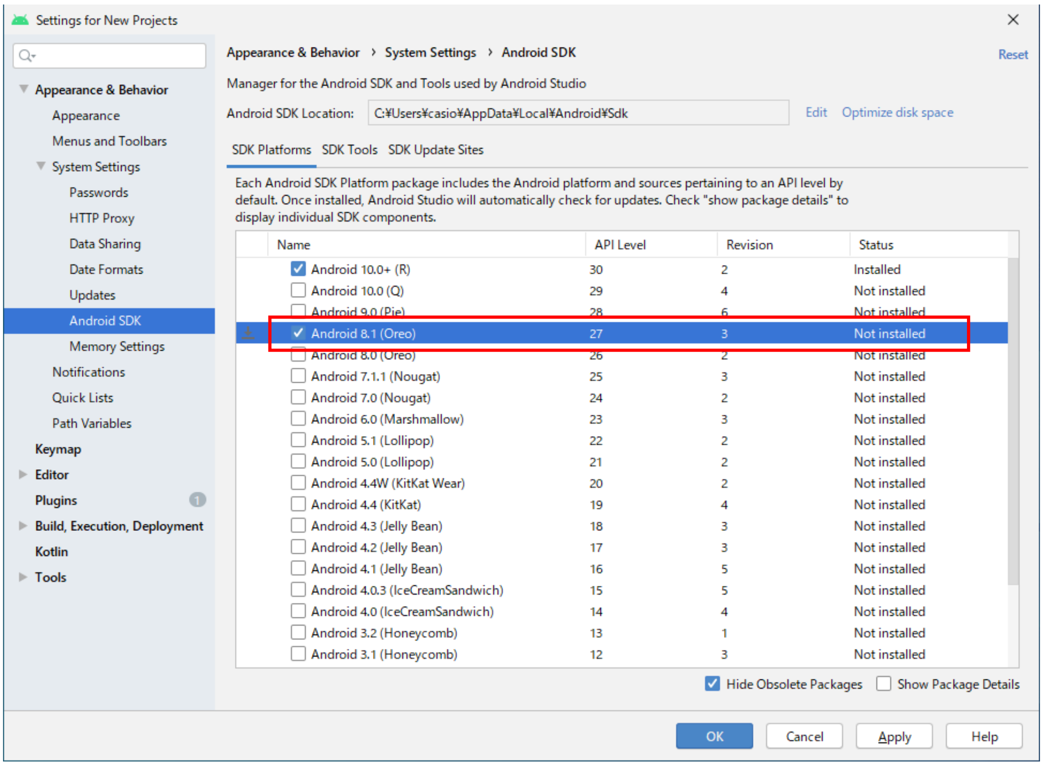


```
Preparing "Install Sources for Android 29 (revision: 1)".
Downloading https://dl.google.com/android/repository/sources-29_r01.zip
"Install Sources for Android 29 (revision: 1)" ready.
Installing Sources for Android 29 in C:\Users\casio\AppData\Local\Android\Sdk\sources\android-29
"Install Sources for Android 29 (revision: 1)" complete.
"Install Sources for Android 29 (revision: 1)" finished.
Parsing C:\Users\casio\AppData\Local\Android\Sdk\build-tools\29.0.3\package.xml
Parsing C:\Users\casio\AppData\Local\Android\Sdk\emulator\package.xml
Parsing C:\Users\casio\AppData\Local\Android\Sdk\extras\intel
 \Hardware_Accelerated_Execution_Manager\package.xml
Parsing C:\Users\casio\AppData\Local\Android\Sdk\patcher\v4\package.xml
Parsing C:\Users\casio\AppData\Local\Android\Sdk\platform-tools\package.xml
Parsing C:\Users\casio\AppData\Local\Android\Sdk\platforms\android-29\package.xml
Parsing C:\Users\casio\AppData\Local\Android\Sdk\sources\android-29\package.xml
Parsing C:\Users\casio\AppData\Local\Android\Sdk\tools\package.xml
Android SDK is up to date.
```

Launch the Android Studio, and chose "SDK Manager" from Configure tab.


Welcome to Android Studio
Android Studio
Version 4.0.1
+ Start a new Android Studio project
Open an existing Android Studio project
Get from Version Control
Profile or debug APK
Import project (Gradle, Eclipse ADT, etc.)
Import an Android code sample
Configure
Get Help
AVD Manager
SDK Manager
Settings
Plugins
Default Project Structure...
Run Configuration Templates for New Projects
Import Settings
Export Settings
Settings Repository...
Compress Logs and Show in Explorer
Edit Custom Properties...
Edit Custom VM Options...
Check for Updates

Select "SDK Platform" tab, then check "Android 8.1 (API Level 27)".



Manager for the Android SDK and Tools used by Android Studio

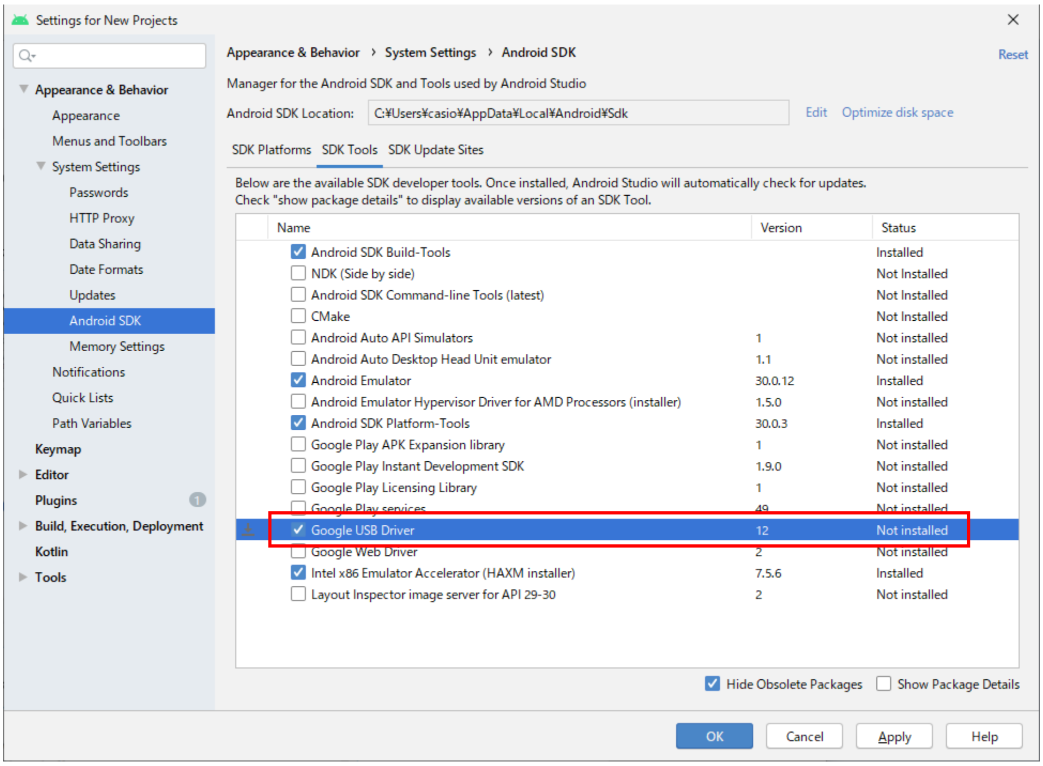
Android SDK Location: C:¥Users¥casio¥AppData¥Local¥Android¥Sdk

Each Android SDK Platform package includes the Android platform and sources pertaining to an API level by default. Once installed, Android Studio will automatically check for updates. Check "show package details" to display individual SDK components.

Name	API Level	Revision	Status
☑ Android 10.0+ (R)	30	2	Installed
☐ Android 10.0 (Q)	29	4	Not installed
☐ Android 9.0 (Pie)	28	6	Not installed
☑ Android 8.1 (Oreo)	27	3	Not installed
☐ Android 8.0 (Oreo)	26	2	Not installed
☐ Android 7.1.1 (Nougat)	25	3	Not installed
☐ Android 7.0 (Nougat)	24	2	Not installed
☐ Android 6.0 (Marshmallow)	23	3	Not installed
☐ Android 5.1 (Lollipop)	22	2	Not installed
☐ Android 5.0 (Lollipop)	21	2	Not installed
☐ Android 4.4W (KitKat Wear)	20	2	Not installed
☐ Android 4.4 (KitKat)	19	4	Not installed
☐ Android 4.3 (Jelly Bean)	18	3	Not installed
☐ Android 4.2 (Jelly Bean)	17	3	Not installed
☐ Android 4.1 (Jelly Bean)	16	5	Not installed
☐ Android 4.0.3 (IceCreamSandwich)	15	5	Not installed
☐ Android 4.0 (IceCreamSandwich)	14	4	Not installed
☐ Android 3.2 (Honeycomb)	13	1	Not installed
☐ Android 3.1 (Honeycomb)	12	3	Not installed

☑ Hide Obsolete Packages ☐ Show Package Details

Next, select "SDK Tools" tab and check "Google USB Driver".



Settings for New Projects

- Appearance & Behavior
  - Appearance
  - Menus and Toolbars
  - System Settings
    - Passwords
    - HTTP Proxy
    - Data Sharing
    - Date Formats
    - Updates
    - Android SDK
    - Memory Settings
  - Notifications
  - Quick Lists
  - Path Variables
- Keymap
- Editor
- Plugins
- Build, Execution, Deployment
- Kotlin
- Tools

Appearance & Behavior › System Settings › Android SDK

Manager for the Android SDK and Tools used by Android Studio

Android SDK Location: C:¥Users¥casio¥AppData¥Local¥Android¥Sdk

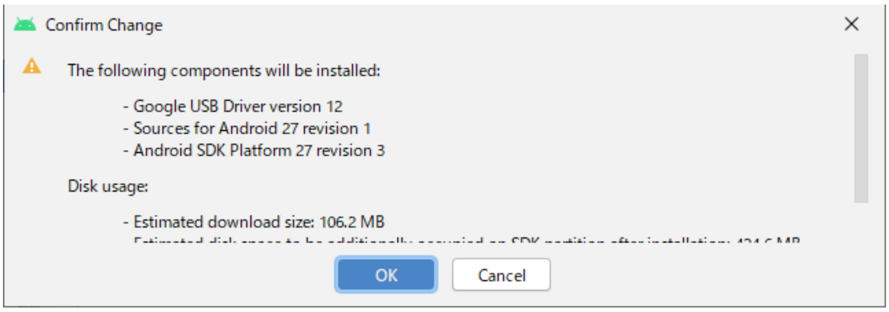
SDK Platforms | SDK Tools | SDK Update Sites

Below are the available SDK developer tools. Once installed, Android Studio will automatically check for updates.
Check "show package details" to display available versions of an SDK Tool.

Name	Version	Status
☑ Android SDK Build-Tools		Installed
□ NDK (Side by side)		Not Installed
□ Android SDK Command-line Tools (latest)		Not Installed
□ CMake		Not Installed
□ Android Auto API Simulators	1	Not installed
□ Android Auto Desktop Head Unit emulator	1.1	Not installed
☑ Android Emulator	30.0.12	Installed
□ Android Emulator Hypervisor Driver for AMD Processors (installer)	1.5.0	Not installed
☑ Android SDK Platform-Tools	30.0.3	Installed
□ Google Play APK Expansion library	1	Not installed
□ Google Play Instant Development SDK	1.9.0	Not installed
□ Google Play Licensing Library	1	Not installed
□ Google Play services	49	Not installed
☑ Google USB Driver	12	Not installed
□ Google Web Driver	2	Not installed
☑ Intel x86 Emulator Accelerator (HAXM installer)	7.5.6	Installed
□ Layout Inspector image server for API 29-30	2	Not installed

☑ Hide Obsolete Packages □ Show Package Details

The following dialog will be displayed, press "OK".



Confirm Change

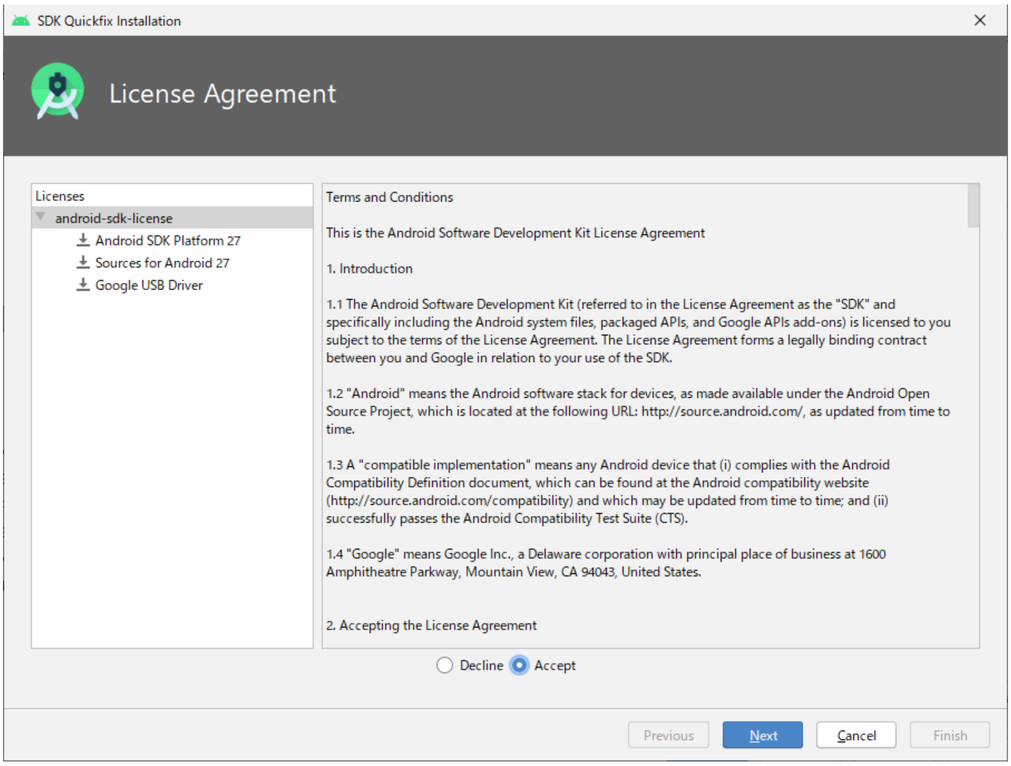
The following components will be installed:

- Google USB Driver version 12
- Sources for Android 27 revision 1
- Android SDK Platform 27 revision 3

Disk usage:

- Estimated download size: 106.2 MB
- Estimated disk space to be additionally occupied on SDK partition after installation: 424.6 MB

Accept the "License Agreement", then press "OK" to start installation.



Terms and Conditions

This is the Android Software Development Kit License Agreement

1. Introduction

1.1 The Android Software Development Kit (referred to in the License Agreement as the "SDK" and specifically including the Android system files, packaged APIs, and Google APIs add-ons) is licensed to you subject to the terms of the License Agreement. The License Agreement forms a legally binding contract between you and Google in relation to your use of the SDK.

1.2 "Android" means the Android software stack for devices, as made available under the Android Open Source Project, which is located at the following URL: http://source.android.com/, as updated from time to time.

1.3 A "compatible implementation" means any Android device that (i) complies with the Android Compatibility Definition document, which can be found at the Android compatibility website (http://source.android.com/compatibility) and which may be updated from time to time; and (ii) successfully passes the Android Compatibility Test Suite (CTS).

1.4 "Google" means Google Inc., a Delaware corporation with principal place of business at 1600 Amphitheatre Parkway, Mountain View, CA 94043, United States.

2. Accepting the License Agreement

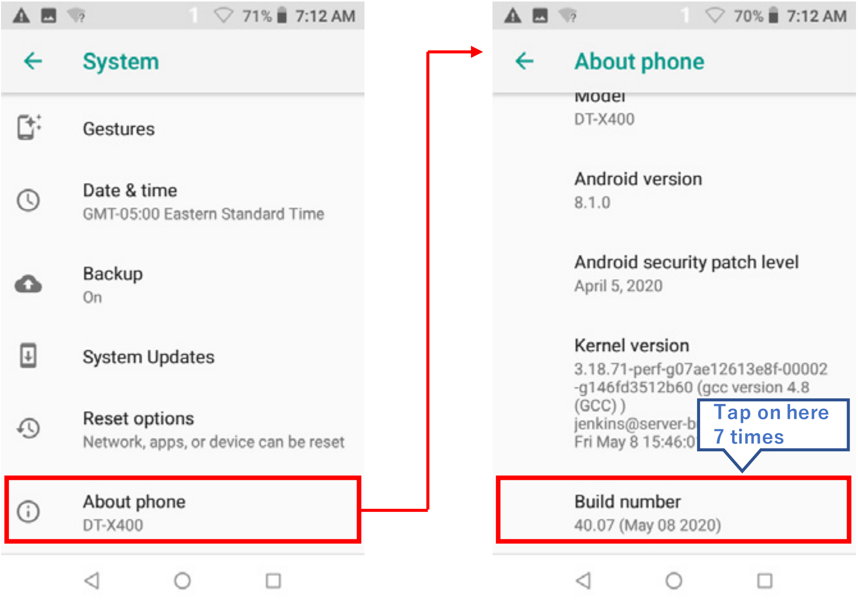
○ Decline ● Accept

The USB driver is stored in "extras\google\usb_driver" under "Android SDK Installation Location".
e.g.) C:\Users\xxx\AppData\Local\Android\Sdk\extras\google\usb_driver
(xxx: username of computer)

USB debugging

Since USB debugging is a function for developers, need to activate the Developer options by the following procedure first.

Open [Settings] -> [About phone] and tap "Build number" on the bottom line seven times, [Developer options] is added above [About phone].



71% 7:12 AM

System

Gestures

Date & time
GMT-05:00 Eastern Standard Time

Backup
On

System Updates

Reset options
Network, apps, or device can be reset

About phone
DT-X400

70% 7:12 AM

About phone

Model
DT-X400

Android version
8.1.0

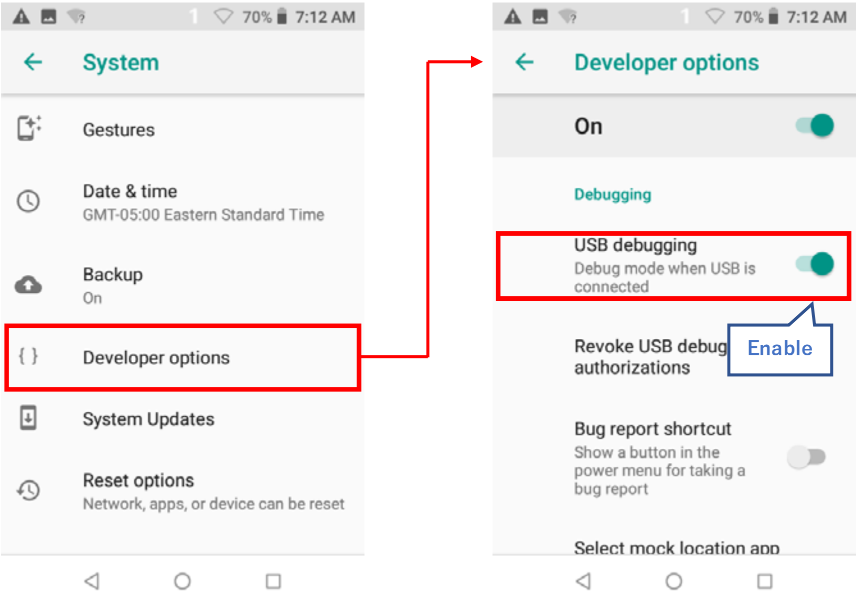
Android security patch level
April 5, 2020

Kernel version
3.18.71-perf-g07ae12613e8f-00002
-g146fd3512b60 (gcc version 4.8
(GCC))
jenkins@server-b
Fri May 8 15:46:0

Build number
40.07 (May 08 2020)

Tap on here
7 times

Open [Settings] -> [About phone] -> [Developer options] and enable "USB debugging".



70% 7:12 AM

System

Gestures

Date & time
GMT-05:00 Eastern Standard Time

Backup
On

Developer options

System Updates

Reset options
Network, apps, or device can be reset

70% 7:12 AM

Developer options

On

Debugging

USB debugging
Debug mode when USB is connected

Revoke USB debug
authorizations

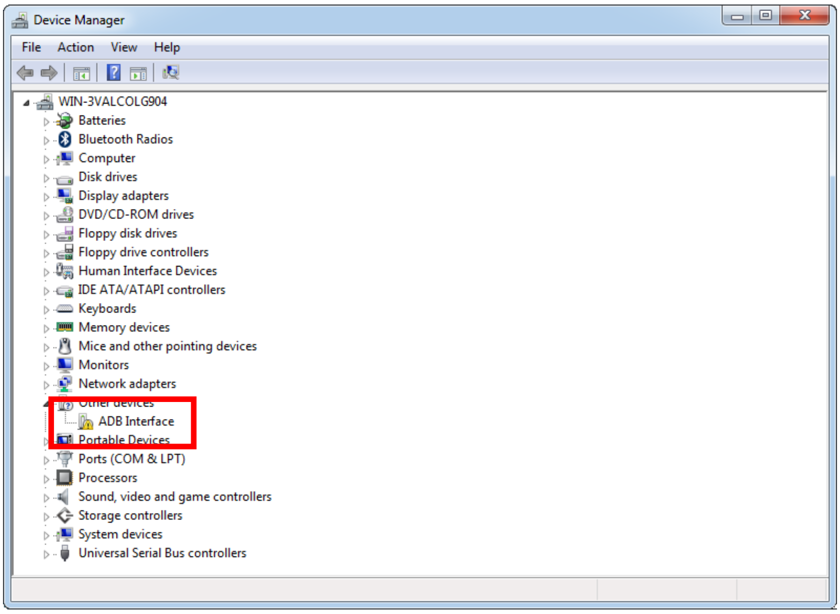
Bug report shortcut
Show a button in the power menu for taking a bug report

Select mock location app

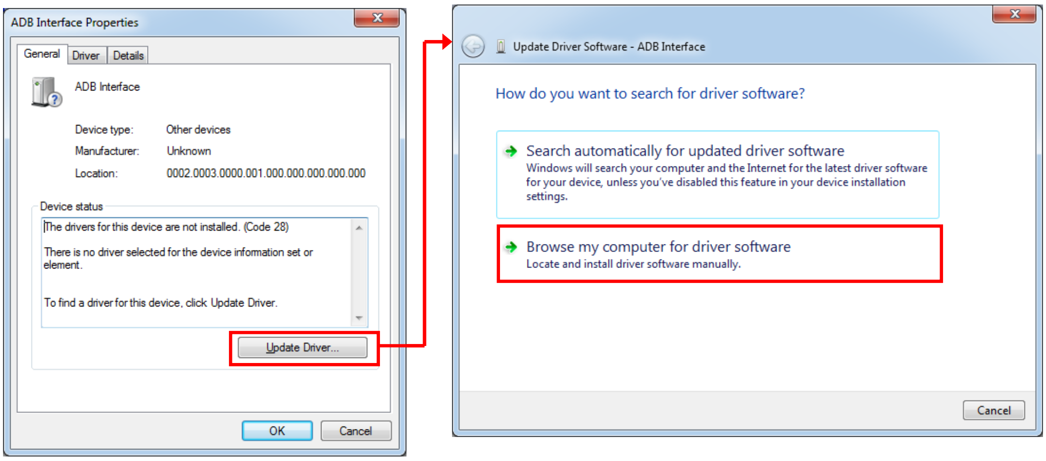
Enable

Install the USB driver for connecting the DT-X400 and PC with the ADB (Android Debug Bridge) protocol.
Skip from this procedure in case of recognizing DT-X400 as ADB on PC already.

Connect the DT-X400 to a PC using a USB cable or a USB cradle.
Then, open the Device Manager by [Control Panel] -> [Hardware and Sound] -> [Device Manager].
In the Device Manager, "ADB Interface" is displayed as an unknown device as shown below.
By right-click the red frame ("Android") in the above figure, displays its properties, and press [Update Driver].



Press "Let me pick from a list of drivers on my computer " at the bottom. In the driver list, with "Show All Devices" is highlighted and press [Next].



Device type: Other devices
Manufacturer: Unknown
Location: 0002.0003.0000.001.000.000.000.000.000

Device status
The drivers for this device are not installed. (Code 28)

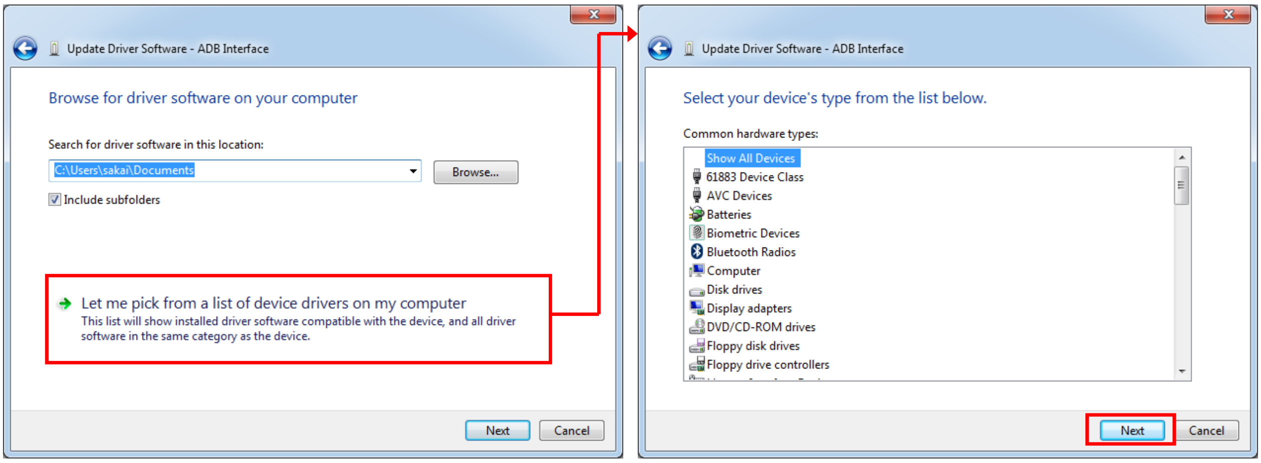
There is no driver selected for the device information set or element.

To find a driver for this device, click Update Driver.

How do you want to search for driver software?

Search automatically for updated driver software
Windows will search your computer and the Internet for the latest driver software for your device, unless you've disabled this feature in your device installation settings.

Browse my computer for driver software
Locate and install driver software manually.



Update Driver Software - ADB Interface

Browse for driver software on your computer

Search for driver software in this location: C:\Users\sakai\Documents

☑ Include subfolders

Let me pick from a list of device drivers on my computer
This list will show installed driver software compatible with the device, and all driver software in the same category as the device.

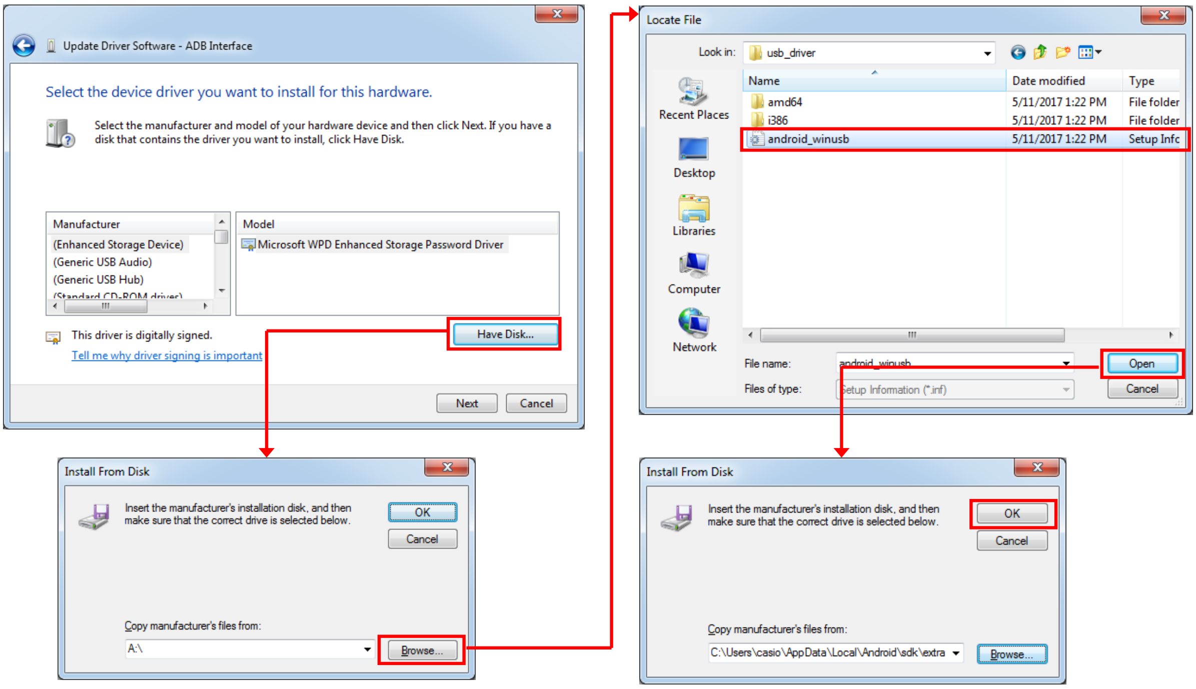
Update Driver Software - ADB Interface

Select your device's type from the list below.

Common hardware types:

- Show All Devices
- 61883 Device Class
- AVC Devices
- Batteries
- Biometric Devices
- Bluetooth Radios
- Computer
- Disk drives
- Display adapters
- DVD/CD-ROM drives
- Floppy disk drives
- Floppy drive controllers

When click [Hard Disk…] in the displayed dialog, a dialog box for specifying the location of the driver opens, so click [Browse…] to open the file dialog box.
In the file dialog box, specify the downloaded USB driver.
(e.g. C:\Users\xxx\AppData\Local\Android\Sdk\extras\google\usb_driver\android_winusb.inf)



Update Driver Software - ADB Interface

Select the device driver you want to install for this hardware.

Select the manufacturer and model of your hardware device and then click Next. If you have a disk that contains the driver you want to install, click Have Disk.

Manufacturer	Model
(Enhanced Storage Device)	Microsoft WPD Enhanced Storage Password Driver
(Generic USB Audio)	
(Generic USB Hub)	
(Standard CD-ROM drives)	

This driver is digitally signed.
Tell me why driver signing is important

Locate File

Look in: usb_driver

Name	Date modified	Type
amd64	5/11/2017 1:22 PM	File folder
i386	5/11/2017 1:22 PM	File folder
android_winusb	5/11/2017 1:22 PM	Setup Infc

File name: android_winusb
Files of type: Setup Information (*.inf)

Install From Disk

Insert the manufacturer's installation disk, and then make sure that the correct drive is selected below.

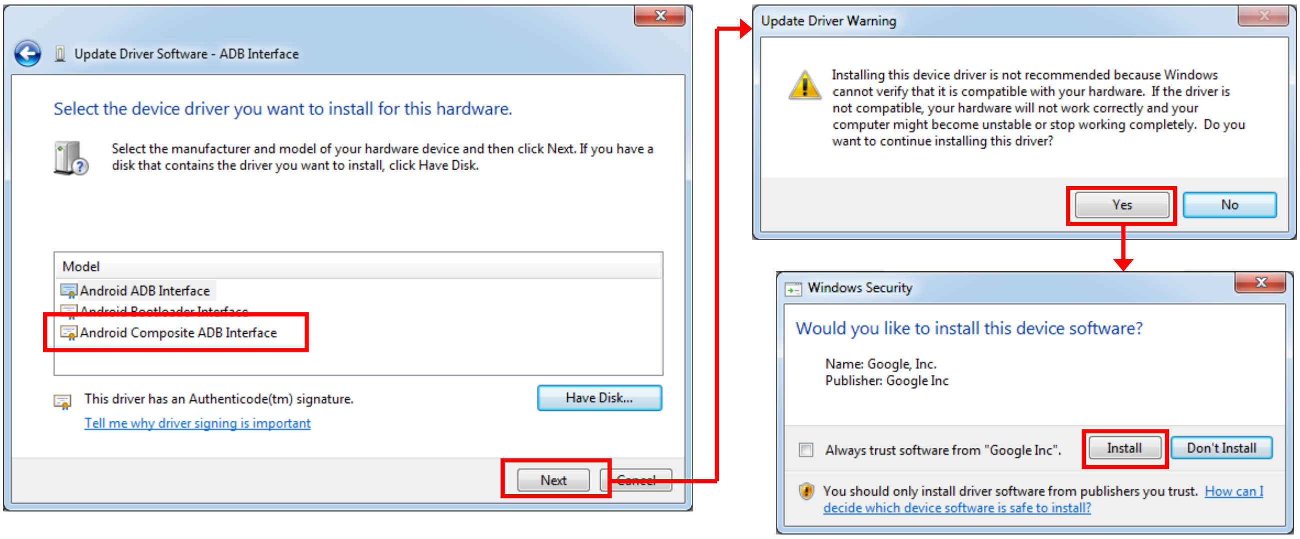
Copy manufacturer's files from: A:\

Install From Disk

Insert the manufacturer's installation disk, and then make sure that the correct drive is selected below.

Copy manufacturer's files from: C:\Users\casio\AppData\Local\Android\sdk\extra

The following driver selection screen will appear. Select "Android Composite ADB Interface" and press [Next].
Answer "Yes" to "Driver Update Warning" and press "Install" of "Windows Security" dialog.



Update Driver Software - ADB Interface

Select the device driver you want to install for this hardware.

Select the manufacturer and model of your hardware device and then click Next. If you have a disk that contains the driver you want to install, click Have Disk.

Model
Android ADB Interface
Android Bootloader Interface
Android Composite ADB Interface

This driver has an Authenticode(tm) signature.

Tell me why driver signing is important

Update Driver Warning

Installing this device driver is not recommended because Windows cannot verify that it is compatible with your hardware. If the driver is not compatible, your hardware will not work correctly and your computer might become unstable or stop working completely. Do you want to continue installing this driver?

Windows Security

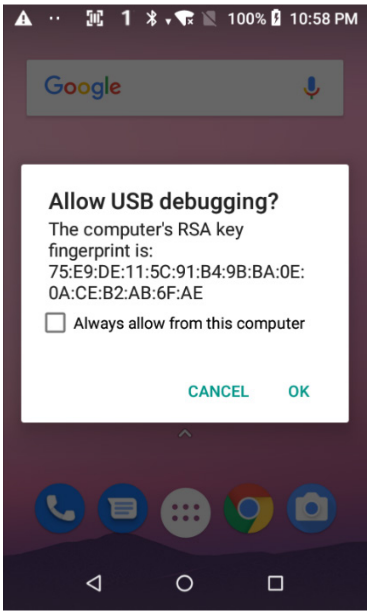
Would you like to install this device software?

Name: Google, Inc.
Publisher: Google Inc

□ Always trust software from "Google Inc".

You should only install driver software from publishers you trust. How can I decide which device software is safe to install?

By the above, the installation of the USB driver for ADB protocol was completed.
Press "OK" when the follow dialog is shown on DT-X400.

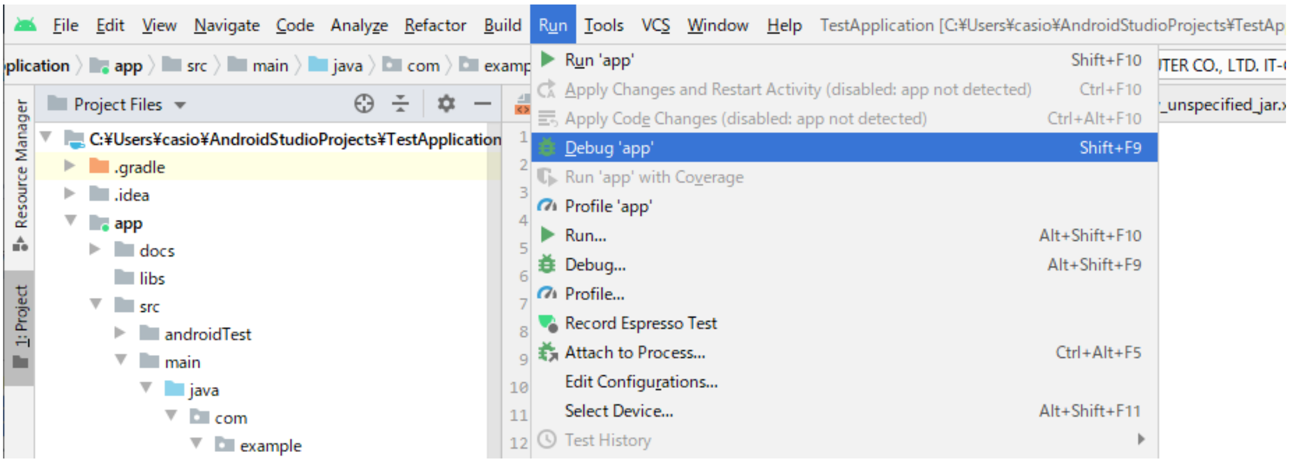


Allow USB debugging?

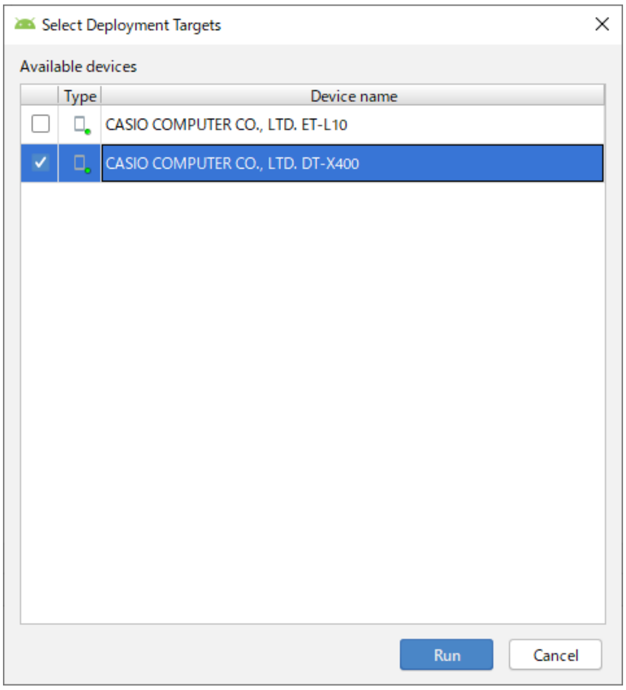
The computer's RSA key fingerprint is:
75:E9:DE:11:5C:91:B4:9B:BA:0E:0A:CE:B2:AB:6F:AE

□ Always allow from this computer

Select "Run" -> "Debug 'app'" on Android Studio.



When the follow dialog is shown, select "CASIO COMPUTER CO., LTD. DT-X400 (Android 8.1.0, API 27)" from "Available devices" and press "Run". Start to debug an application on DT-X400.



Available devices

	Type	Device name
☐		CASIO COMPUTER CO., LTD. ET-L10
☑		CASIO COMPUTER CO., LTD. DT-X400

3.3.2 CASIO extended function

The following method is available for developing applications that control keys and barcode scanners.
For details on each function, refer to the "Device Control Library Manual".

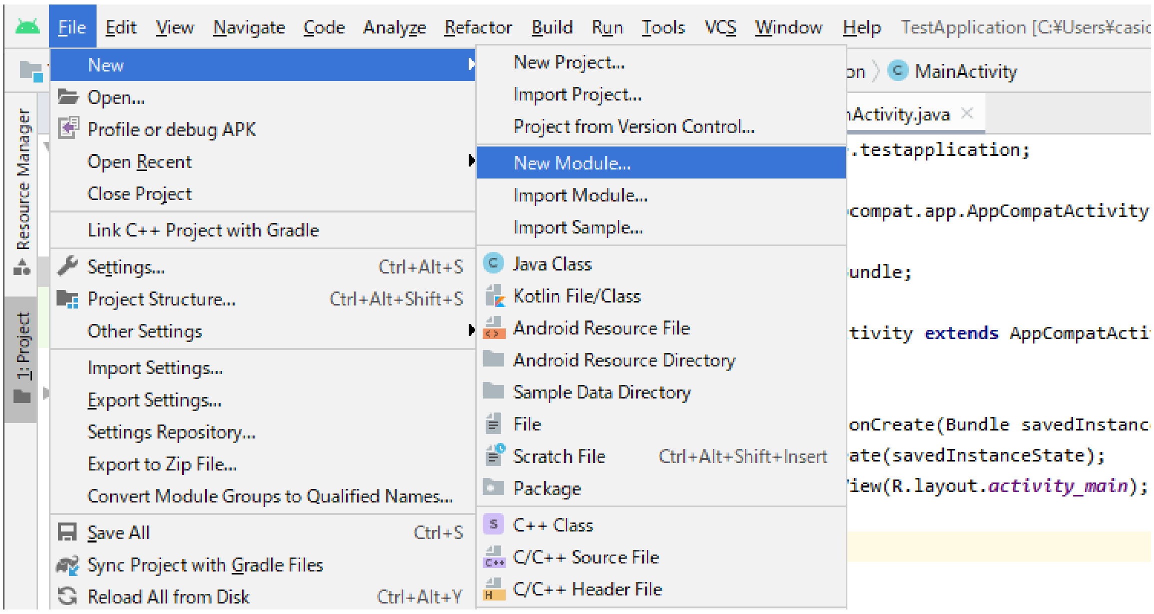
- Application development with imported device control library

Device Library

When developing an application that controls CASIO extended functions (barcode scanner, trigger key, etc.), import the DeviceLibrary. The library import is required for each project.

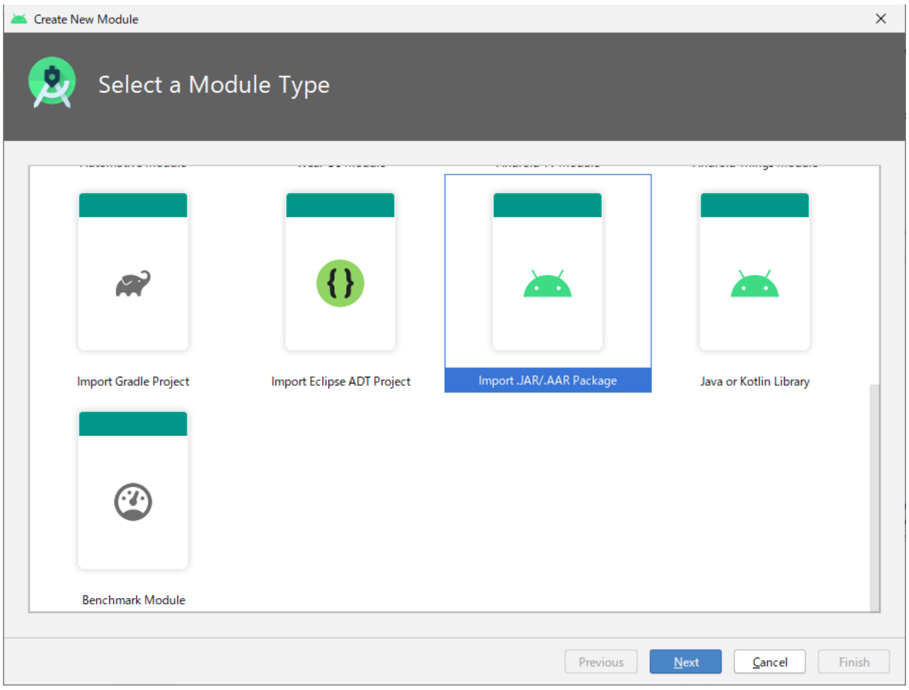
Import

Open the project from the Android Studio, select "New Module" dialog with [File] -> [New] -> [New Module].

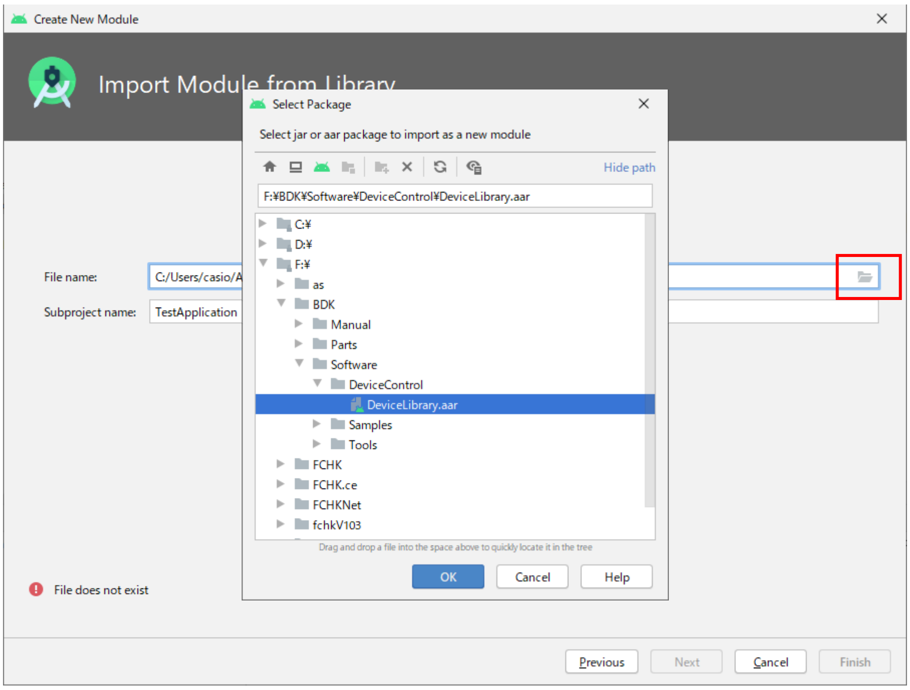


```
.testapplication;

compat.app.AppCompatActivity

undle;

tivity extends AppCompatActi

onCreate(Bundle savedInstanc
ate(savedInstanceState);
iew(R.layout.activity_main);
```

Select "Import .JAR/.AAR Package" and select "Next".



Press the 🗁 of "Filename" and specify the DeviceLibrary (AAR) to be imported.



Select jar or aar package to import as a new module

F:¥BDK¥Software¥DeviceControl¥DeviceLibrary.aar

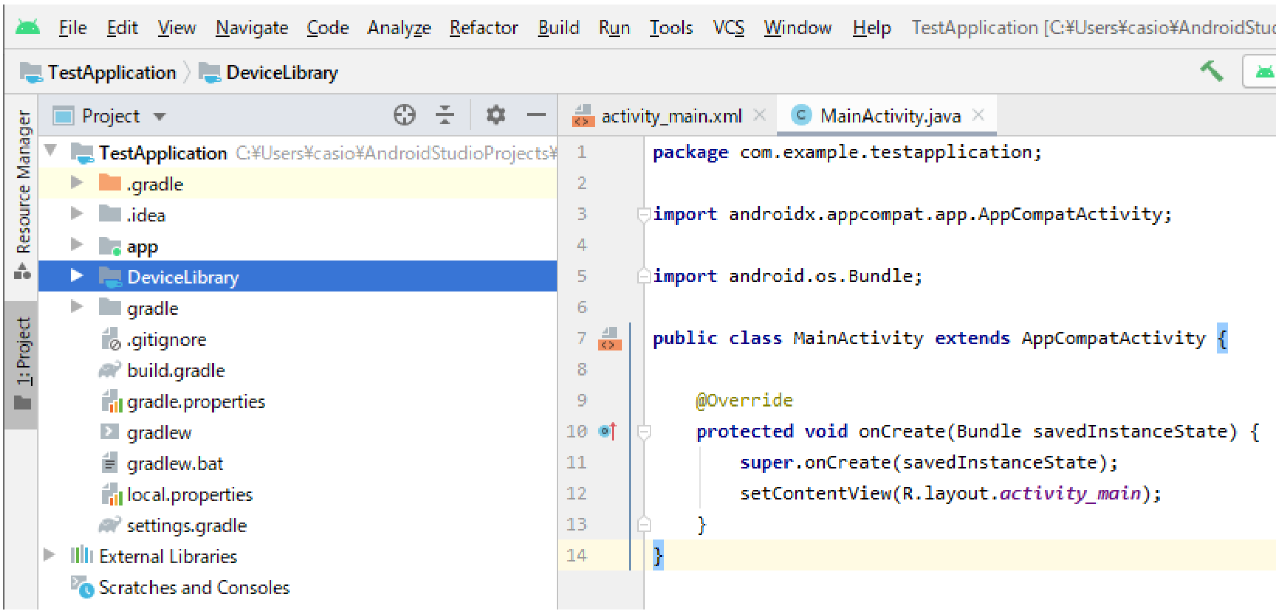
File name: C:/Users/casio/A

Subproject name: TestApplication

Drag and drop a file into the space above to quickly locate it in the tree

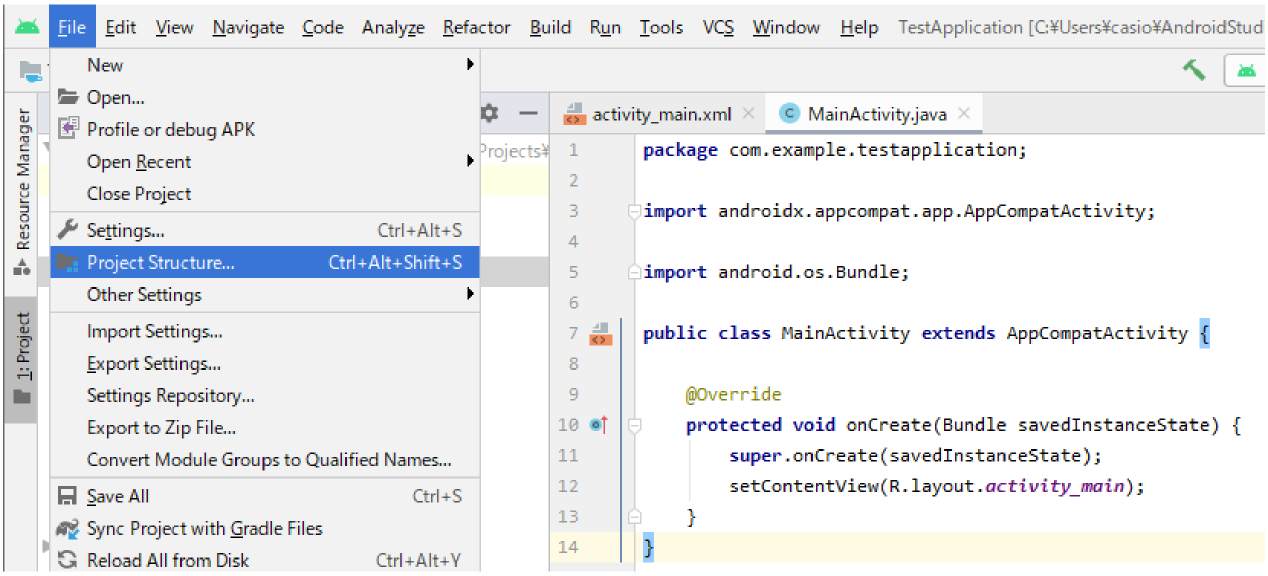
File does not exist

Confirm that the DeviceLibrary is added to the Android Studio Project tree.



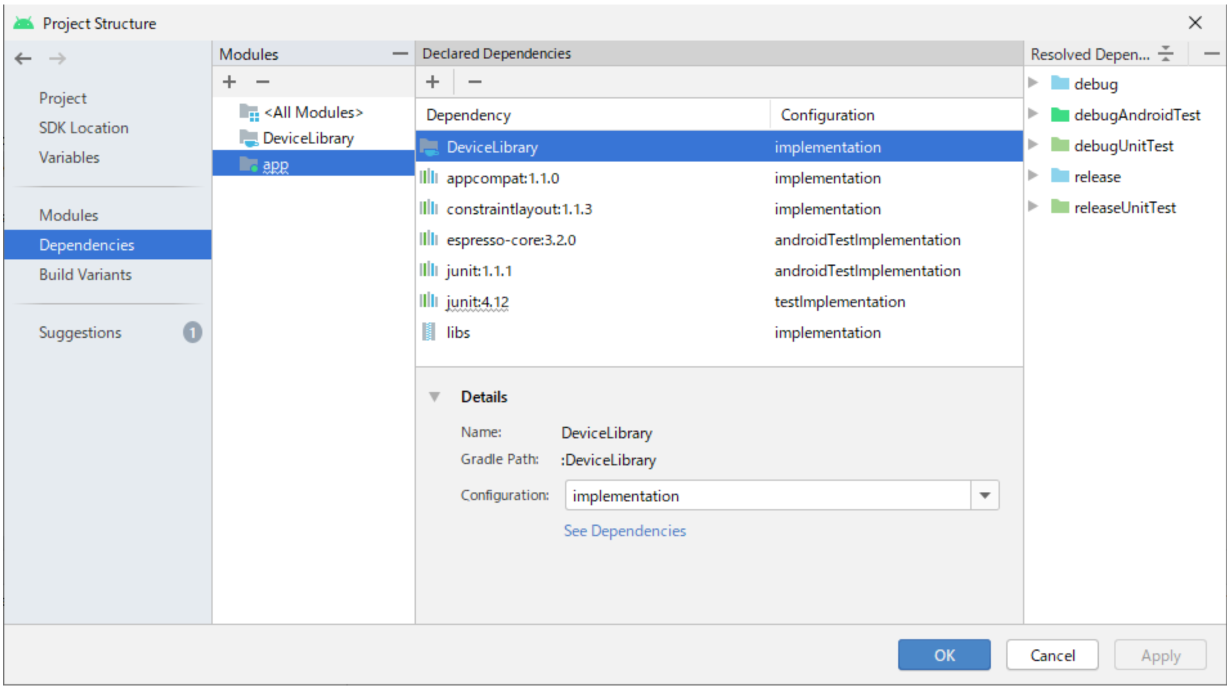
```
package com.example.testapplication;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

Select [File] -> [Project Structure ...].


File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help TestApplication [C:¥Users¥casio¥AndroidStud
New
Open...
Profile or debug APK
Open Recent
Close Project
Settings... Ctrl+Alt+S
Project Structure... Ctrl+Alt+Shift+S
Other Settings
Import Settings...
Export Settings...
Settings Repository...
Export to Zip File...
Convert Module Groups to Qualified Names...
Save All Ctrl+S
Sync Project with Gradle Files
Reload All from Disk Ctrl+Alt+Y
Resource Manager
1: Project
activity_main.xml
MainActivity.java
Projects¥

```
package com.example.testapplication;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

Open the [Dependencies] tab, and check if "DeviceLibrary" exists. If it exists, subsequent operations are unnecessary.



Dependency	Configuration
DeviceLibrary	implementation
appcompat:1.1.0	implementation
constraintlayout:1.1.3	implementation
espresso-core:3.2.0	androidTestImplementation
junit:1.1.1	androidTestImplementation
junit:4.12	testImplementation
libs	implementation

Details

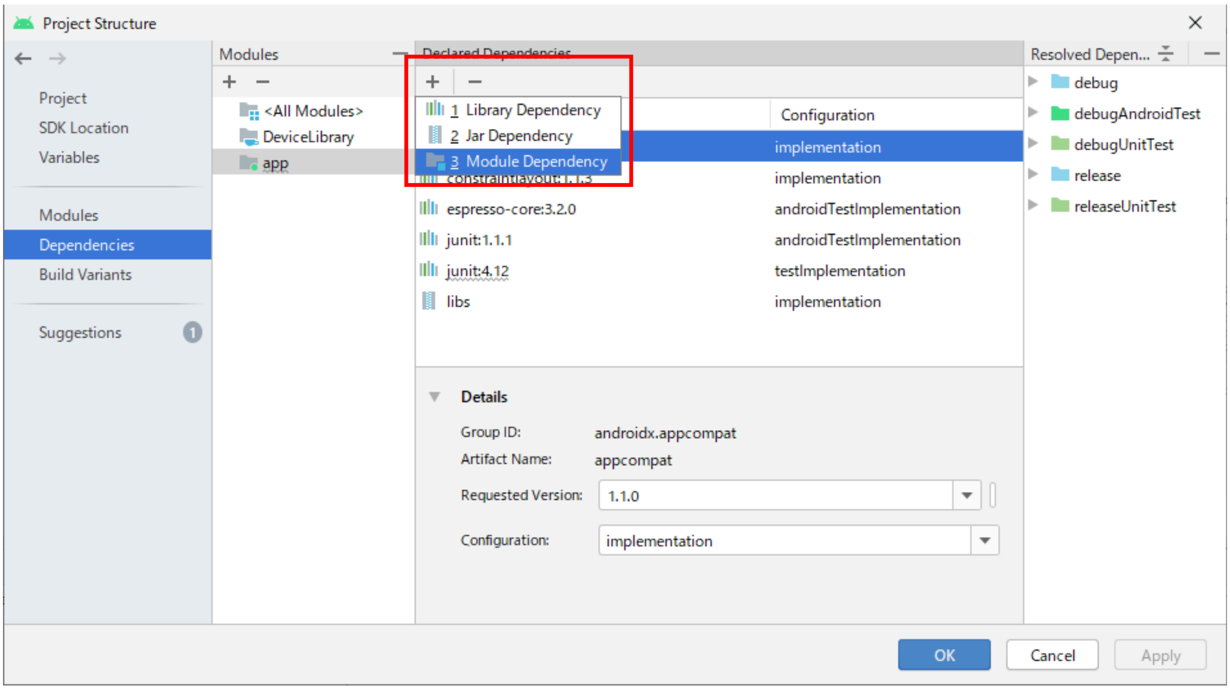
Name: DeviceLibrary

Gradle Path: :DeviceLibrary

Configuration: implementation

See Dependencies

Select "+" button and select [Module Dependency].



	Configuration
	implementation
constraintlayout:1.1.3	implementation
espresso-core:3.2.0	androidTestImplementation
junit:1.1.1	androidTestImplementation
junit:4.12	testImplementation
libs	implementation

Details

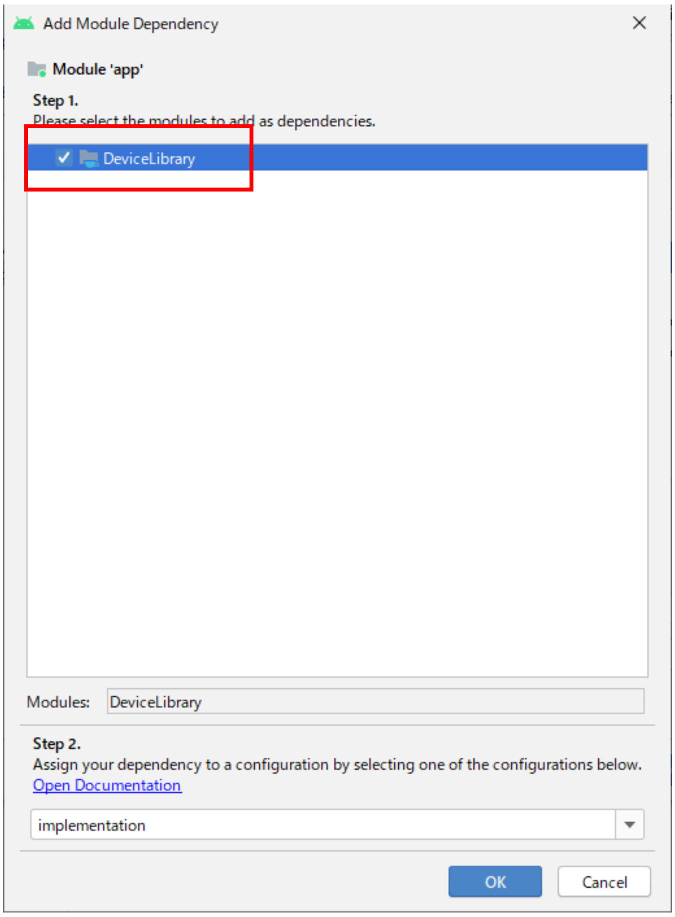
Group ID: androidx.appcompat

Artifact Name: appcompat

Requested Version: 1.1.0

Configuration: implementation

Check the "DeviceLibrary" and select "OK".



Module 'app'

Step 1.
Please select the modules to add as dependencies.

☑ DeviceLibrary

Modules: DeviceLibrary

Step 2.
Assign your dependency to a configuration by selecting one of the configurations below.
Open Documentation

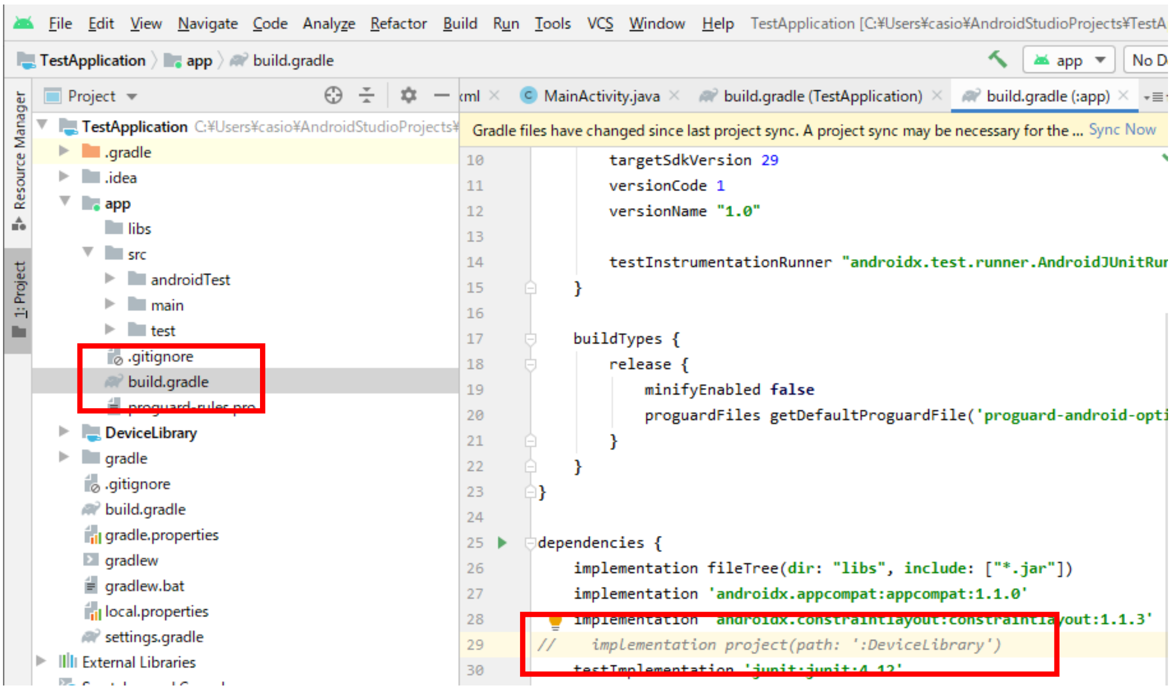
implementation

This completes the import of the DeviceLibrary.

Update

To update the DeviceLibrary imported in the project of Android Studio (exchange to the new version), need to unregister them and import new one.

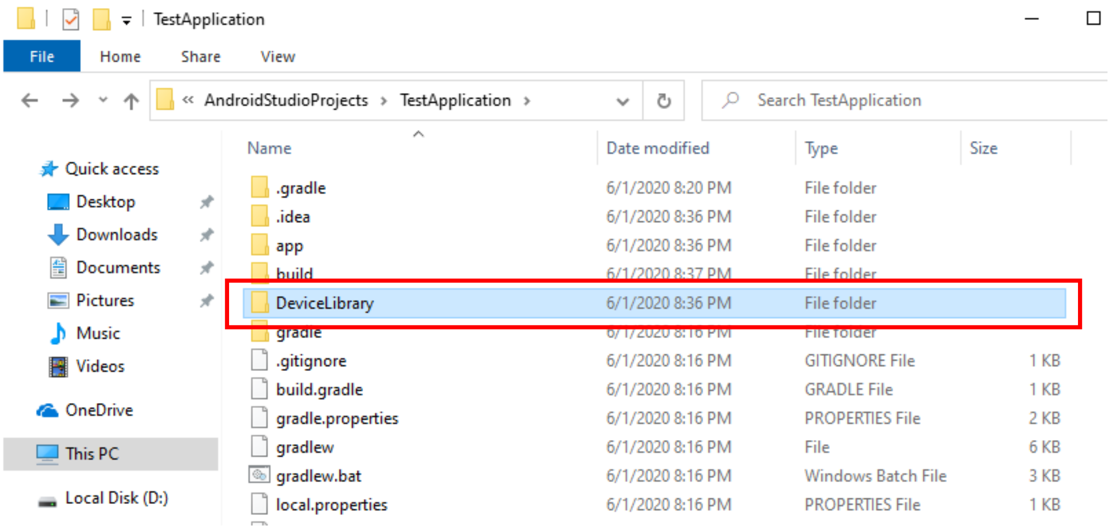
To unregister the DeviceLibrary, remove the implementation of the DeviceLibrary in build.gradle.



Gradle files have changed since last project sync. A project sync may be necessary for the ... Sync Now

```
            targetSdkVersion 29
            versionCode 1
            versionName "1.0"

            testInstrumentationRunner "androidx.test.runner.AndroidJUnitRur
        }

        buildTypes {
            release {
                minifyEnabled false
                proguardFiles getDefaultProguardFile('proguard-android-opti
            }
        }
    }

    dependencies {
        implementation fileTree(dir: "libs", include: ["*.jar"])
        implementation 'androidx.appcompat:appcompat:1.1.0'
        implementation 'androidx.constraintlayout:constraintlayout:1.1.3'
    //    implementation project(path: ':DeviceLibrary')
        testImplementation 'junit:junit:4.12'
```

Delete the DeviceLibrary folder in the root folder of the project.



Name	Date modified	Type	Size
.gradle	6/1/2020 8:20 PM	File folder	
.idea	6/1/2020 8:36 PM	File folder	
app	6/1/2020 8:36 PM	File folder	
build	6/1/2020 8:37 PM	File folder	
DeviceLibrary	6/1/2020 8:36 PM	File folder	
gradle	6/1/2020 8:16 PM	File folder	
.gitignore	6/1/2020 8:16 PM	GITIGNORE File	1 KB
build.gradle	6/1/2020 8:16 PM	GRADLE File	1 KB
gradle.properties	6/1/2020 8:16 PM	PROPERTIES File	2 KB
gradlew	6/1/2020 8:16 PM	File	6 KB
gradlew.bat	6/1/2020 8:16 PM	Windows Batch File	3 KB
local.properties	6/1/2020 8:16 PM	PROPERTIES File	1 KB

Import new DeviceLibrary as in the described before section.
Finally, undo the comment out of build.gradle and select "Sync".
The update of the DeviceLibrary is complete.

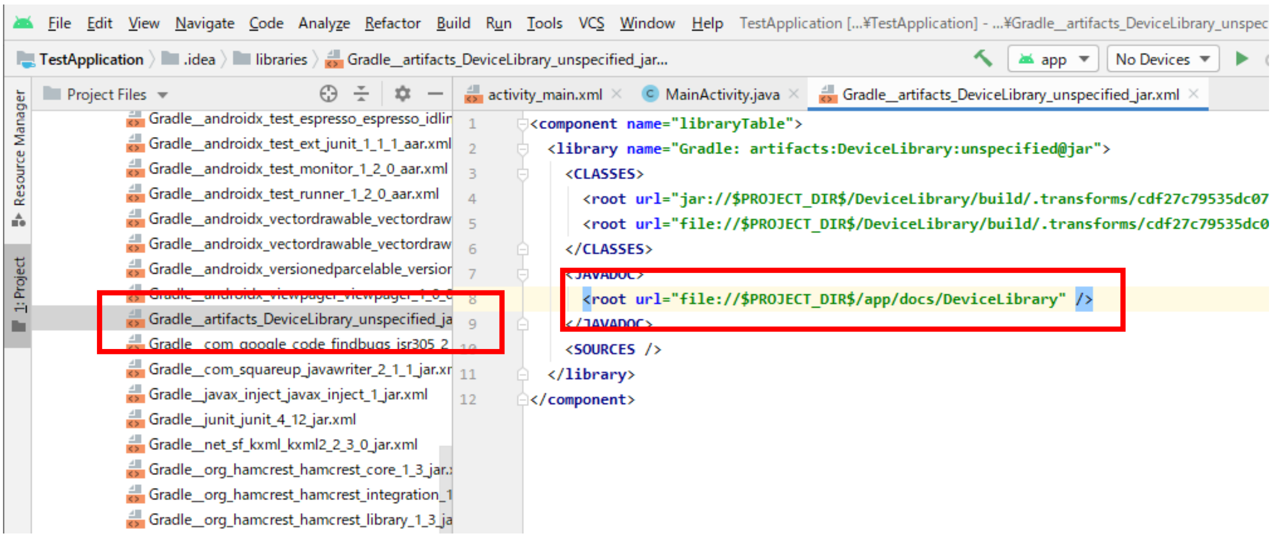
Help files

This section explains how to enable pop-up help (pops up help message when you move the cursor to the function name) of DeviceLibrary.

Copy the javadoc in the Basic Development Kit to any folder. Here, it is assumed that it is copied to the "app/docs/DeviceLibrary" folder.

With the import of the DeviceLibrary completed, open "Gradle__artifacts_DeviceLibrary.xml".
For "<JAVADOC>" in this xml file, specify the location of the javadoc.
If a xml file is not created, restart Android Studio.

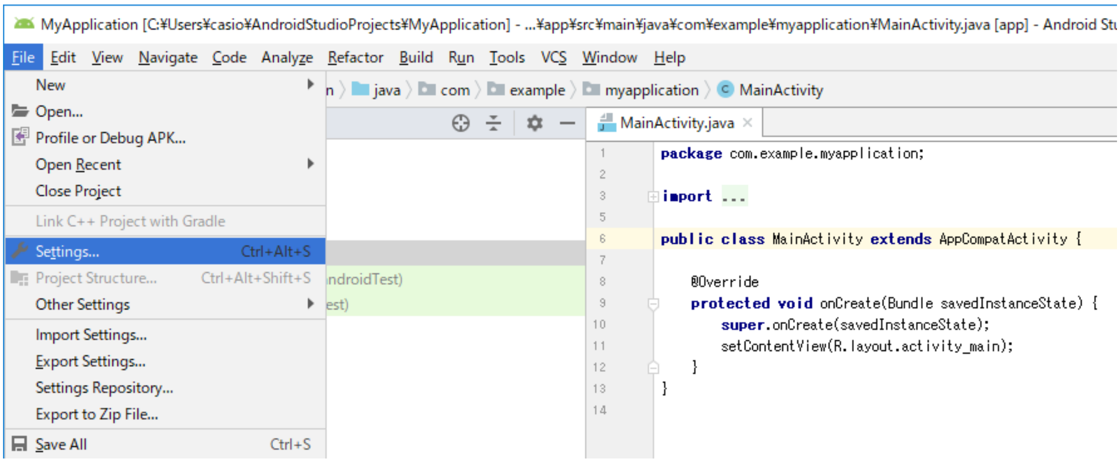
```
<JAVADOC>
<root url="file://$PROJECT_DIR$/app/docs/DeviceLibrary" />
</JAVADOC>
```



```
<component name="libraryTable">
  <library name="Gradle: artifacts:DeviceLibrary:unspecified@jar">
    <CLASSES>
      <root url="jar://$PROJECT_DIR$/DeviceLibrary/build/.transforms/cdf27c79535dc07
      <root url="file://$PROJECT_DIR$/DeviceLibrary/build/.transforms/cdf27c79535dc0
    </CLASSES>
    <JAVADOC>
      <root url="file://$PROJECT_DIR$/app/docs/DeviceLibrary" />
    </JAVADOC>
    <SOURCES />
  </library>
</component>
```

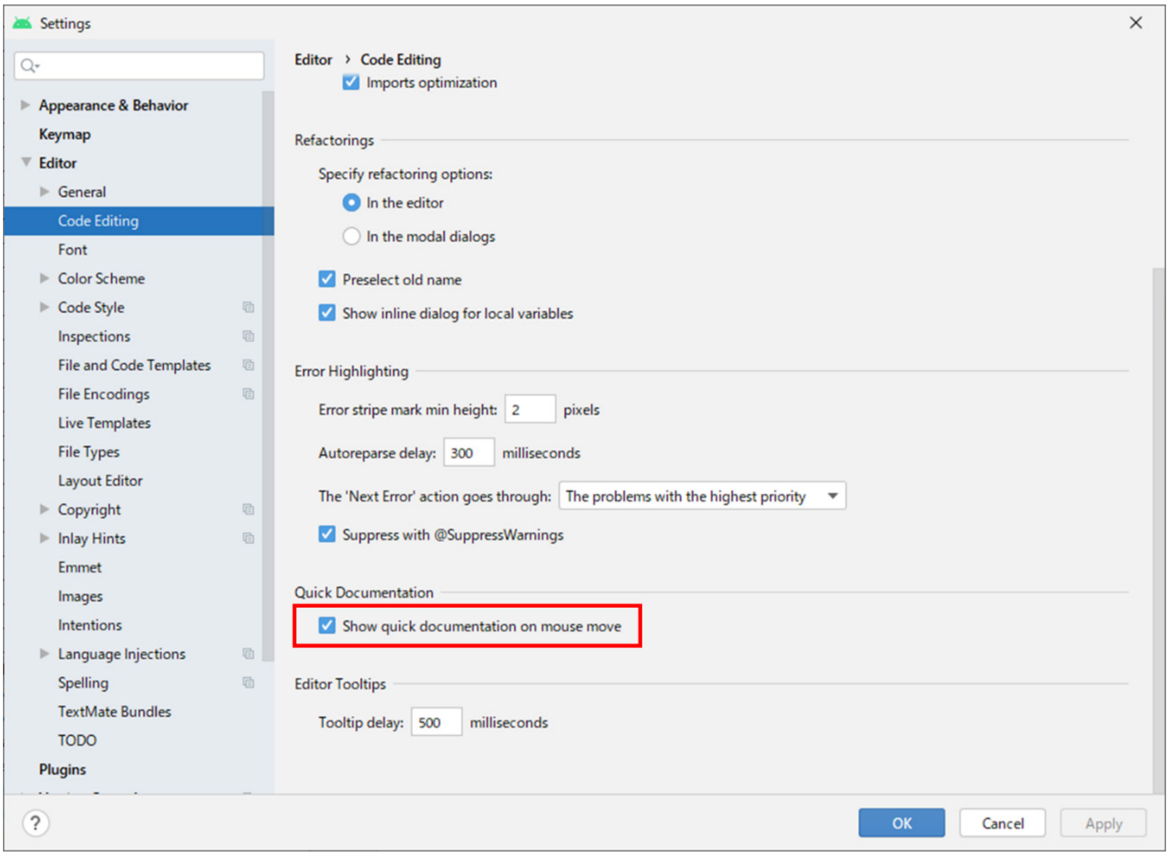
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
TestApplication [...¥TestApplication] - ...¥Gradle__artifacts_DeviceLibrary_unspec
TestApplication
.idea
libraries
Gradle__artifacts_DeviceLibrary_unspecified_jar...
app
No Devices
Project Files
activity_main.xml
MainActivity.java
Gradle__artifacts_DeviceLibrary_unspecified_jar.xml
Resource Manager
1: Project
Gradle__androidx_test_espresso_espresso_idlir
Gradle__androidx_test_ext_junit_1_1_1_aar.xml
Gradle__androidx_test_monitor_1_2_0_aar.xml
Gradle__androidx_test_runner_1_2_0_aar.xml
Gradle__androidx_vectordrawable_vectordraw
Gradle__androidx_vectordrawable_vectordraw
Gradle__androidx_versionedparcelable_versior
Gradle__androidx_viewpager_viewpager_1_0_
Gradle__artifacts_DeviceLibrary_unspecified_ja
Gradle__com_google_code_findbugs_jsr305_2
Gradle__com_squareup_javawriter_2_1_1_jar.xr
Gradle__javax_inject_javax_inject_1_jar.xml
Gradle__junit_junit_4_12_jar.xml
Gradle__net_sf_kxml_kxml2_2_3_0_jar.xml
Gradle__org_hamcrest_hamcrest_core_1_3_jar.:
Gradle__org_hamcrest_hamcrest_integration_1
Gradle__org_hamcrest_hamcrest_library_1_3_ja

Next, select [File] -> [Settings ...] from the Android Studio.


MyApplication [C:¥Users¥casio¥AndroidStudioProjects¥MyApplication] - ...¥app¥src¥main¥java¥com¥example¥myapplication¥MainActivity.java [app] - Android Stu
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
New
Open...
Profile or Debug APK...
Open Recent
Close Project
Link C++ Project with Gradle
Settings... Ctrl+Alt+S
Project Structure... Ctrl+Alt+Shift+S
Other Settings
Import Settings...
Export Settings...
Settings Repository...
Export to Zip File...
Save All Ctrl+S
n java com example myapplication MainActivity
MainActivity.java
ndroidTest)
est)

```
package com.example.myapplication;

import ...

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

Check "Show quick documentation on mouse move" in [Editor] -> [Code Editing].



Editor › Code Editing

☑ Imports optimization

Refactorings

Specify refactoring options:

● In the editor

○ In the modal dialogs

☑ Preselect old name

☑ Show inline dialog for local variables

Error Highlighting

Error stripe mark min height: 2 pixels

Autoreparse delay: 300 milliseconds

The 'Next Error' action goes through: The problems with the highest priority

☑ Suppress with @SuppressWarnings

Quick Documentation

☑ Show quick documentation on mouse move

Editor Tooltips

Tooltip delay: 500 milliseconds

The registration of help file is complete. Mouse over the function name in Android Studio to display help for the function.