

# *FIRST*<sup>®</sup> Tech Challenge

# PushBot Build Guide

## Part III: Programming the Robot



Revision History		
Revision	Date	Description
1	8/15/2014	Initial Release – by FTC Team #003 Australia, The Southport School
2	9/1/2014	Replaced MATRIX with TETRIX content by former FTC Team #2843, Under the Son
3	8/6/2015	Updated using the new kit of parts and new programming environment.

## Contents

---

Contents.....	2
Programming the Robot.....	3
Opening Internet Explorer .....	4
Opening File Explorer (aka Windows Explorer) .....	6
Determining whether Android Studio can be installed .....	7
Installing the FTC SDK.....	14
Installing the Java Development Kit (JDK) .....	21
Installing the Android Studio Integrated Development Environment (IDE) .....	33
Registering the PushBot Op modes .....	57
Preparing the Robot Controller and Driver Station cell phones.....	60
Naming the Cell Phone .....	72
Enabling Debugging Mode on the Robot Controller Cell Phone .....	76
Installing the USB Device Driver .....	79
Deploying the Robot Controller .....	89
Deploying the Driver Station.....	91

## Programming the Robot

---

Android Studio is one of the Integrated Development Environments approved for FTC Team use by FIRST. It is a text-based IDE that can be used to build, deploy, and debug Java applications. The IDE is available without cost for Windows, Mac OS X, and Linux and can be downloaded from the <https://developer.android.com> website. Select “Develop”, then “Tools” and then the next selection will depend onto which operating system Android Studio will be installed. This document will show all of the steps to download the Windows 64-bit version of Android Studio and install it onto a Windows 8 computer.

Android Studio is built around Java, so the Java Development Kit (JDK) must be installed before Android Studio is installed. Version 7 has been tested with the FTC SDK, so this document will show all of the steps to download and install it onto a Windows 8 computer. Note that Java version 8 is available, but hasn't been tested with the FTC SDK.

The FTC SDK contains the Robot controller module which is the application that runs on the cell phone located on the Robot. That cell phone controls the electronics on the Robot and communicates with another cell phone that acts as the driver station. The driver station allows the Team to select Op modes, run the selected Op modes and send gamepad inputs to the Robot controller.

The PushBot software package contains Op modes designed specifically for the PushBot. A simple manual (i.e. TeleOp) Op modes is included in the package that allows a Team to control the Robot's drive wheels using one gamepad's joysticks, control the Robot's arm using a second gamepad's left joystick, and control the Robot's hands (i.e. servos) using two of the buttons on the second gamepad.

Installation will begin with the FTC SDK, then the PushBot package. It will continue with the installation of Android Studio and finish with deploying the Robot controller application to the cell phone.

Note: The sections in this portion build on actions made in the previous sections. It is recommended that sections be completed in the order shown.

This document shows some steps that use the Windows Start Screen. This screen will be shown when the Windows key is pressed (shown below). Once on the start screen, typing the name of the application will cause a list to be shown of known applications. The arrow keys can be used to select an item in the list. Once a list item is selected, pressing enter will start the application.

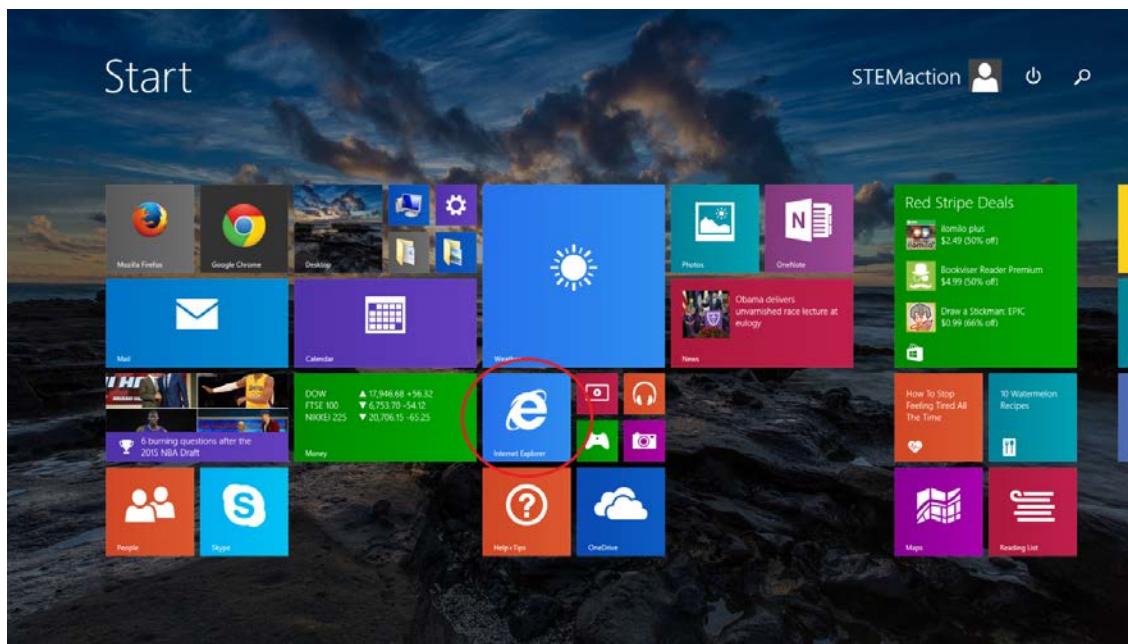


Note: Before beginning, charge the phones.

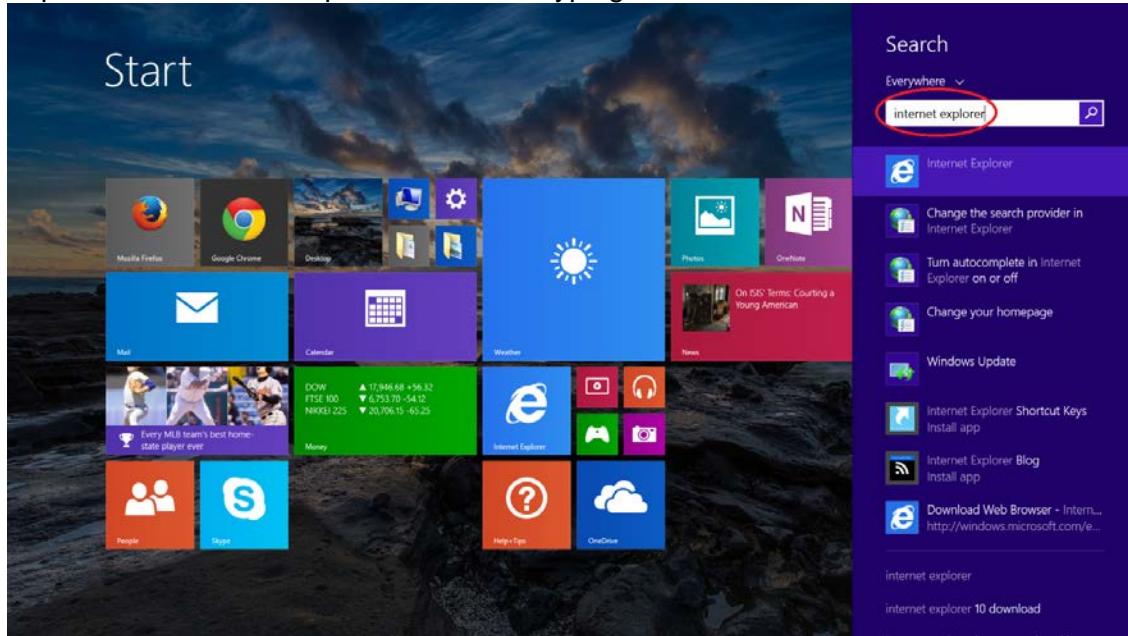
## Opening Internet Explorer

Some of the steps in this section use Internet Explorer (IE). This section shows how to start an application using the Windows logo key. A similar procedure can be used to open other applications (as shown in the next section – Opening File Explorer). The instructions for opening IE are shown once here and referenced in the sections that follow. Two options are shown below.

Option A: Press the Windows key and select the Internet Explorer icon from the Windows start screen (shown in the red circle).



Option B: Press the Windows key and type the name of the application – in this case the name is ‘Internet Explorer’. Remember to press enter after typing the name.





The Internet Explorer application is shown below.

New tab



### Opening File Explorer (aka Windows Explorer)

Some of the steps in this section use File Explorer (FE). The instructions for opening it are shown once here and referenced in the sections that follow. Two options are shown below.

Option A: Press the Windows key and select the File Explorer icon from the Windows start screen (shown in the red circle).



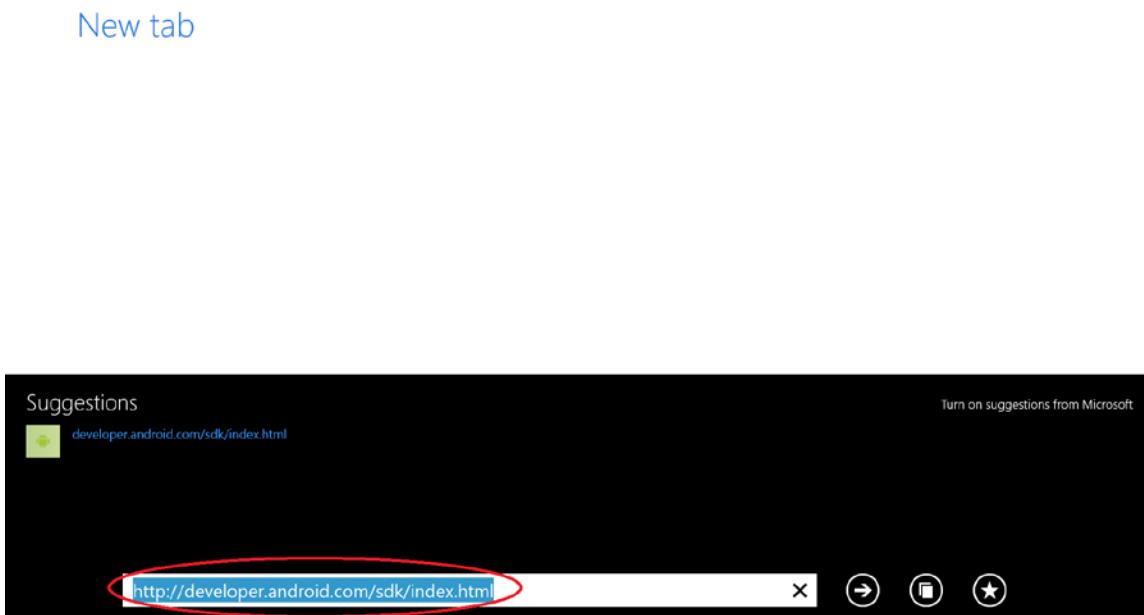
Option B: Press the Windows key and type the name of the application – in this case the name is 'file explorer'. Remember to press enter after typing the name.



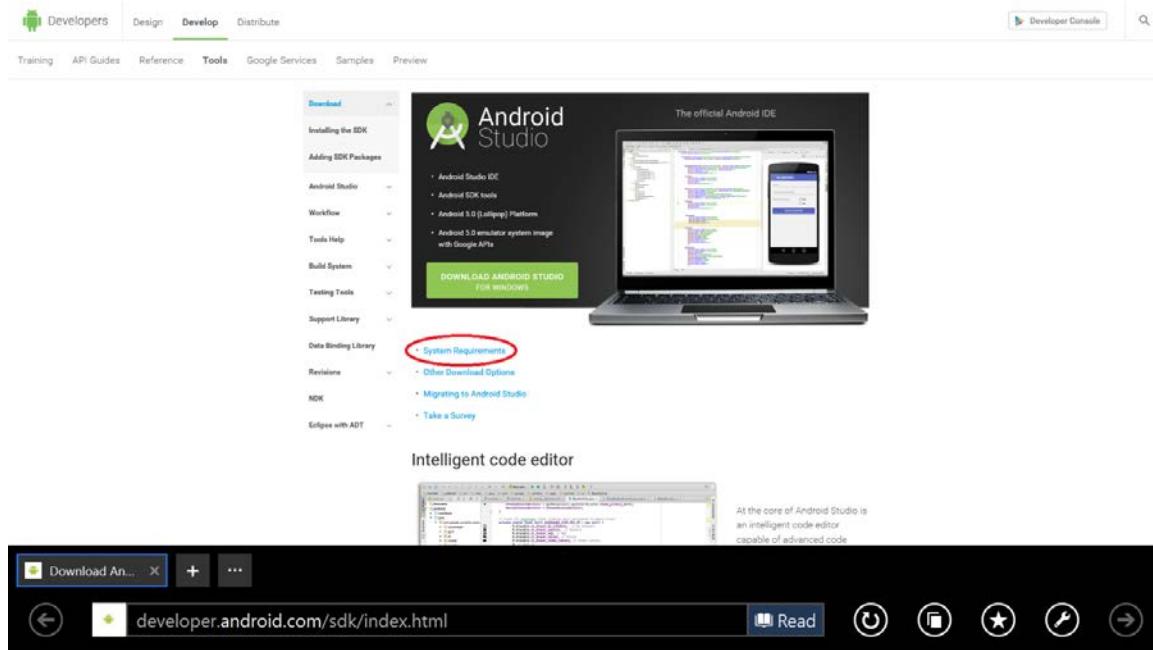
### **Determining whether Android Studio can be installed**

Android Studio publishes the qualities a computer must have to be able to run the IDE. Before installing the Android Studio IDE, Java, or the FTC SDK, check whether the machine can run the IDE. If the machine can't support the Android Studio IDE's minimum requirements, then there is no need to install Java or the FTC SDK, so perform this check before installing anything. If the test fails, then a different machine will be needed.

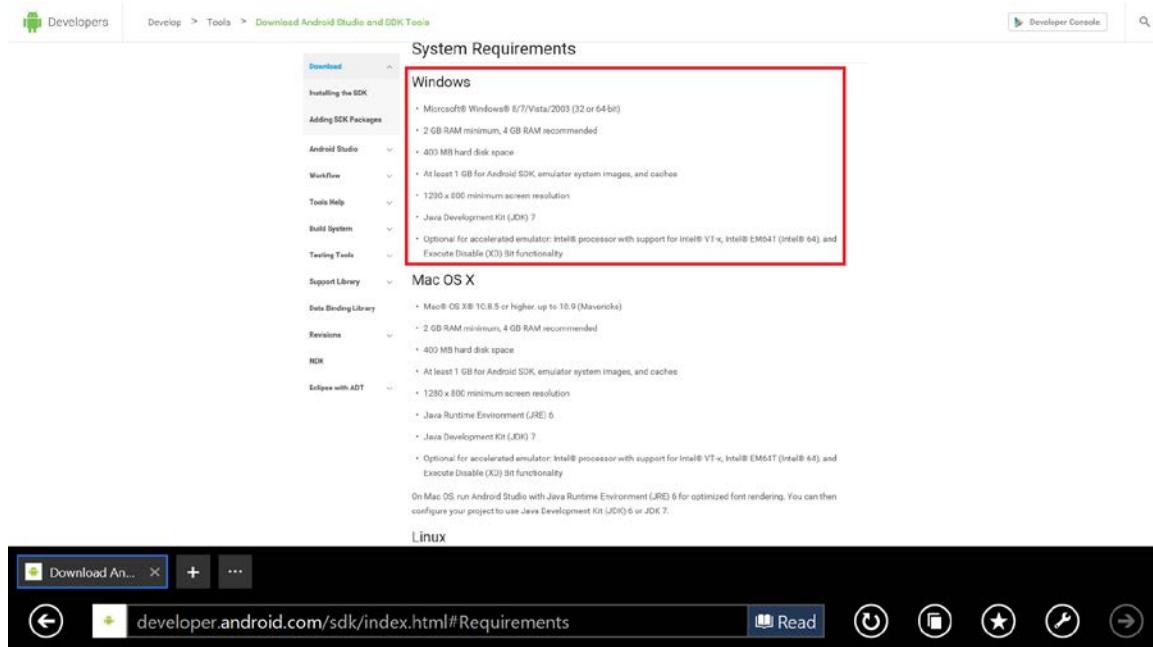
Step 1: Open Internet Explorer and enter '<http://developer.android.com/sdk/index.html>' into the Address field.



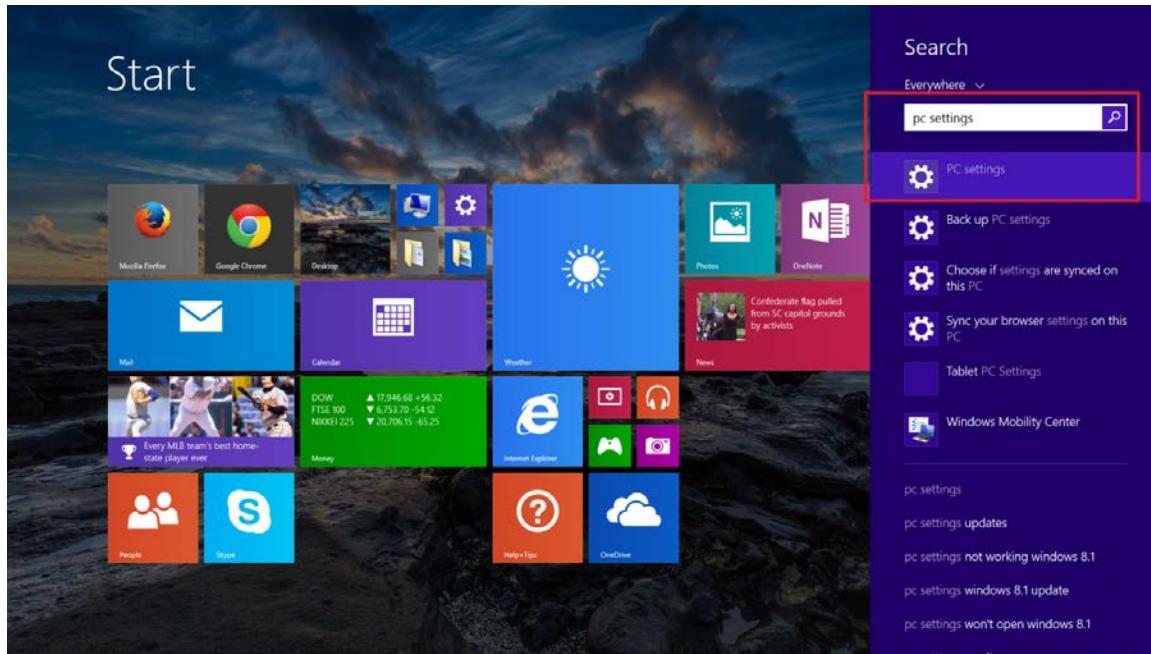
## Step 2: Select ‘System Requirements’.



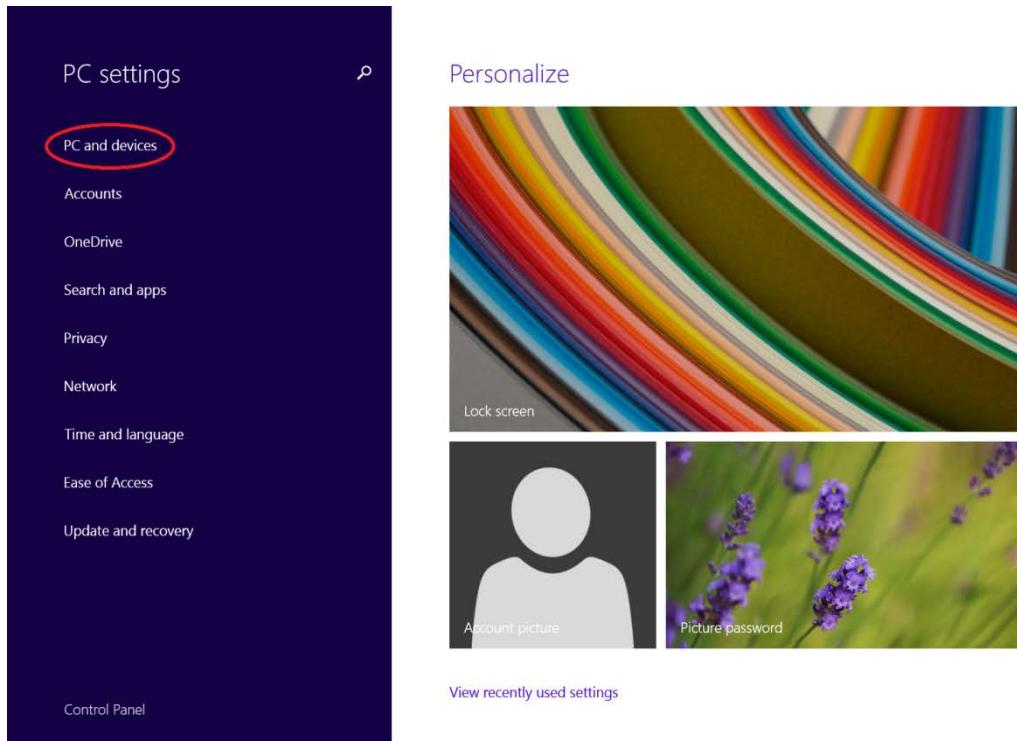
Step 3: Examine the system requirements that match the installation operating system’s type. The system requirements are farther down the page shown in the previous step; clicking the button will scroll the page downward to the System Requirements portion. The requirements for the Windows operating system are shown in the red box.



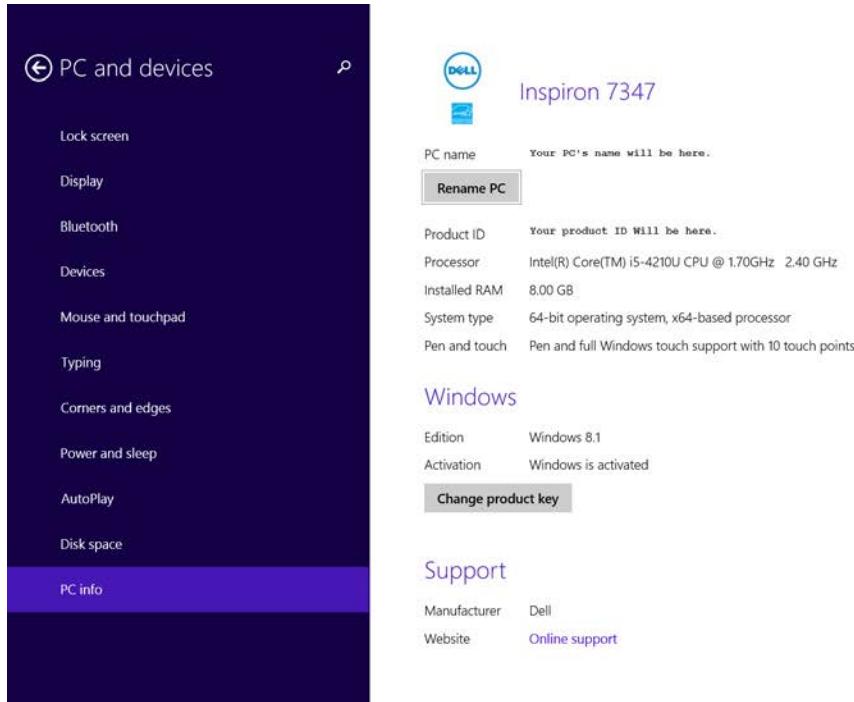
Step 4: Press the Windows key and type 'PC settings'. Remember to press enter after typing the name.



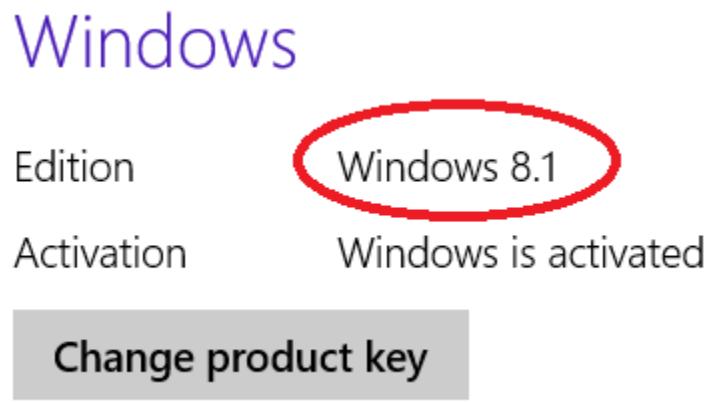
Step 5: Select PC and devices.



Step 6: Match the information from the following window with the requirements shown in step 3 – the process is shown below the following image. Note the System Type; it will be used later when installing the Java Development Kit. For this document the system type is 64-bit.



Requirement 1: Microsoft Windows 8/7/Vista/2003 (32 or 64-bit). Since Windows 8.1 is higher than Windows 8, the requirement has been met.

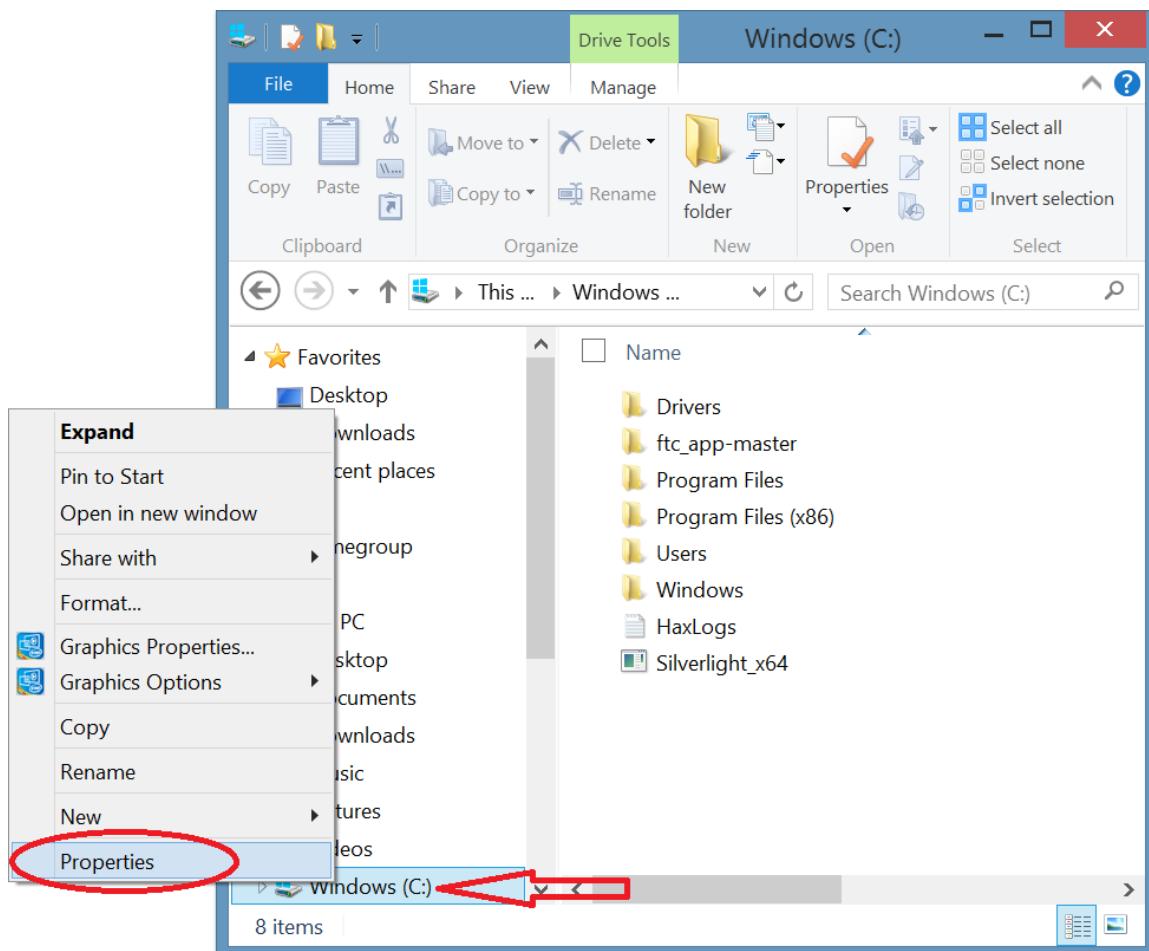


Requirement 2: 2GB RAM minimum, 4GB RAM recommended. Since 8GB is greater than 4GB, the requirement has been met.

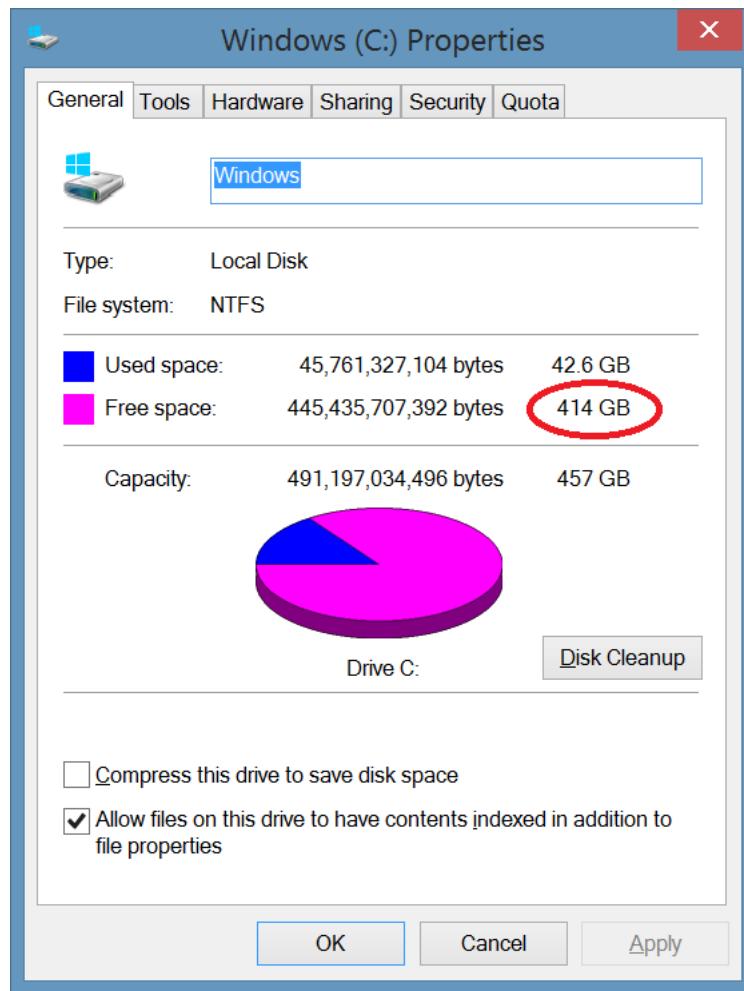
Processor	Intel(R) Core(TM) i5-4210U CPU @ 1.70GHz 2.40 GHz
Installed RAM	8.00 GB
System type	64-bit operating system, x64-based processor
Pen and touch	Pen and full Windows touch support with 10 touch points

## Windows

Step 7: To compare the hard disk space another Windows tool will be needed. Open File Explorer (known as Windows Explorer on older systems). Right click “Windows (C:)” and select Properties. If the computer has a touch screen, hold a stylus or finger on “Windows (C:)” until the context menu is shown.

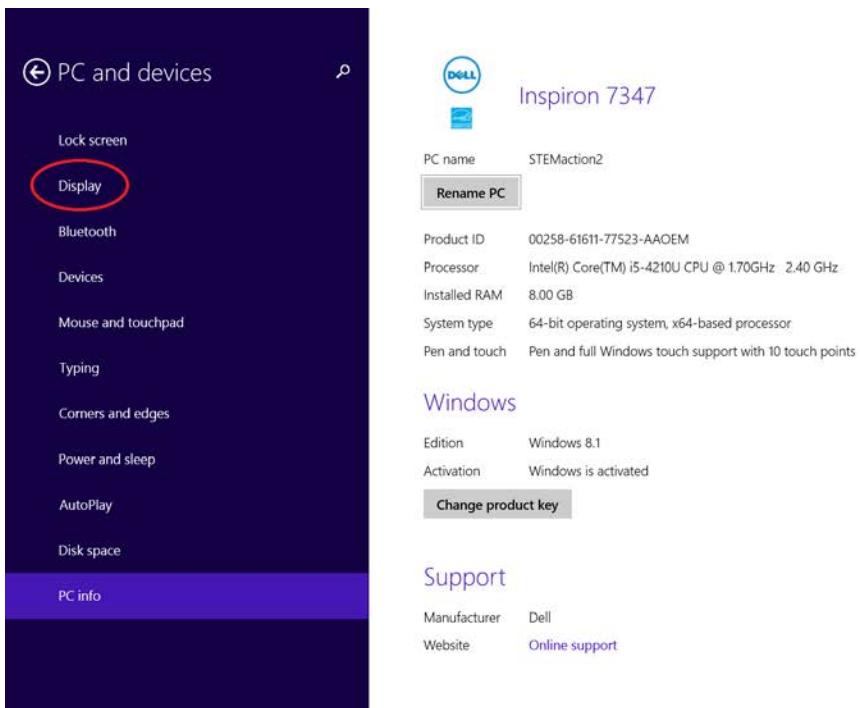


Requirement 3: 400MB hard disk space. Since 414 GB is greater than 400MB (1GB=1000MB), the requirement has been met.

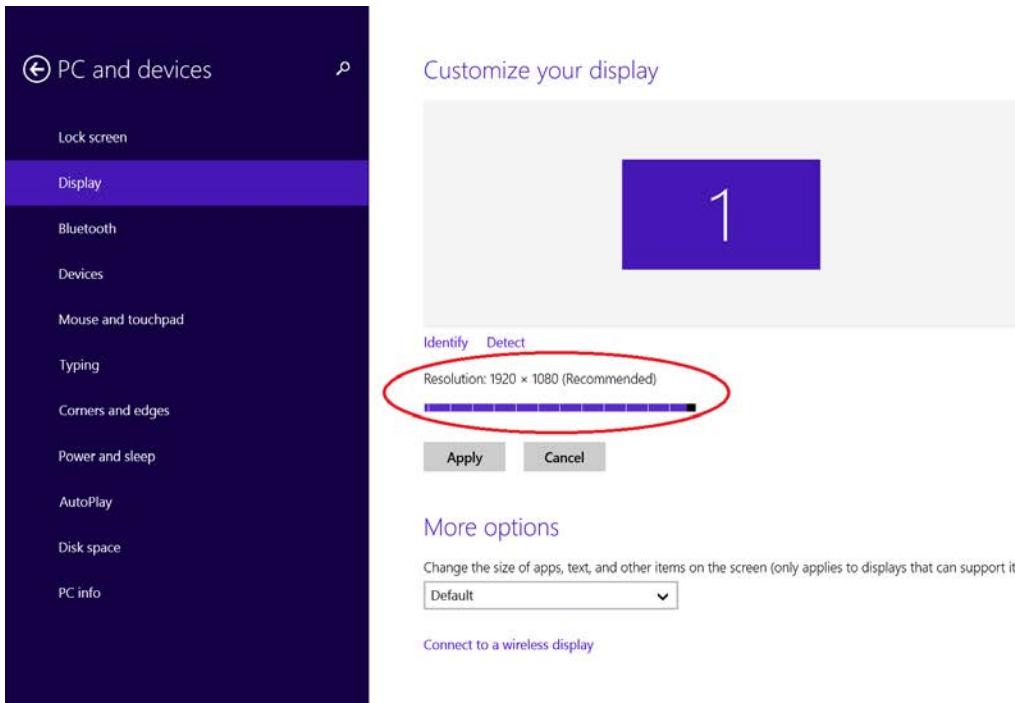


Requirement 4: At least 1 GB for Android SDK, emulator system images, and caches. Android Studio uses the Android SDK and thus requires hard disk space for the SDK. Since 414 GB (refer to the image above) is greater than 1GB, the requirement has been met. The total hard drive space required for the IDE installation mentioned in this document is 1GB + 400MB, which is 1.4GB. Since 414 GB is larger, both of the above requirements have been met.

Step 8: To compare the screen resolution, return to the PC Settings page (see steps 4 and 5) and select Display.



Step 9: Note the scroll bar near the center of the window.



Step 10: Drag the cursor left and right until the resolution is greater than or equal to the minimum resolution. As the cursor is being dragged, the 1280 x 800 minimum screen resolution might be shown.

Requirement 5: 1280 x 800 minimum screen resolution. Since the supported width (1920) is greater than (or equal to) the minimum required width (1280) and the supported height (1080) is greater than (or equal to) the minimum required height (800), the requirement has been met.

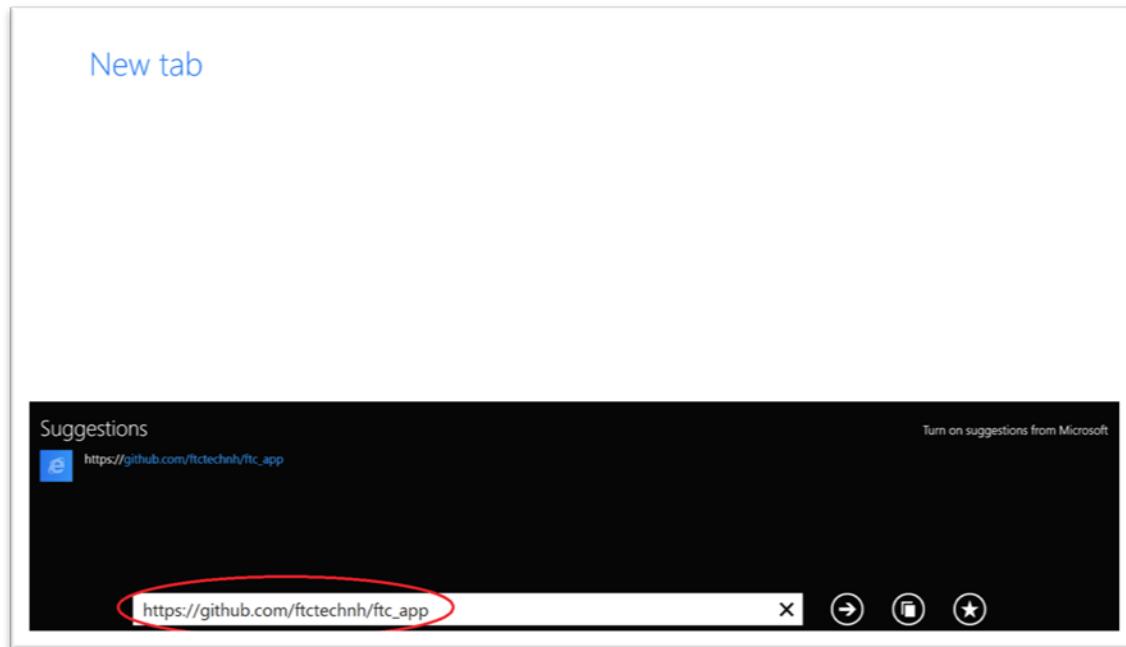
Requirement 6: Java Development Kit (JDK) 7. Since this document describes the process to install JDK 7, this requirement will be met.

Optional Requirement 7: Since the last requirement is optional, this document will not determine whether this requirement has been met.

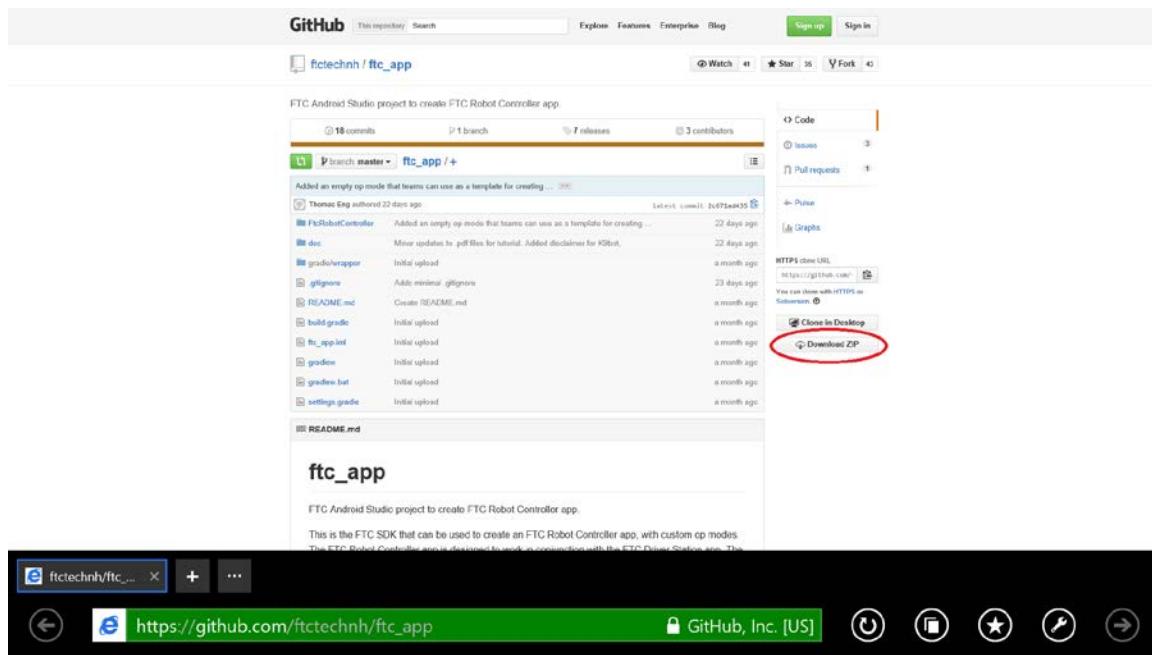
### ***Installing the FTC SDK***

QualComm developed the FTC SDK (Software Development Kit) to allow a cell phone to communicate with other electronic components such as the DC motor controller, the servo controller, and the legacy module. This SDK allows Teams to write Java classes known as op-modes, which allow Teams to customize the functionality of the Robot. These Op modes are compiled using the SDK and deployed to the cell phone using Android Studio. The FTC SDK must be installed onto the same machine that will host the Java SDK and the Android Studio IDE. The FTC SDK is located at [https://github.com/ftctechnh/ftc\\_app](https://github.com/ftctechnh/ftc_app).

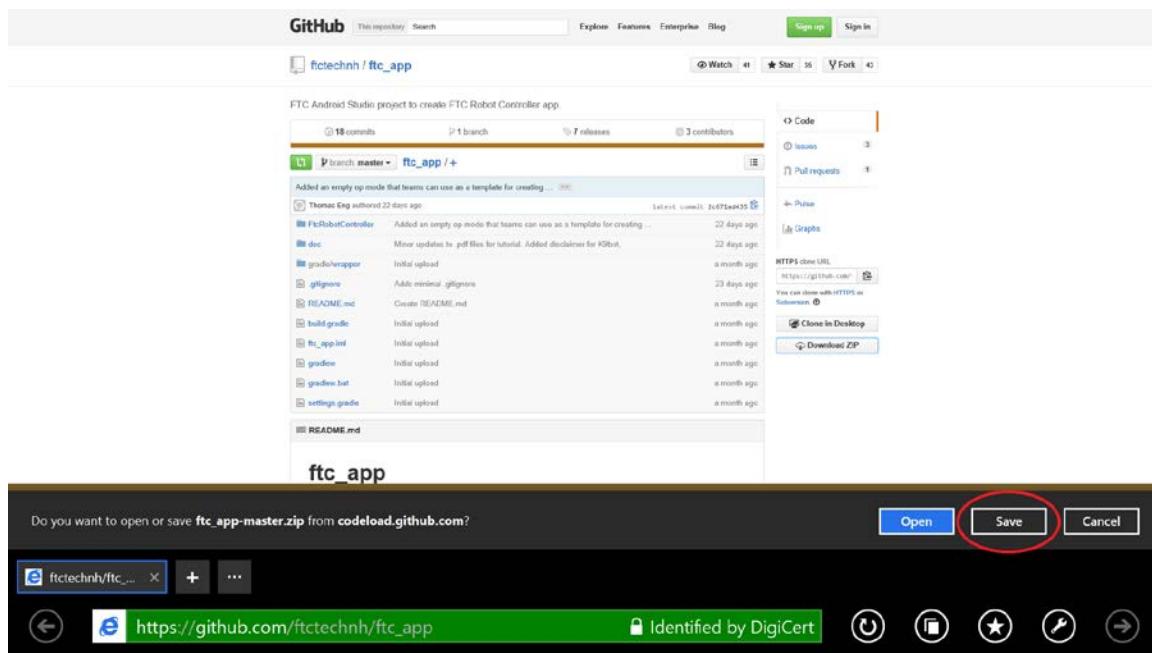
Step 1: Open Internet Explorer and enter '[https://github.com/ftctechnh/ftc\\_app](https://github.com/ftctechnh/ftc_app)' into the Address field.



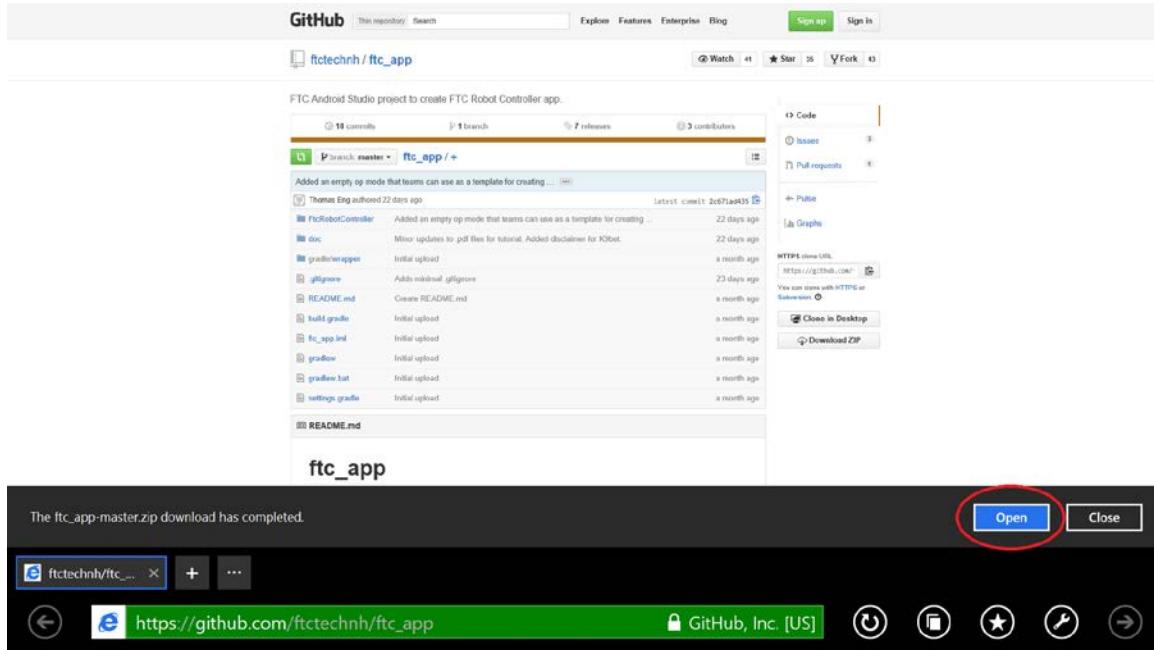
Step 2: After entering the URL from the above step, the following page is shown. Select the “Download ZIP” option.



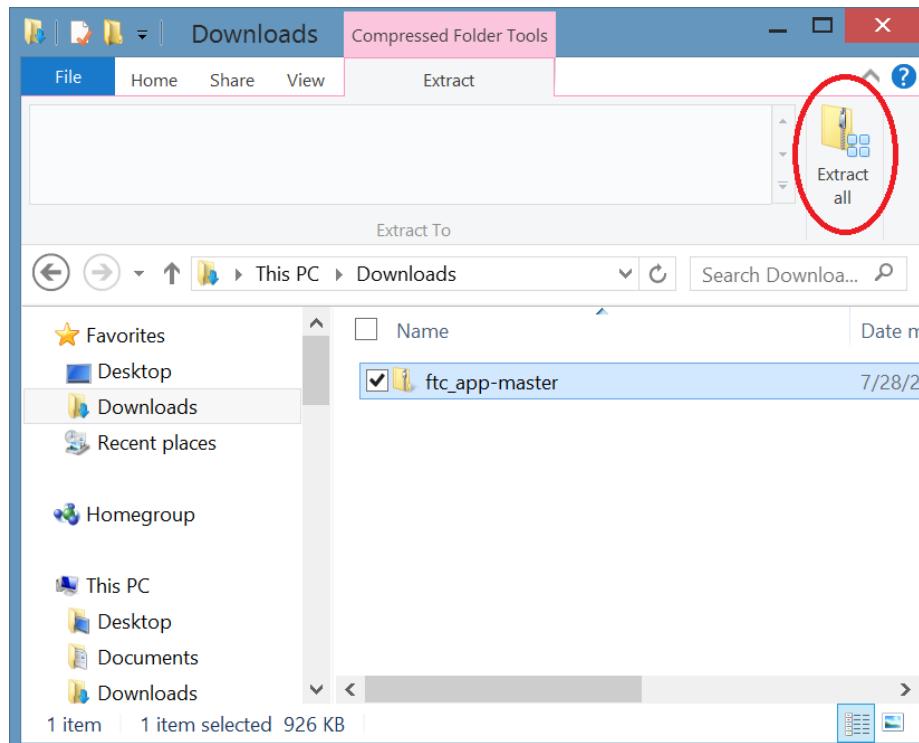
Step 3: After selecting the Download ZIP button, the browser displays a banner to open the file using a decompression utility or to save the file to a location on the machine. Select Save.

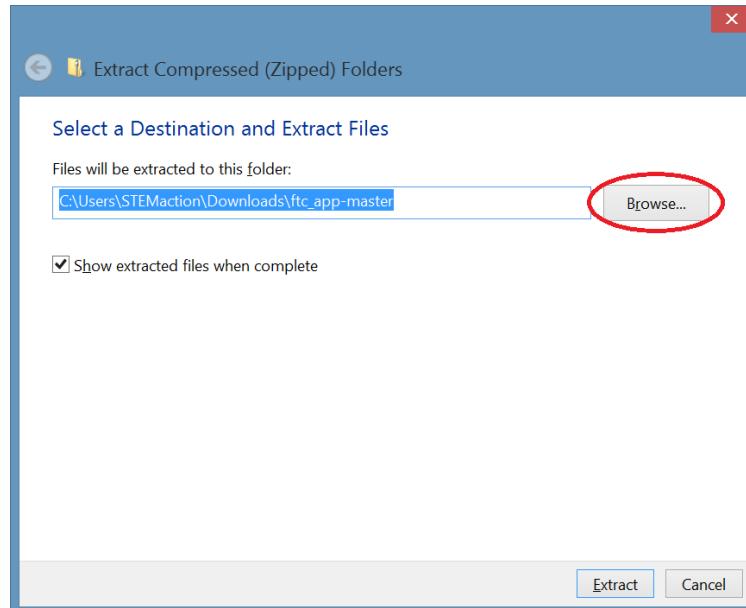


Step 4: After downloading the file, select Open.

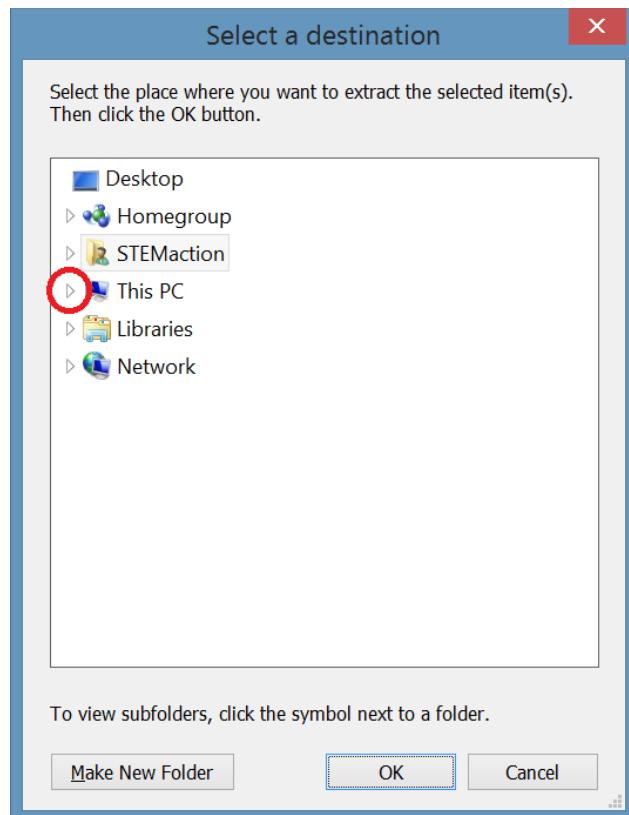


Step 5: After performing the previous step, File Explorer should open. The FTC SDK is a compressed file with a '.zip' extension. The full name of the file is ftc\_app-master.zip or ftc\_app.zip; the name will vary depending on how the zip file is obtained. If Windows is configured to hide file extensions (as shown below) only the first portion of the file name will be shown, the ".zip" extension will be hidden. Notice that Compressed Folder Tools is displayed in the Title Bar. Select 'Extract all'.

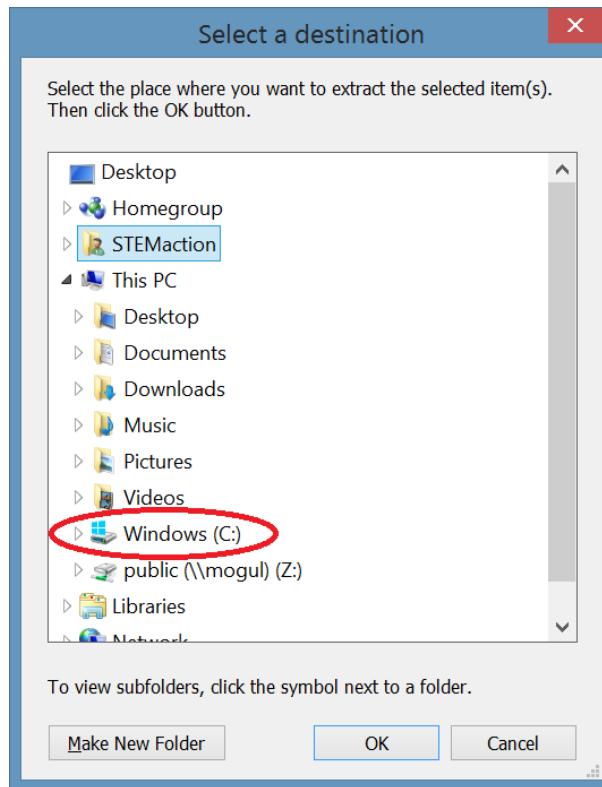


**Step 6: Select 'Browse'.**

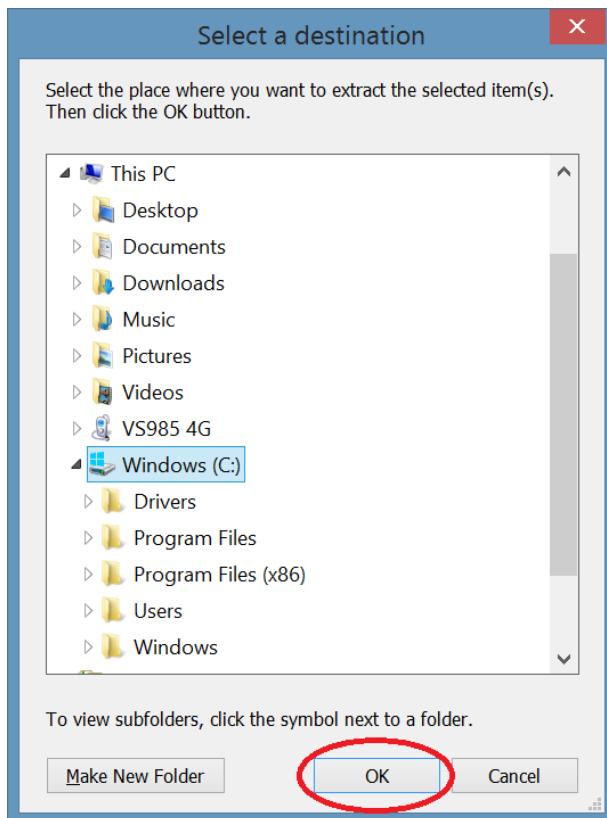
Step 7: Select a directory on your machine's hard drive for the location of the copied file. For this document the "Windows (C:)" folder will eventually be chosen. It will take several steps to find it. Select the small arrow next to 'This PC'.



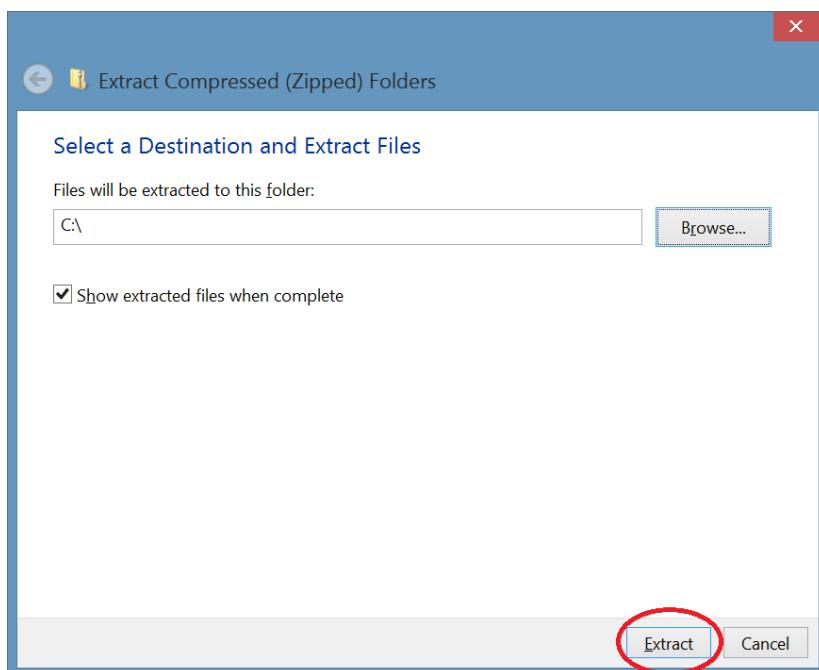
Step 8: Select 'Windows (C:)'.



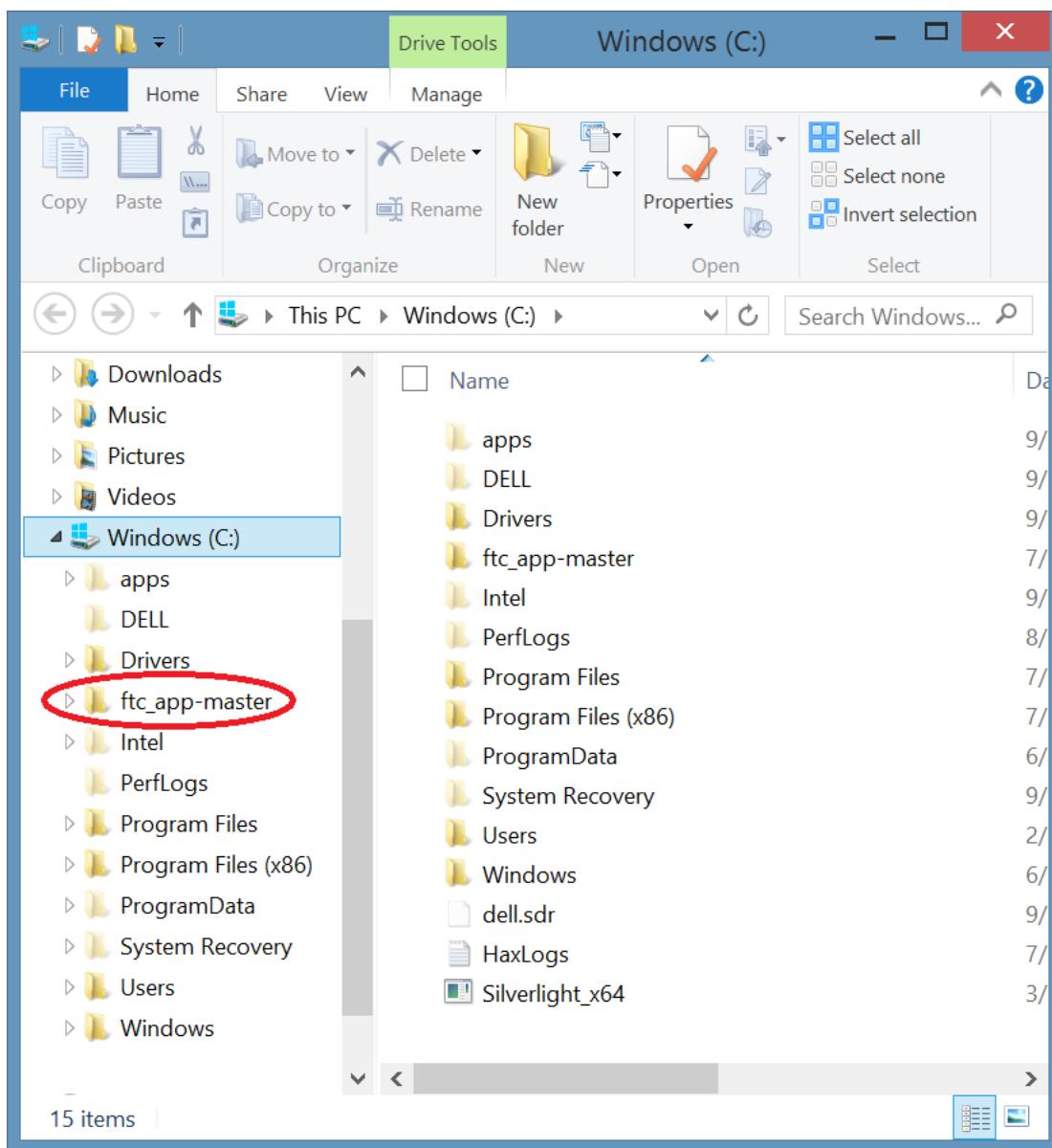
Step 9: Select OK and the dialog will close and the Extract Compressed (Zipped) Folders dialog will become active (shown in the next step).



Step 10: Select 'Extract'. While it is extracting, another window may be shown.



Step 11: The 'ftc\_app-master' folder should then exist under the 'Window (C:)' folder.



## Installing the Java Development Kit (JDK)

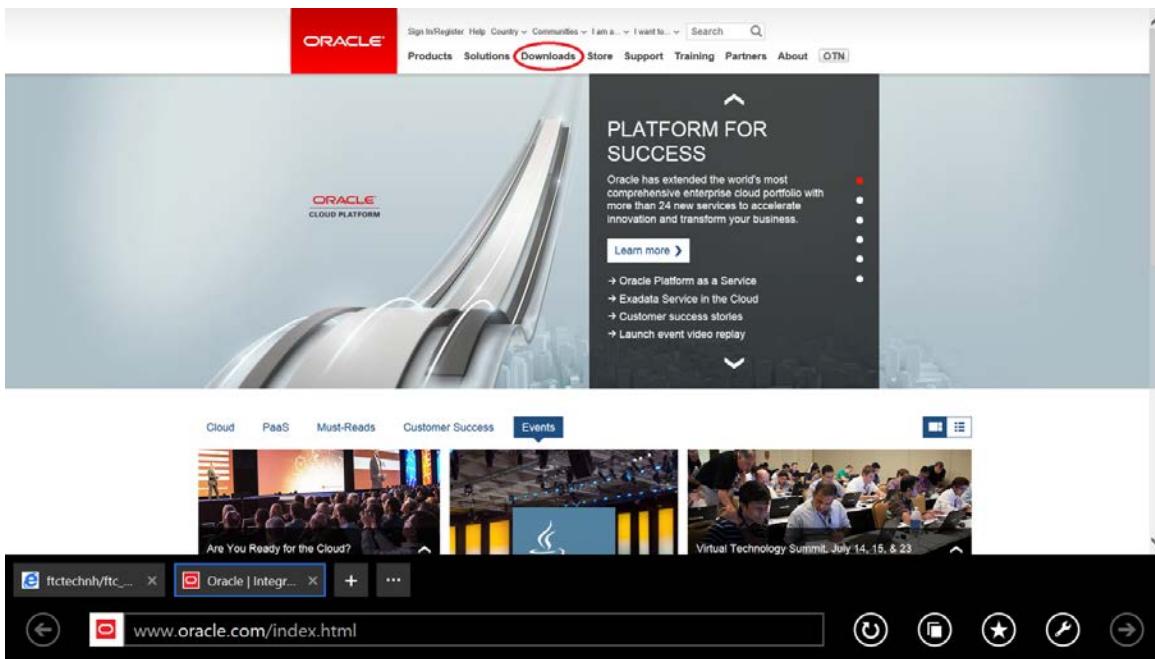
The Java Development Kit must be installed before Android Studio. It provides all of the classes that are used by the FTC driver station and Robot controller applications. This section will describe how to install it.

Step 1: The Java Development Kit (JDK) can be downloaded from the Oracle website at [www.oracle.com](http://www.oracle.com). Open Internet Explorer and enter the URL as shown below.

New tab



Step 2: Select the Downloads button from the top of the page. The menu contains Oracle, Products, Downloads, etc. Please note that the graphics under the menu bar may be different than the following image.



Step 3: Select the Java SE button from the middle right of the page.

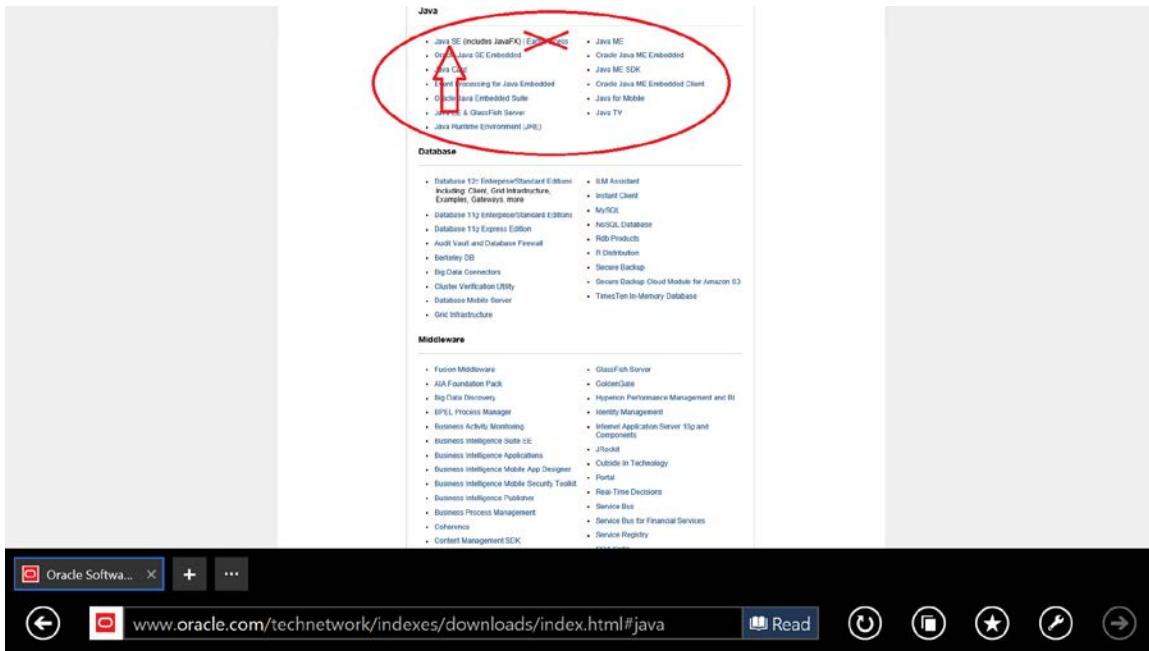
The screenshot shows the Oracle Downloads page. At the top, there's a navigation bar with links for Sign In/Register, Help, Country, Communities, I am a..., I want to..., Search, Products, Solutions, Downloads, Store, Support, Training, Partners, About, and OTN. Below the navigation is a large banner with two men smiling. To the right of the banner are icons for Call and Social. The main content area has a heading "Oracle Downloads" and a sub-section "Top Downloads". On the left, there's a list of categories like Database Downloads, Database 12c, Developer Tools, Application Downloads, Enterprise Management Downloads, and Middleware Downloads. On the right, there's a list of Java-related options: Java Downloads, Java Runtime Environment, Java SE (which is circled in red), Java EE and Oracle GlassFish, and Java for your computer. A "Download Oracle Database 12c now" button is also visible.



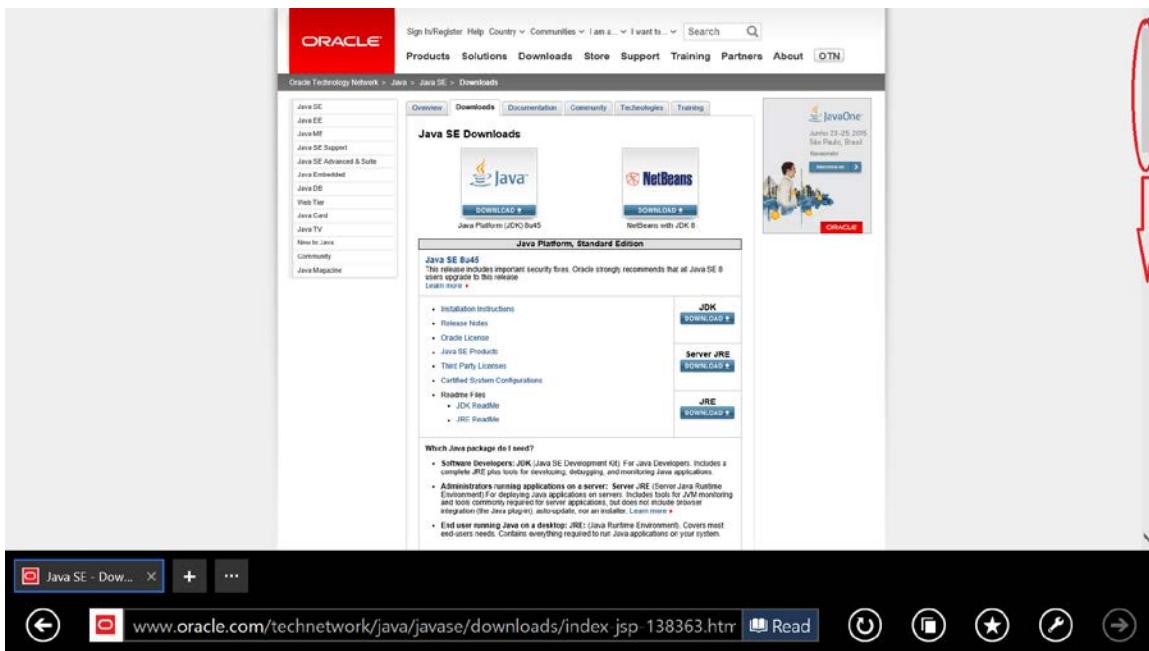
The following image may very briefly be displayed.

This screenshot shows the Oracle Technology Network Software Downloads page. The URL in the browser is <http://www.oracle.com/technetwork/indexes/downloads/index.html#java>. The page layout is similar to the Oracle Downloads page, featuring a navigation bar at the top and a "Software Downloads" section. This section includes sections for Developers (with a note about VMs) and Customers (with a note about Oracle Software Delivery Cloud). On the right side, there's a "Popular Downloads" sidebar listing various Oracle products like Database, Enterprise Manager, and MySQL. Below that is an "Oracle Cloud" section with a "Get Started" button. At the bottom of the page, there's a "Browse by Category: Java" section with a link to "Java SE (Includes JavaFX) Early Access". The "Java SE" link in this section is also circled in red.

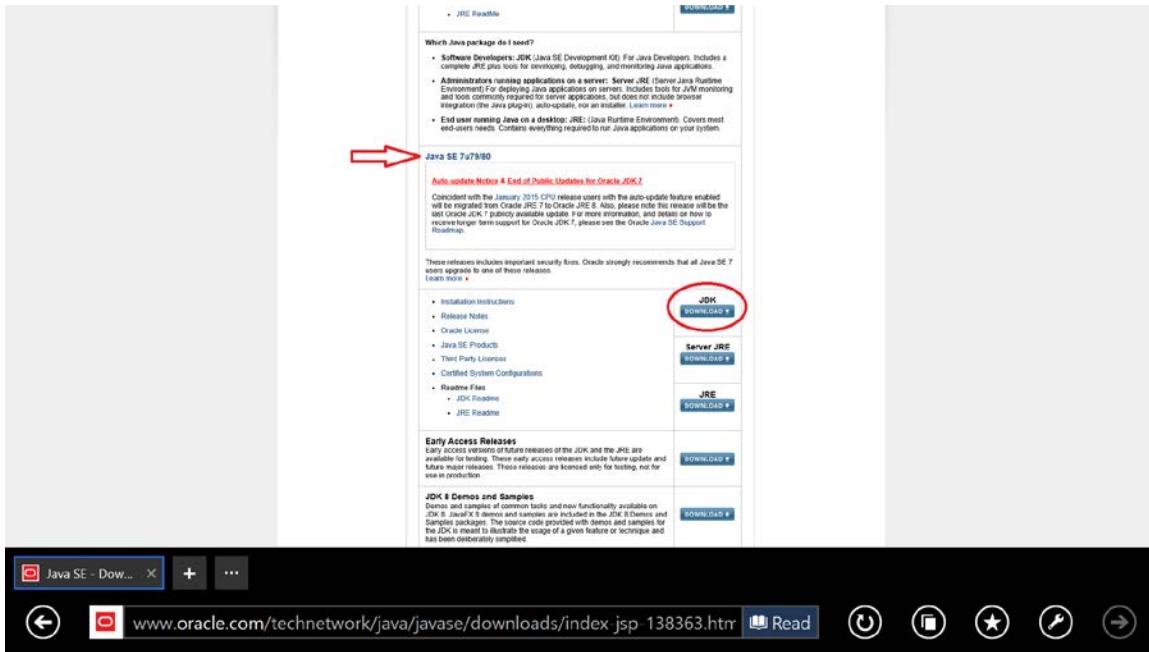
Step 4: If your browser is displaying the above page, then scroll downwards until the Java title is shown. All of the available Java downloads are in one block on the page, as shown below. Select “Java SE”; do NOT select Early Access.



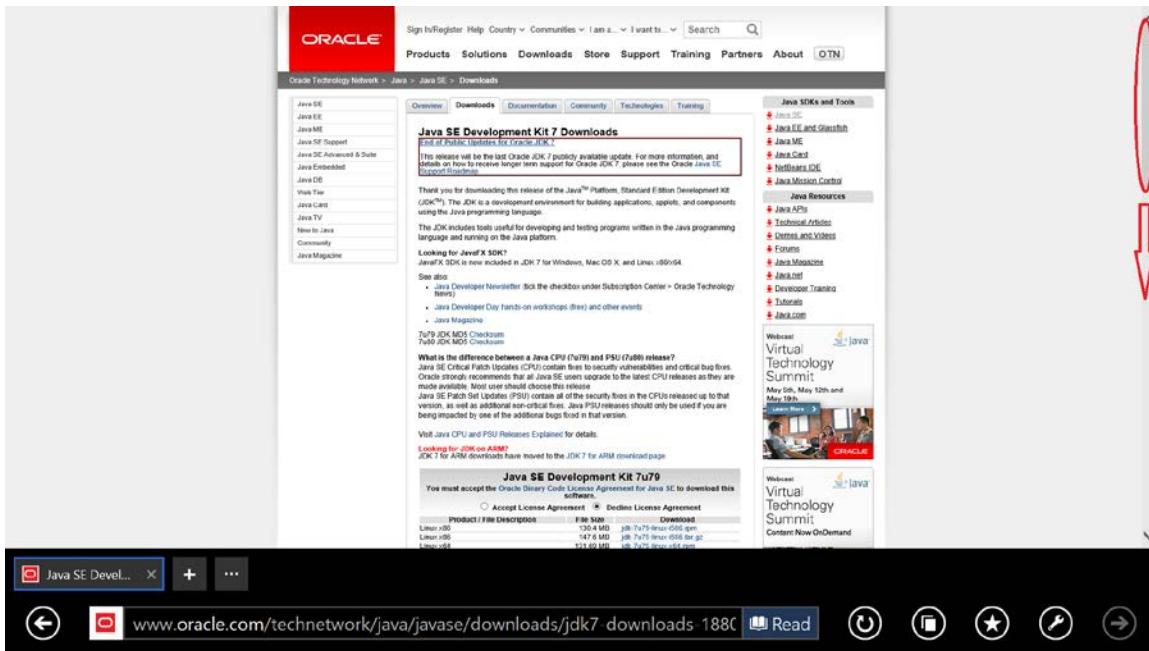
Step 5: After selecting ‘Java SE’, the following page should be shown. Java Version 8 is now the current version. It hasn’t been tested with the FTC SDK, so version 7 should be used instead. Scroll downward until the Java SE 7u[...] block (currently 7u79/80) is displayed (red arrow in the next step’s image). Note that if the scroll bar is not shown, move the mouse slightly.



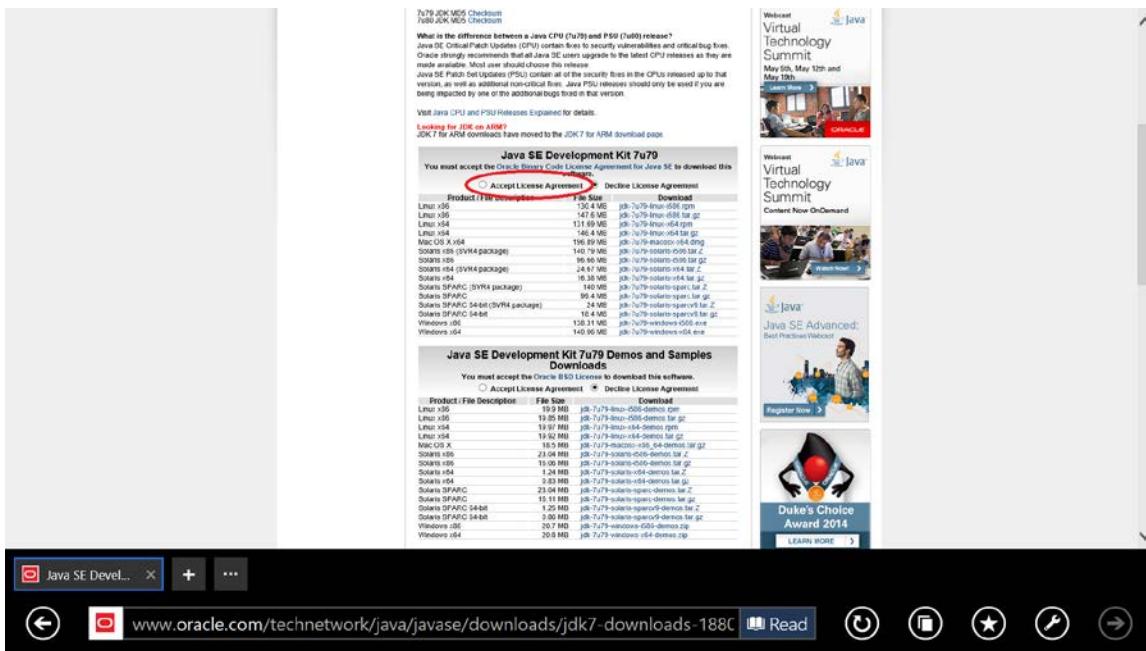
### Step 6: Select the 'JDK Download' button (red oval).



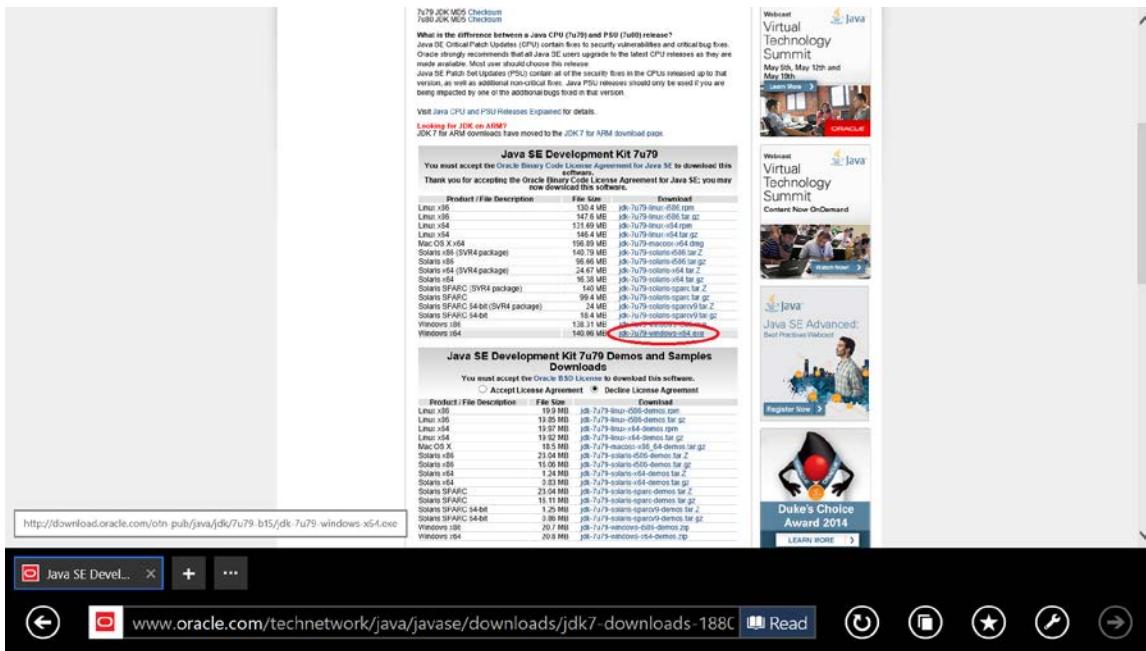
Step 7: The following page should be shown. Scroll downward until the Java SE Development Kit 7u[...] block is displayed (shown in the next step). Note that if the scroll bar is not shown, move the mouse slightly.



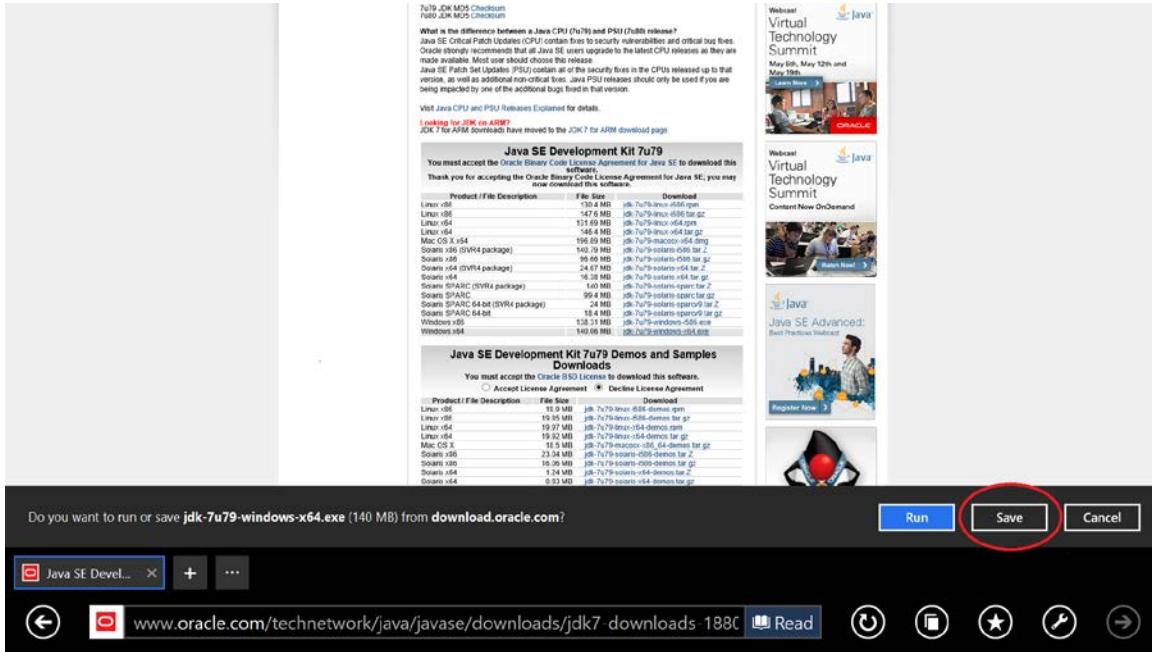
Step 8: Scroll downward until the Java SE Development Kit 7u79 block is displayed. Accept the License Agreement.



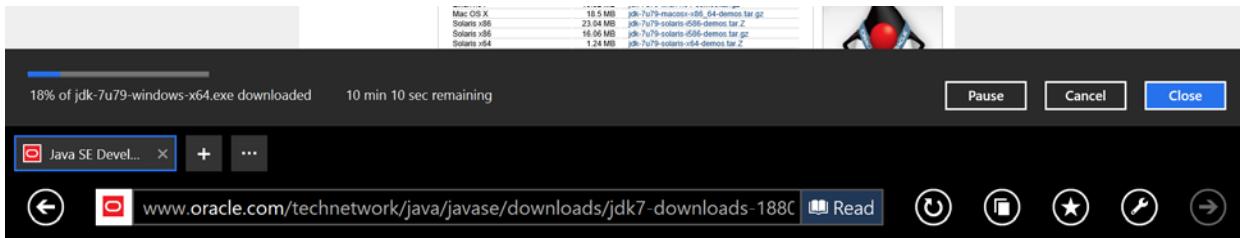
Step 9: Select the download that matches your computer. Refer to section (Determining whether Android Studio can be installed, step 6 to determine the system type. For this example, it is jdk-7u79-windows-x64.exe.



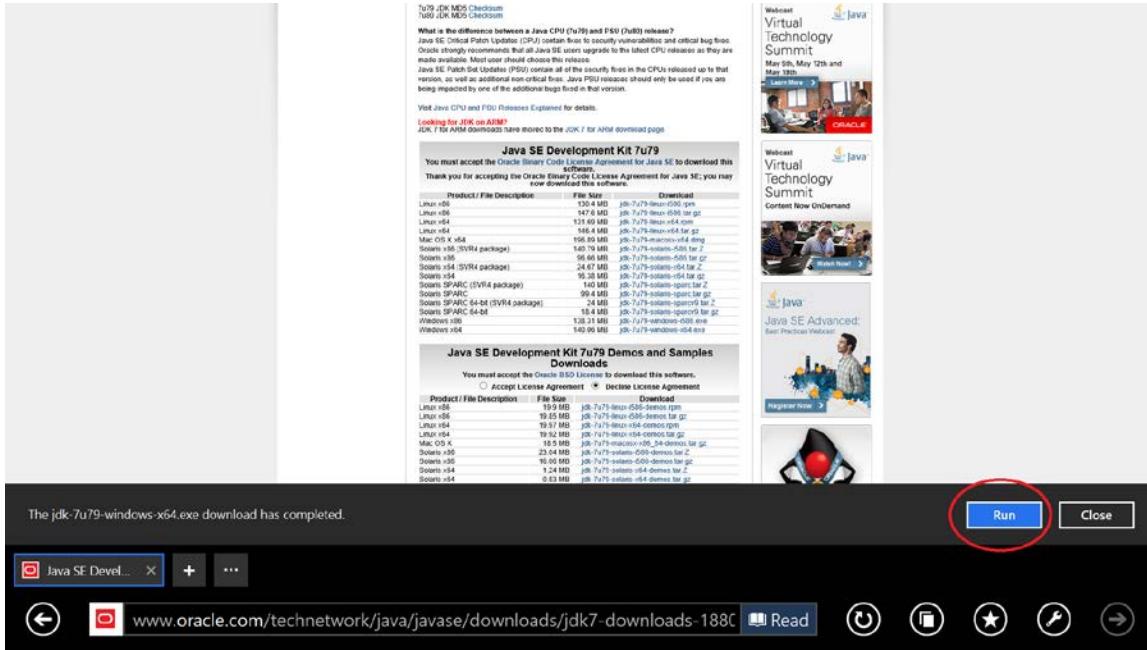
Step 10: Internet Explorer should display a downloading banner. Select Save (arrow).



After selecting save, the download will begin. Internet Explorer will display the estimated amount of time remaining.



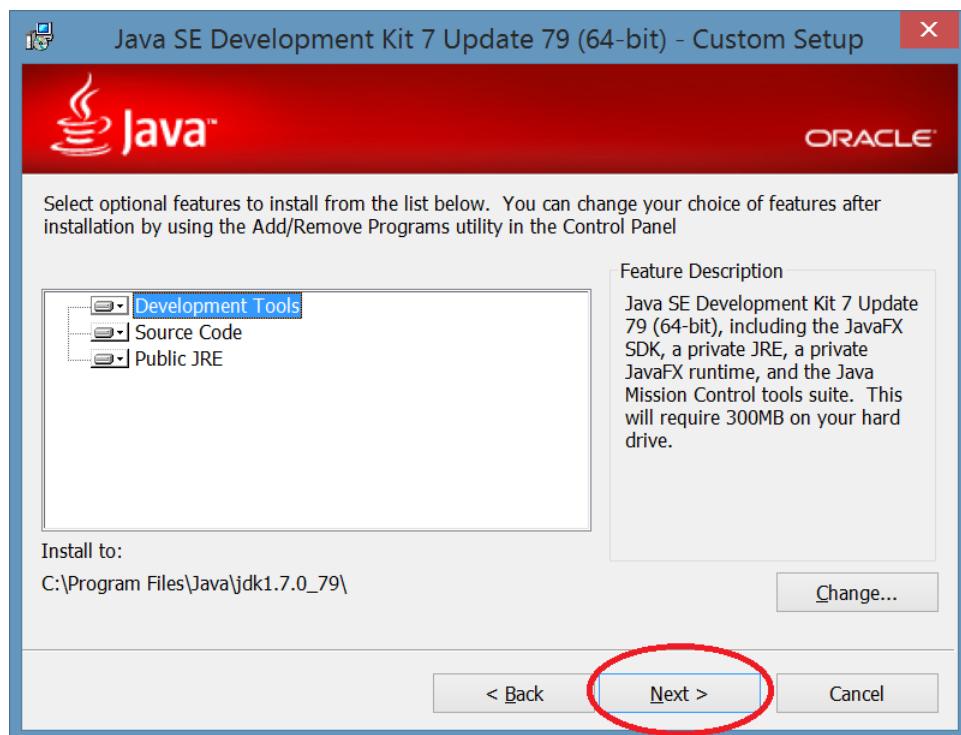
Step 11: After the download is complete, select Run. If a window is displayed that asks “Do you want to allow the following program to make changes to this computer?”, select Yes.



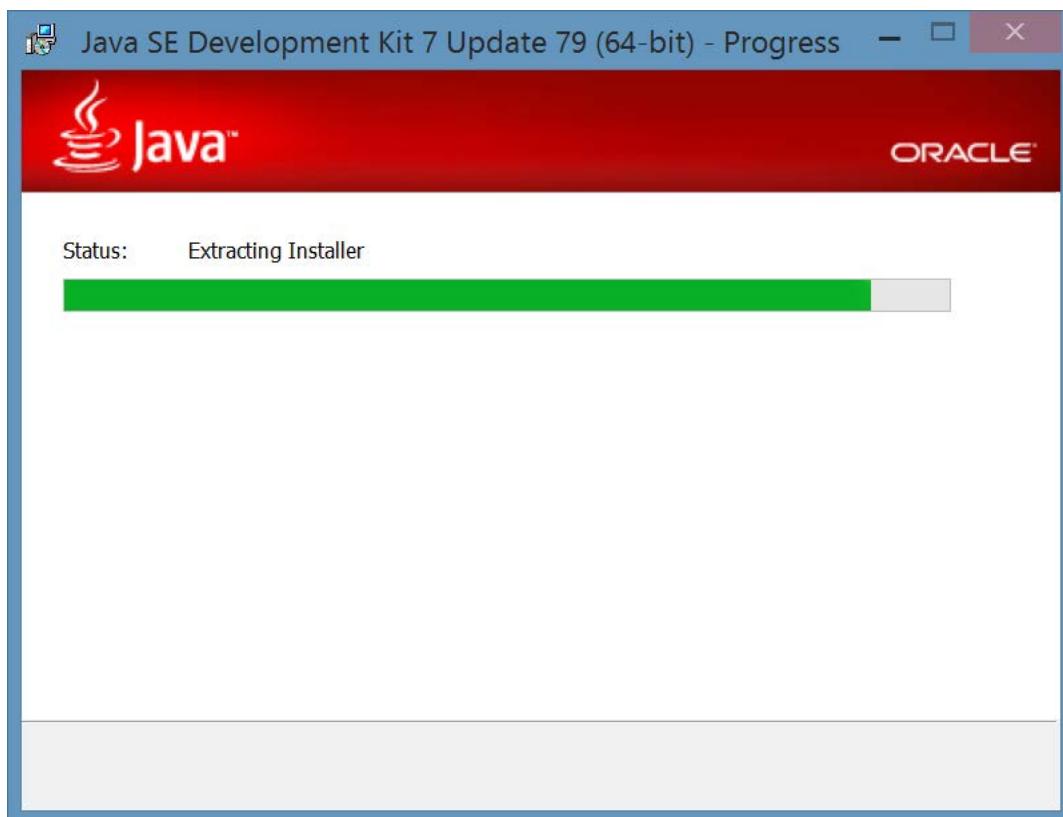
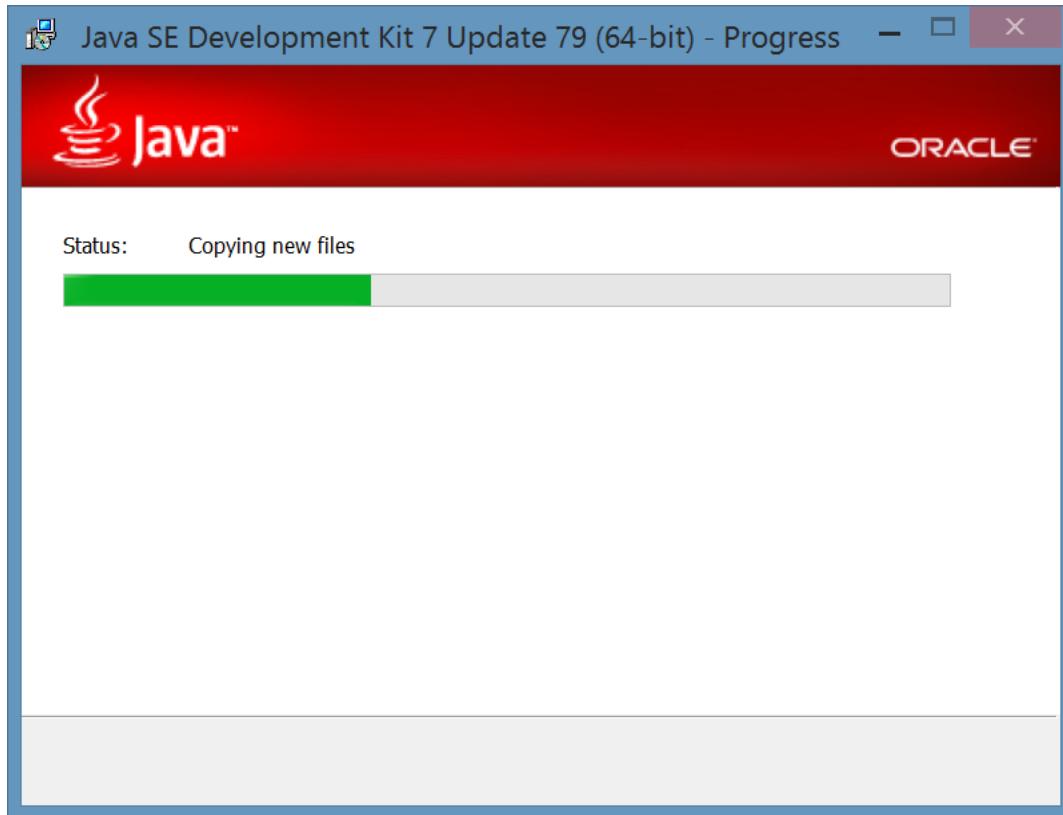
Step 12: The installation of Java will begin by displaying the following window. Select Next.



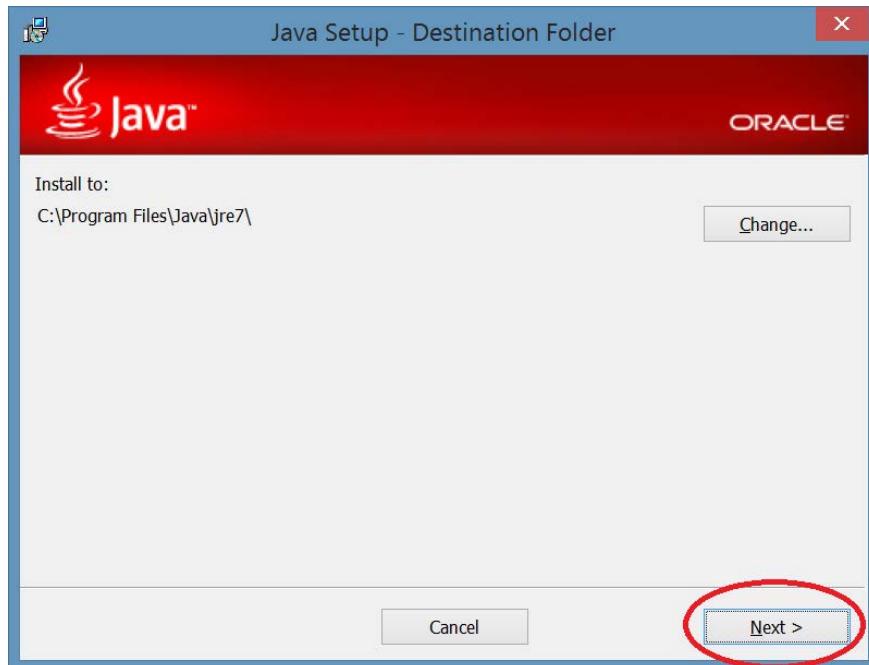
Step 13: The following dialog can be used to select which parts are installed and where the files will be located. Choose Next to accept the default installation.



While the files are being installed the installer will display the following windows.



Step 14: The installation continues by selecting where the files will be located. Choose Next to accept the default installation.



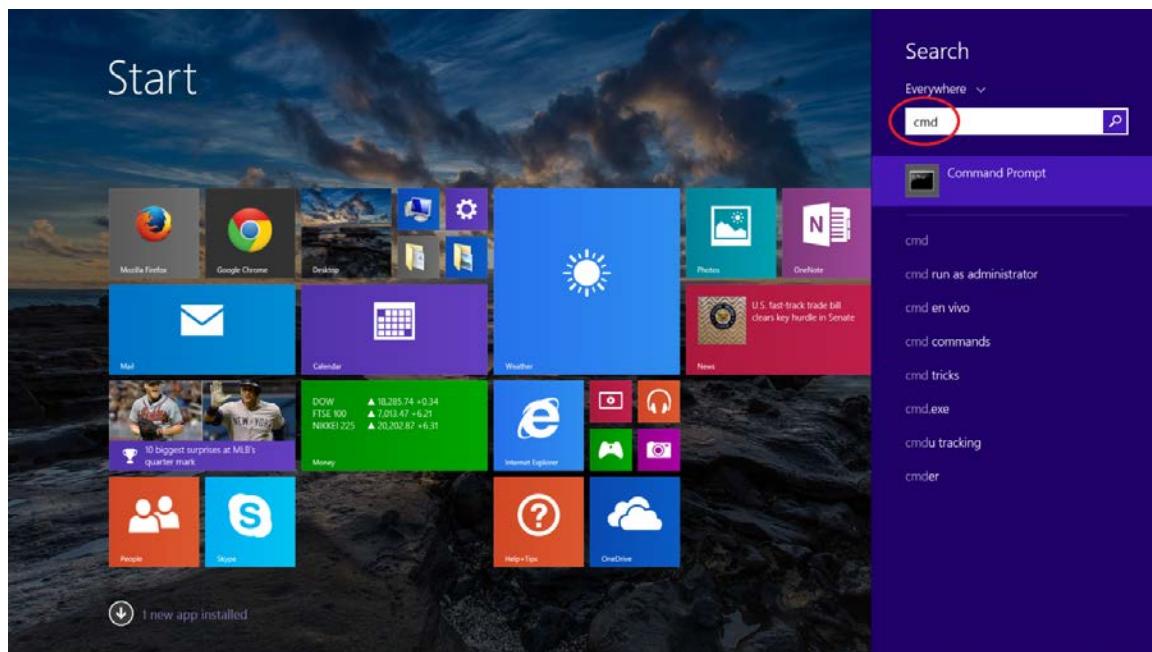
While the files are being installed the following dialog will be displayed.



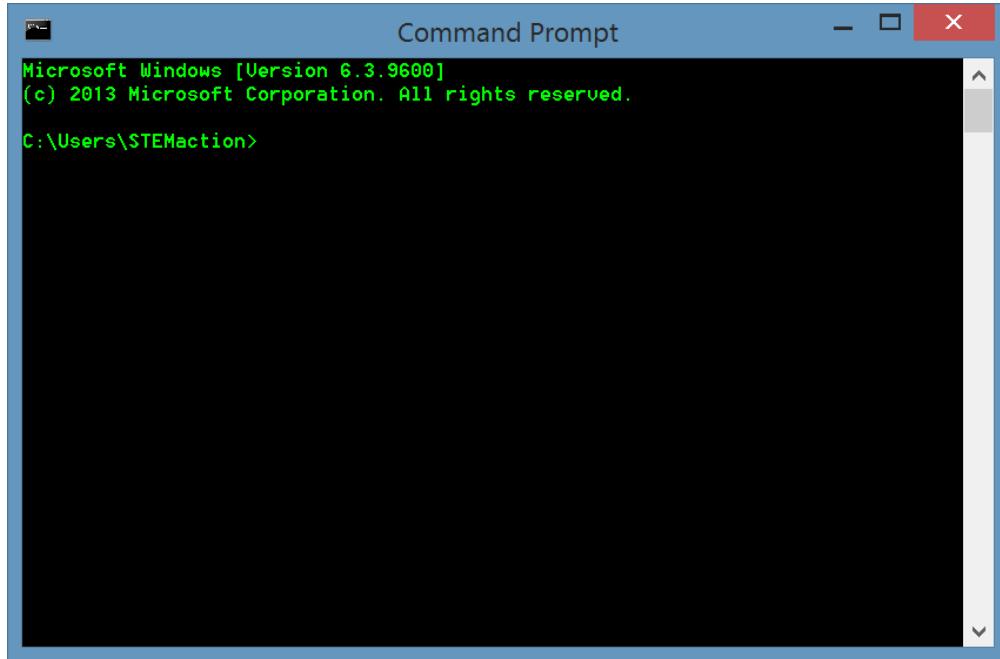
Step 15: After the installation is complete, choose Close to exit the installer.



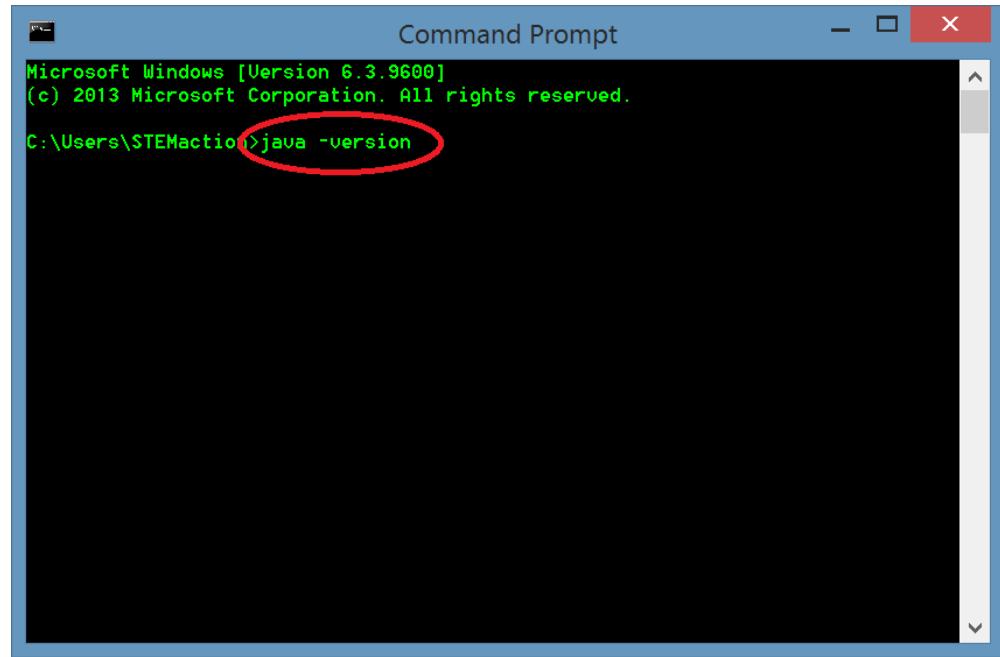
Step 16: To determine whether the installation was a success, use the command prompt. Press the Windows key and type 'cmd'. Remember to press enter after typing the name.



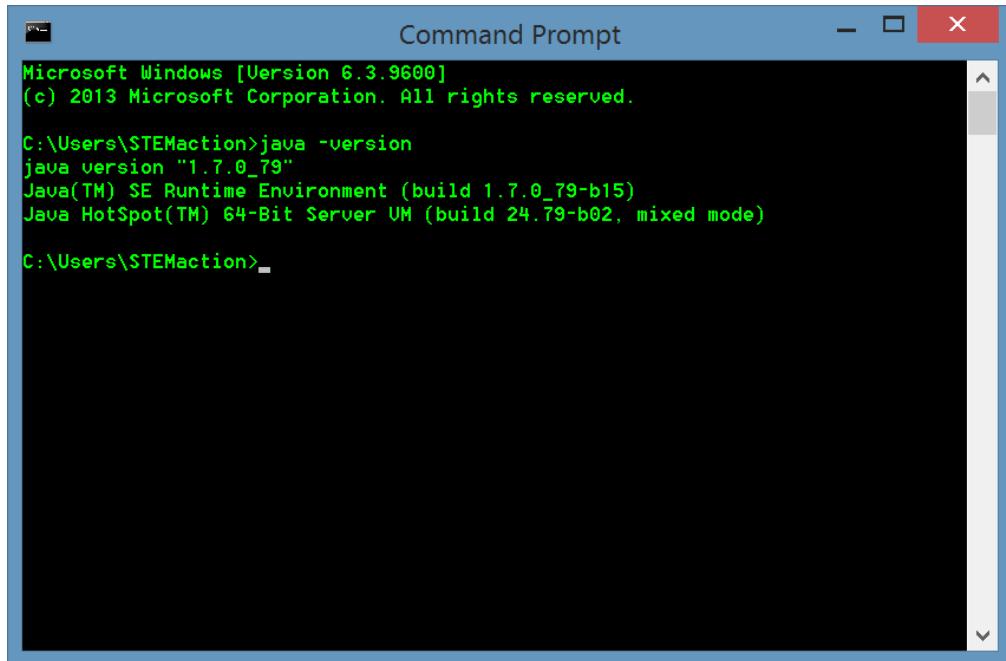
The following window will be displayed.



Step 17: Type “java –version” and press return/enter



Step 18: Verify that the version is “1.7.XXX”. The close button (  ) can be used to close the window.



Microsoft Windows [Version 6.3.9600]  
(c) 2013 Microsoft Corporation. All rights reserved.  
  
C:\Users\STEMaction>java -version  
java version "1.7.0\_79"  
Java(TM) SE Runtime Environment (build 1.7.0\_79-b15)  
Java HotSpot(TM) 64-Bit Server VM (build 24.79-b02, mixed mode)  
  
C:\Users\STEMaction>~

### ***Installing the Android Studio Integrated Development Environment (IDE)***

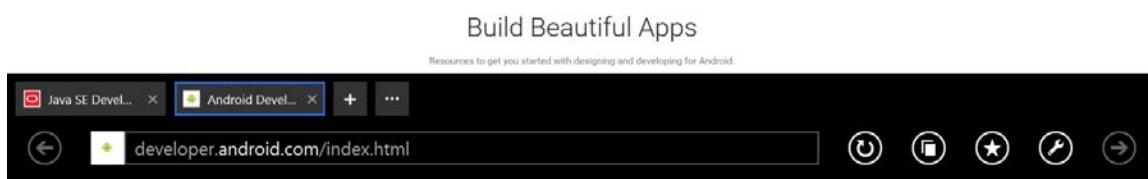
The FTC SDK and Java Development kit have previously been installed. The next task is to install the Android Studio IDE.

Step 1: The IDE is available for Windows, Mac OS X and Linux and can be downloaded from the <https://developer.android.com> website. Open Internet Explorer and enter the URL as shown below.

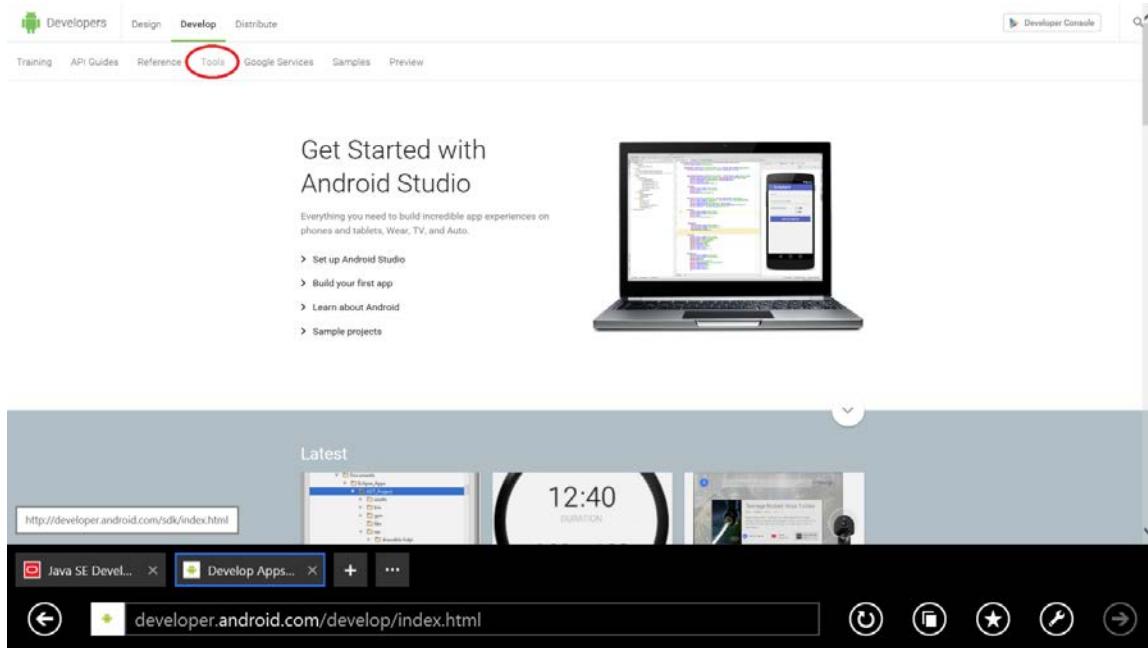
New tab



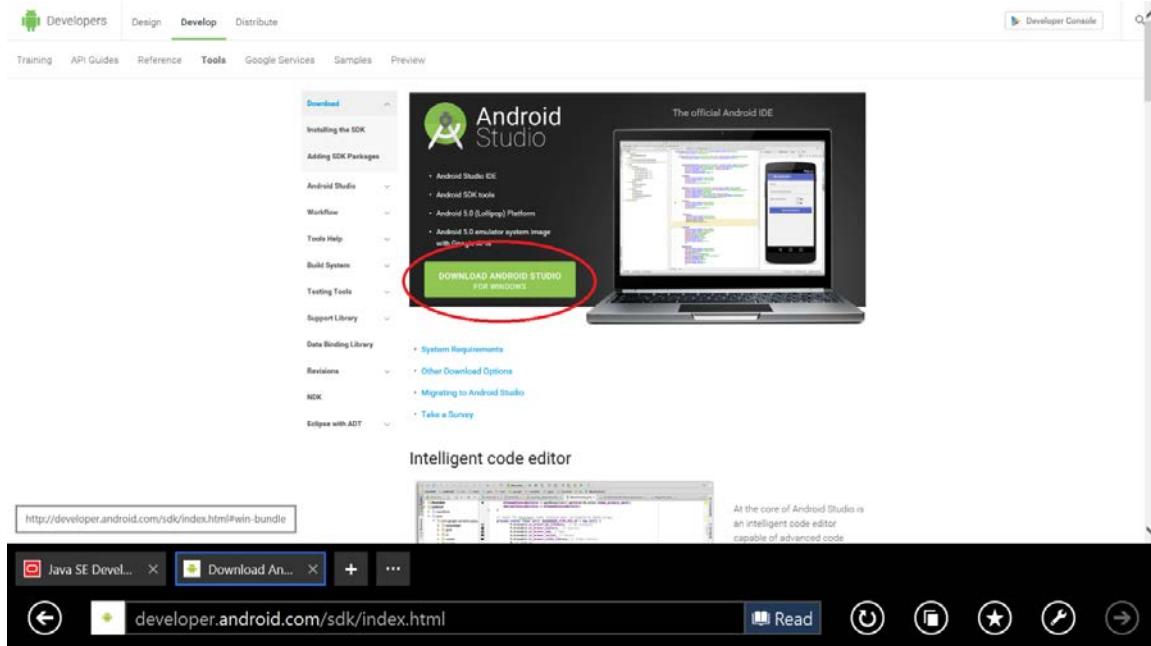
## Step 2: Select “Develop”.



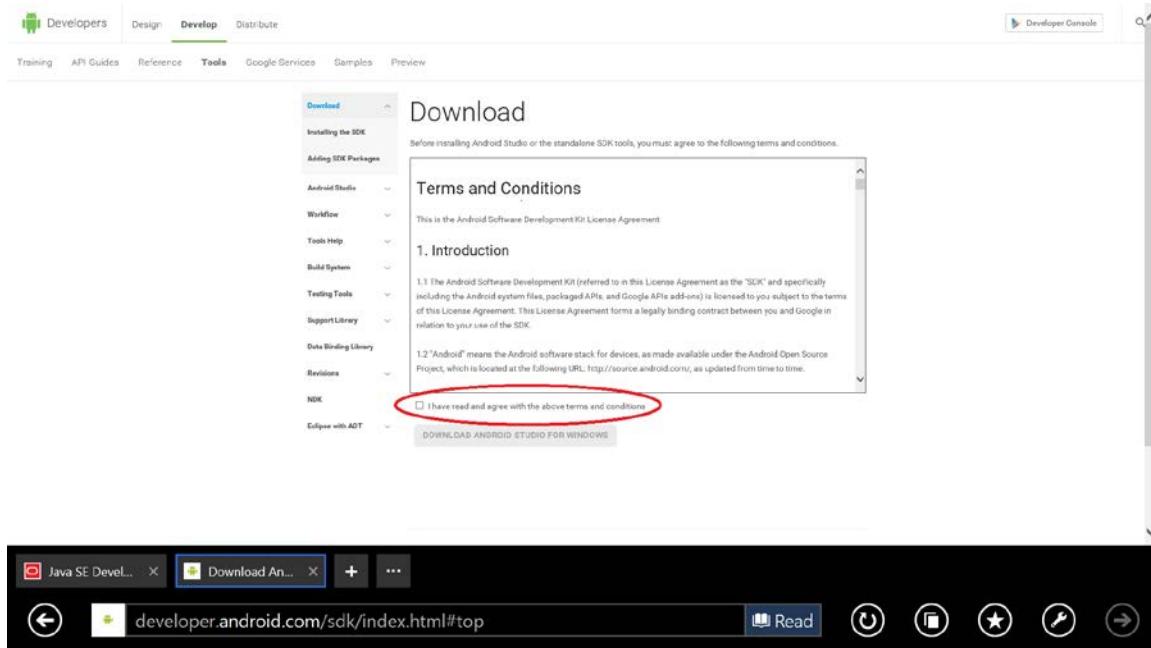
## Step 3: Select Tools on the submenu.



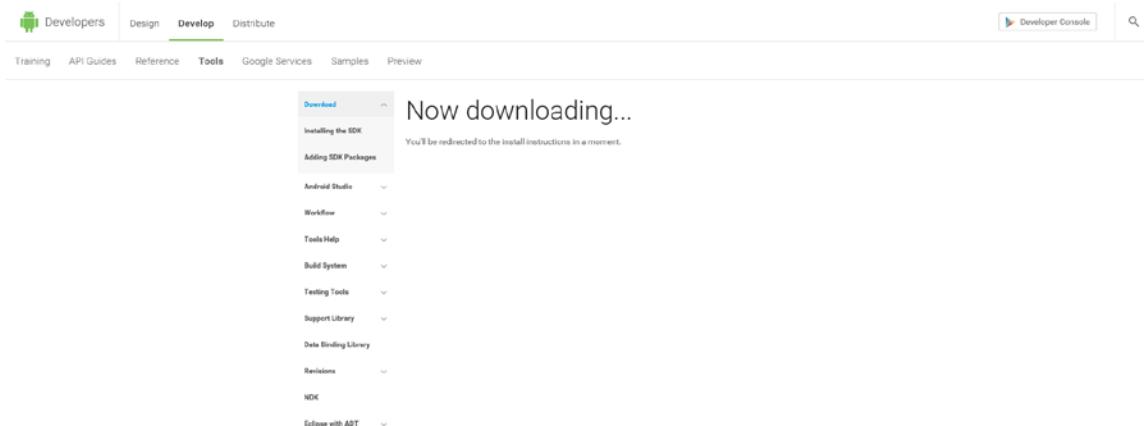
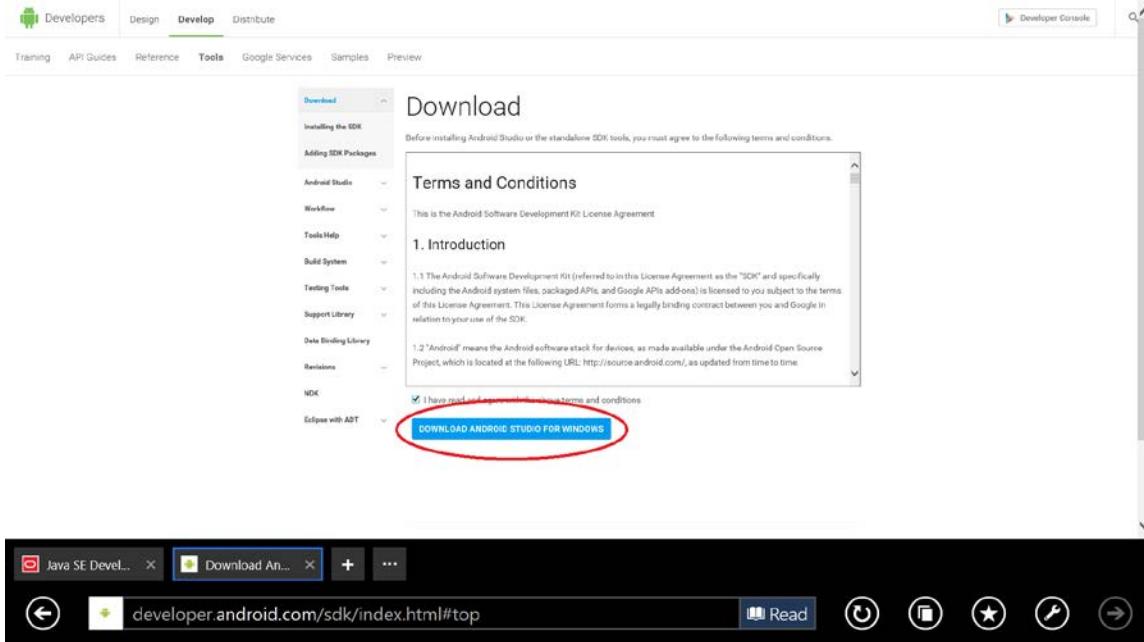
Step 4: Select the green “Download Android Studio for Windows” button.



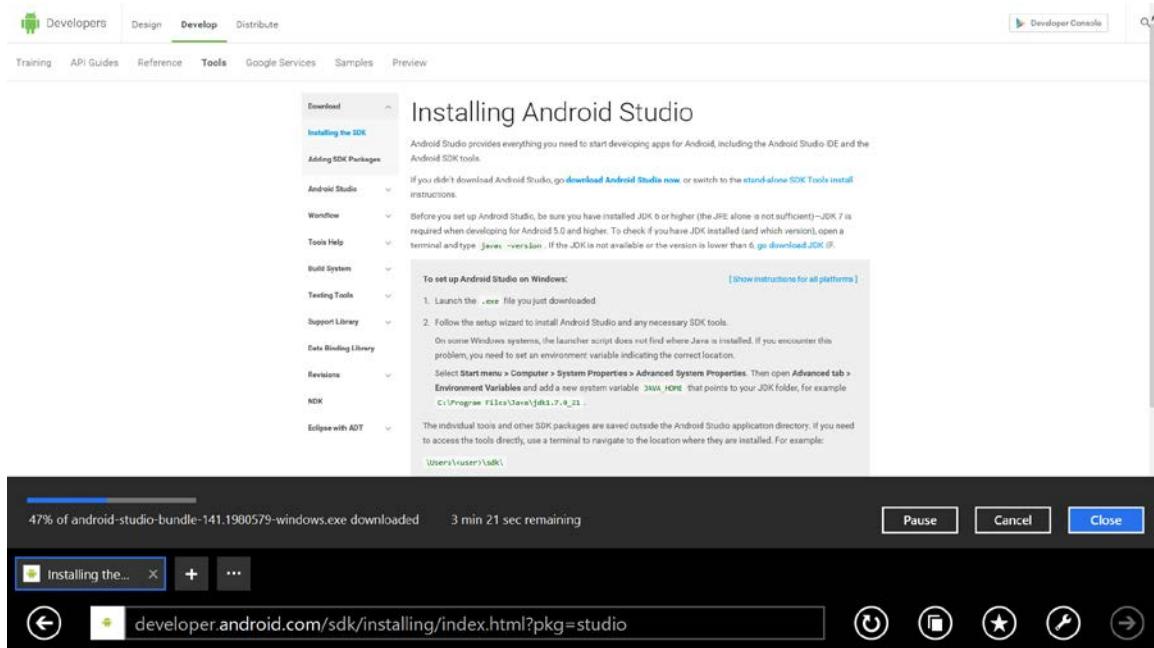
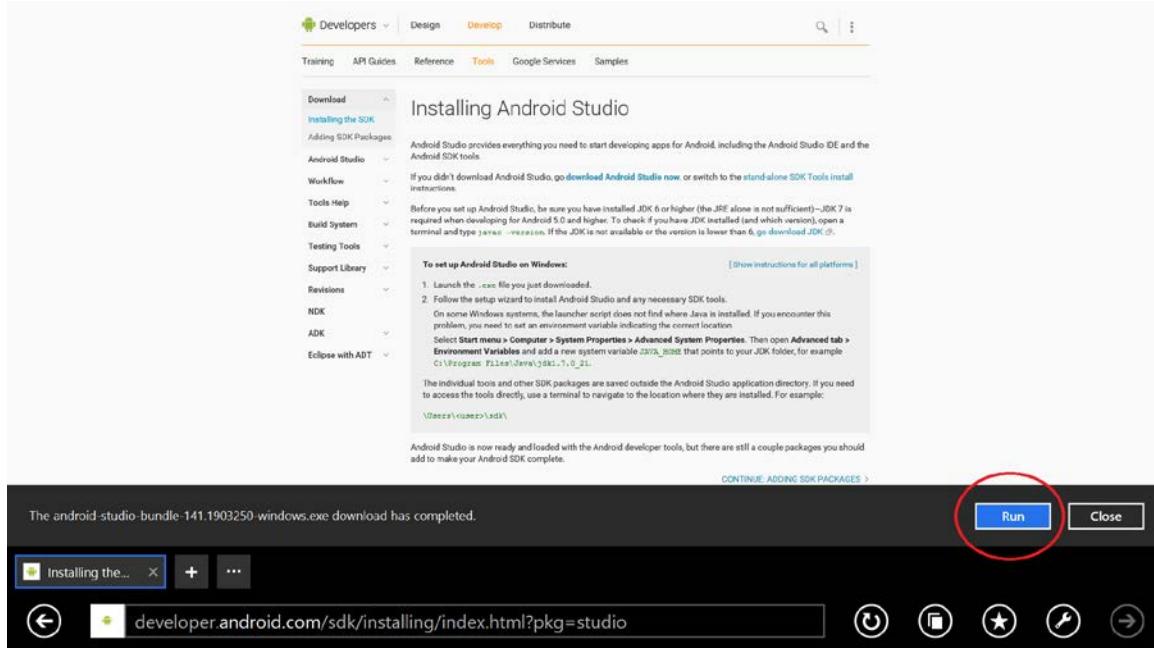
Step 5: Accept the terms and conditions from the next page to begin the download.



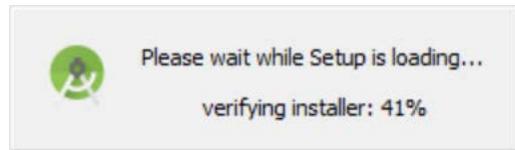
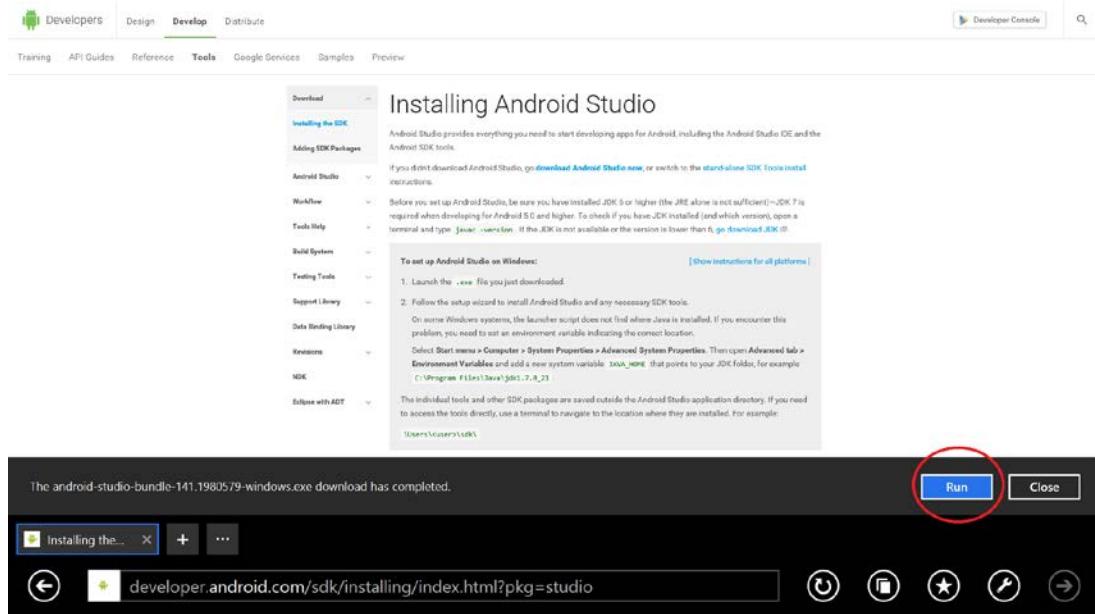
Step 6: After checking the terms and conditions, the download button will be selectable. Select the Download Android Studio for Windows button. Another page might be displayed before the download starts.



Step 7: Internet Explorer should display a downloading banner. Select Save. This might take a while (as shown in the image below the following image).



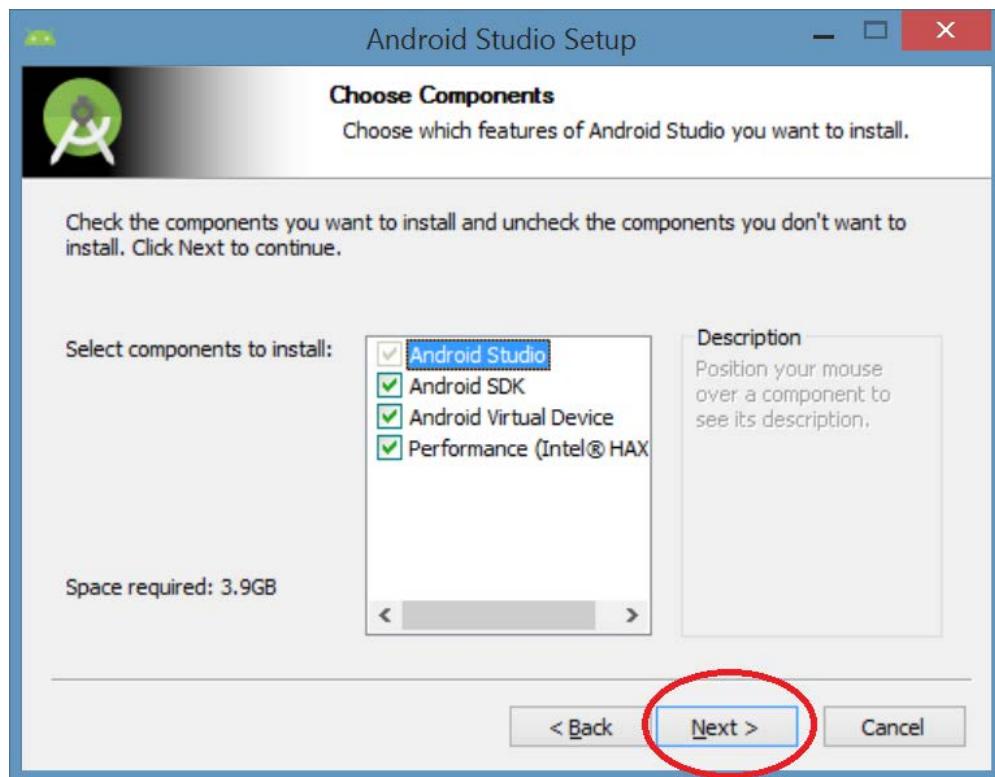
Step 8: After the download is complete, select Run. As the installer is loading, the image below the following image may be shown.



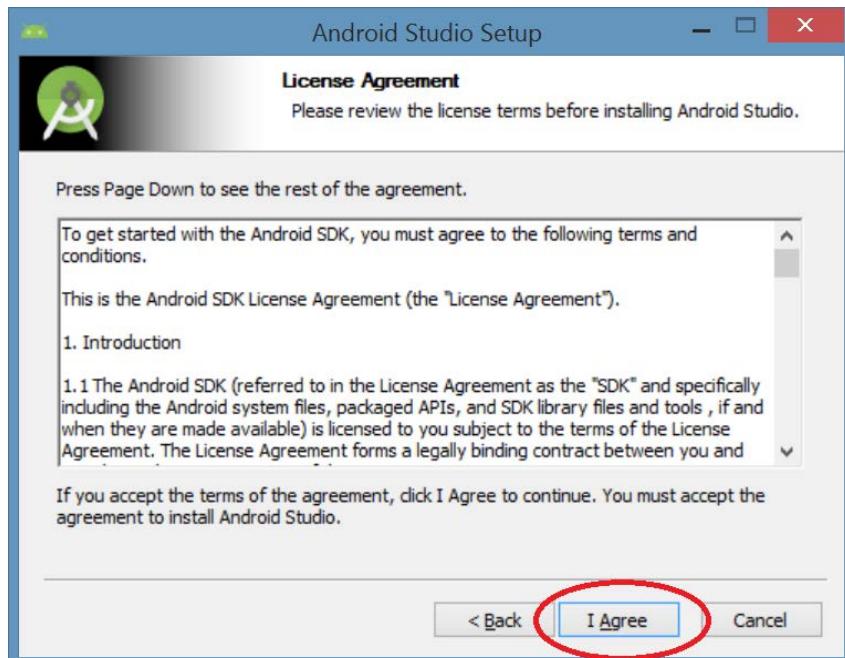
Step 9: To begin the installation process select Next. If a window is displayed that asks “Do you want to allow the following program to make changes to this computer?”, select Yes.



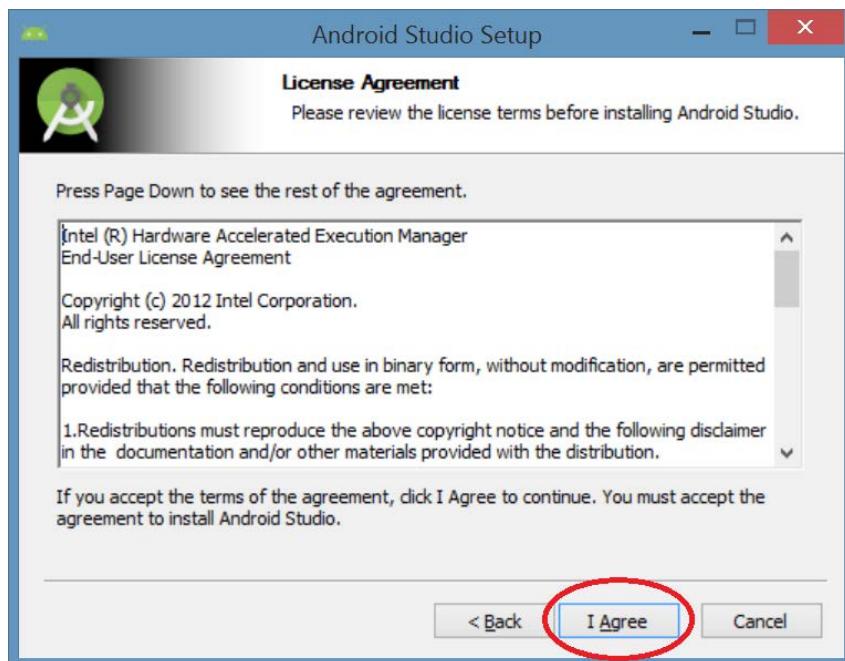
Step 10: The following dialog can be used to select which parts are installed and where the files will be located. Choose Next to accept the default installation.



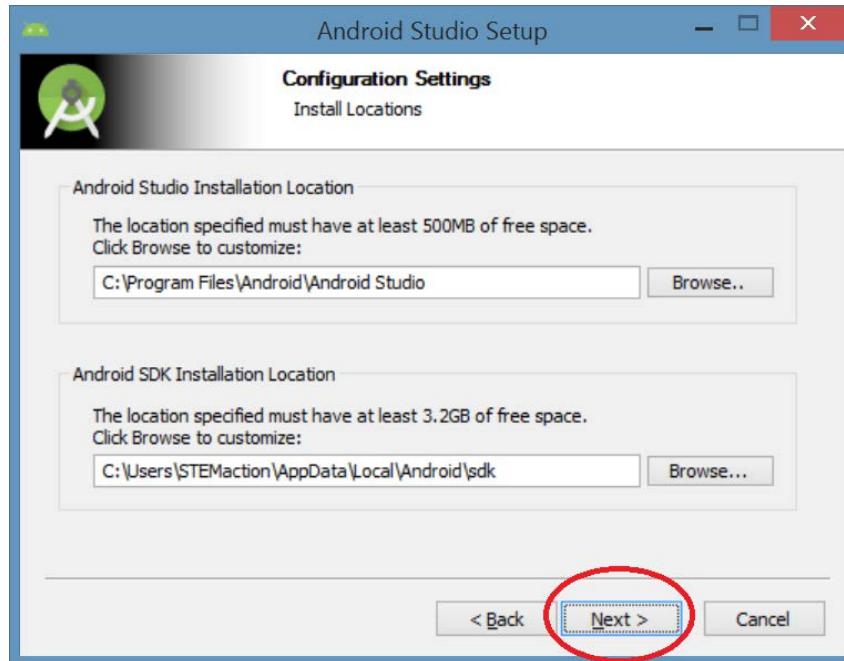
Step 11: Accept the License Agreement for the Android SDK by selecting 'I Agree'.



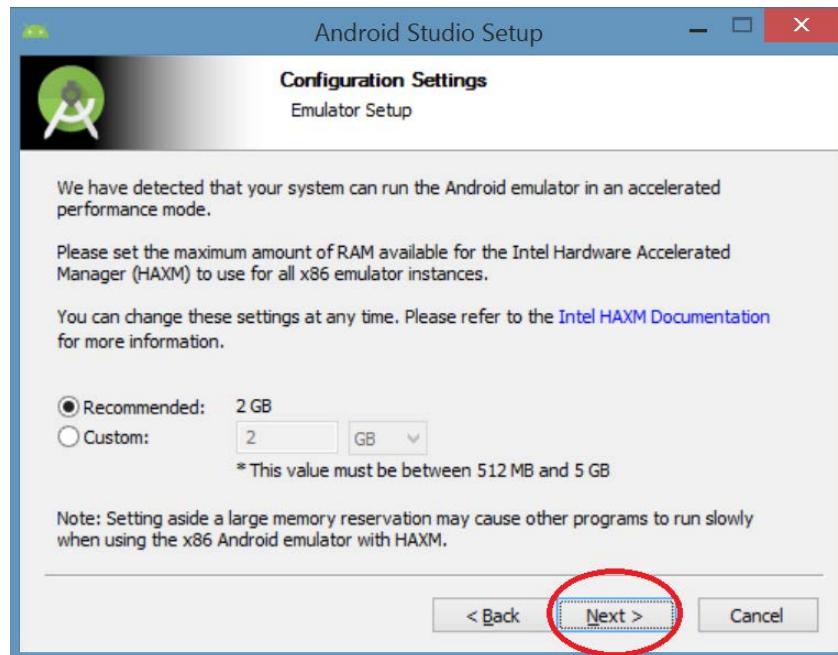
Step 12: Accept the License Agreement for the Android Studio IDE by selecting 'I Agree'.



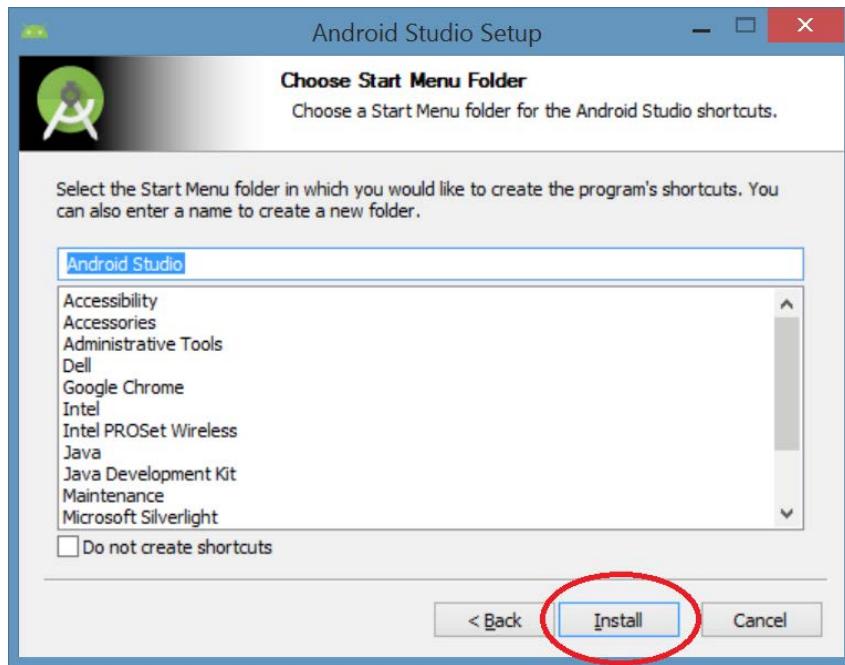
Step 13: Choose Next to accept the default configuration settings.



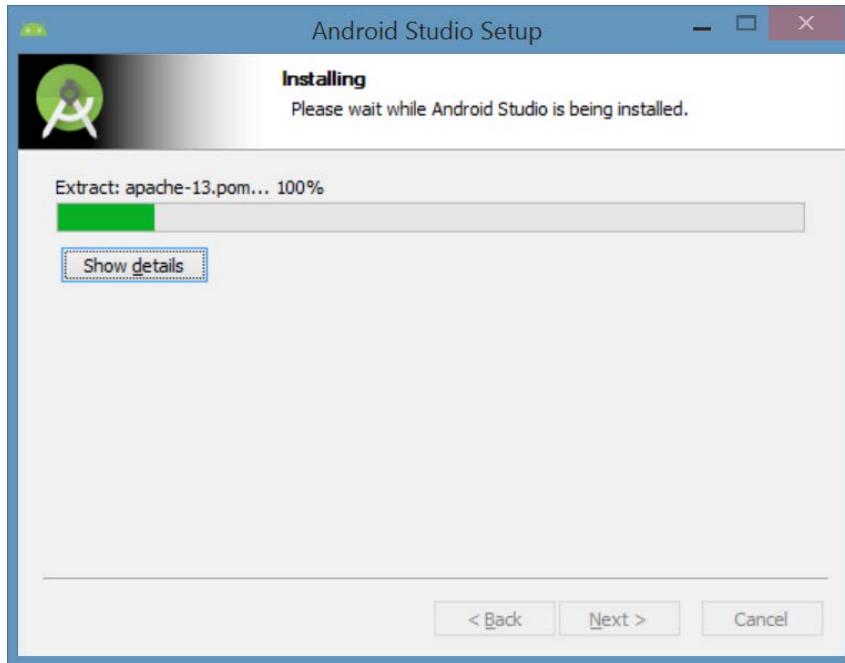
Step 14: Choose 'Next' to accept the default emulator settings.

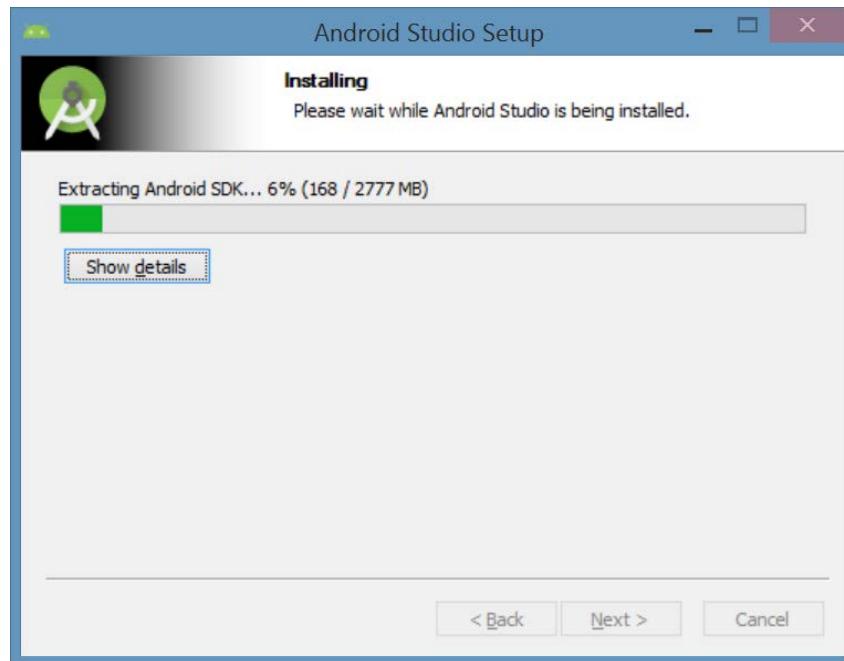


Step 15: Choose 'Install' to accept the default start menu folder.

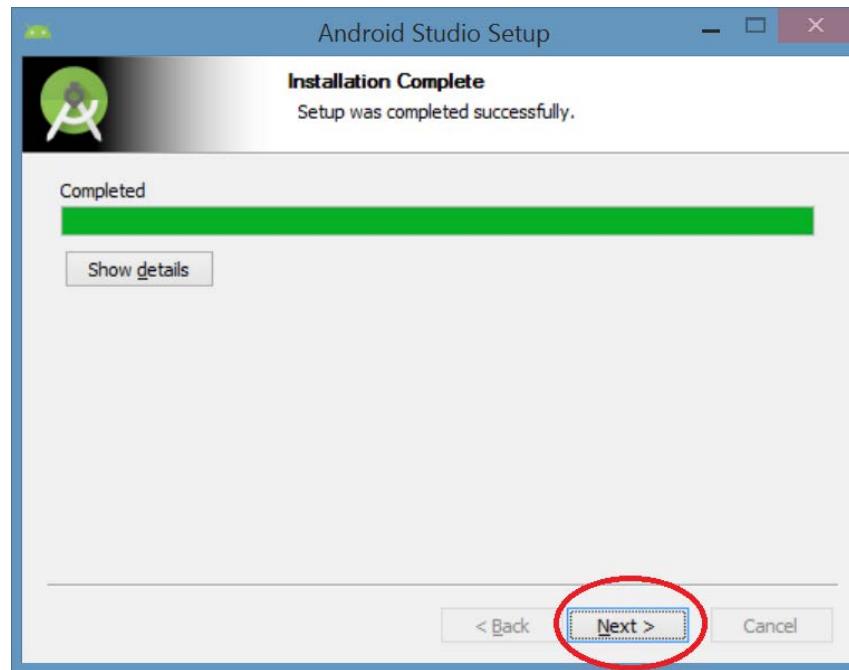


While the install continues, the following pages may be displayed.

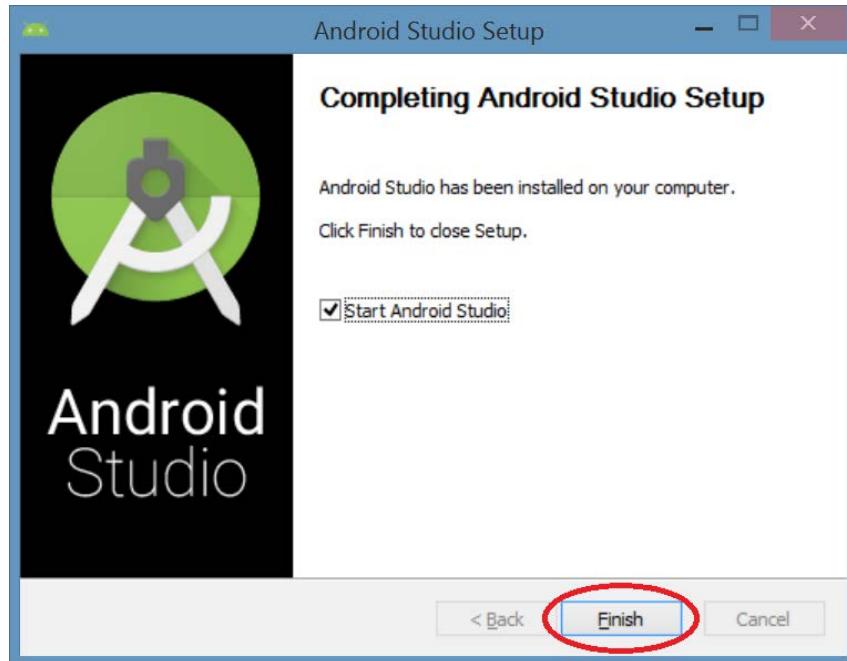




Step 16: When the installation is complete, select 'Next'.



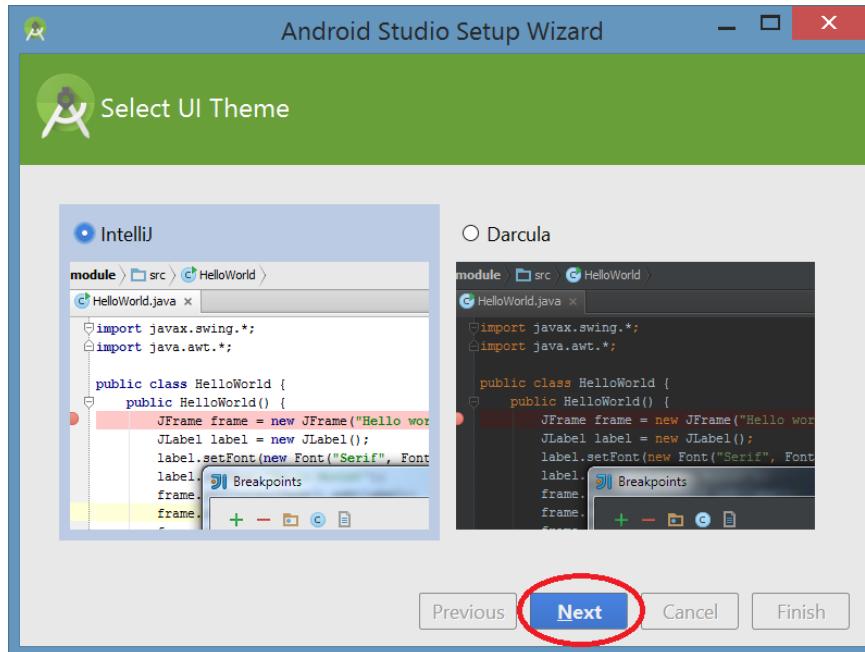
Step 17: Open Android Studio by selecting 'Finish'.



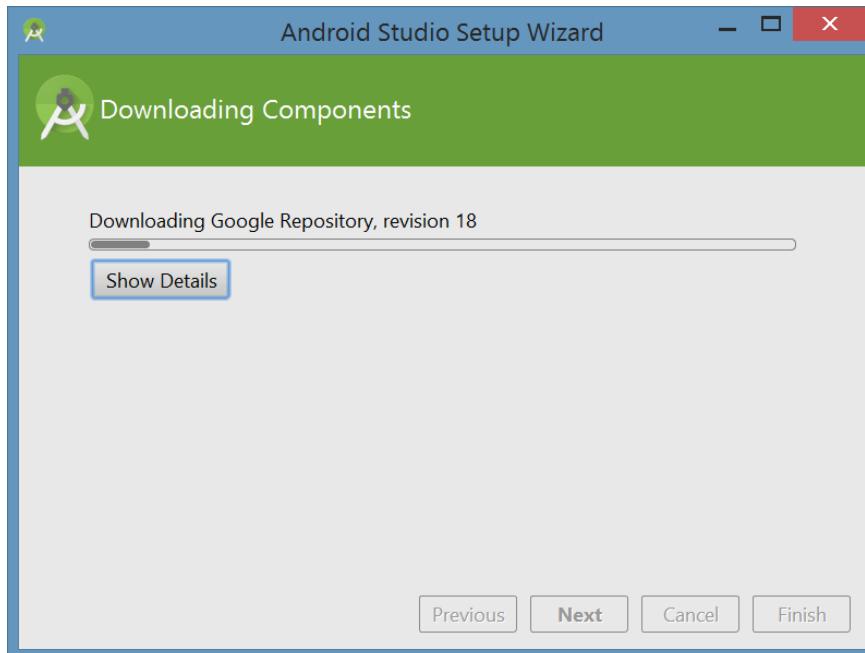
The following screen will be displayed while Android Studio is opening.

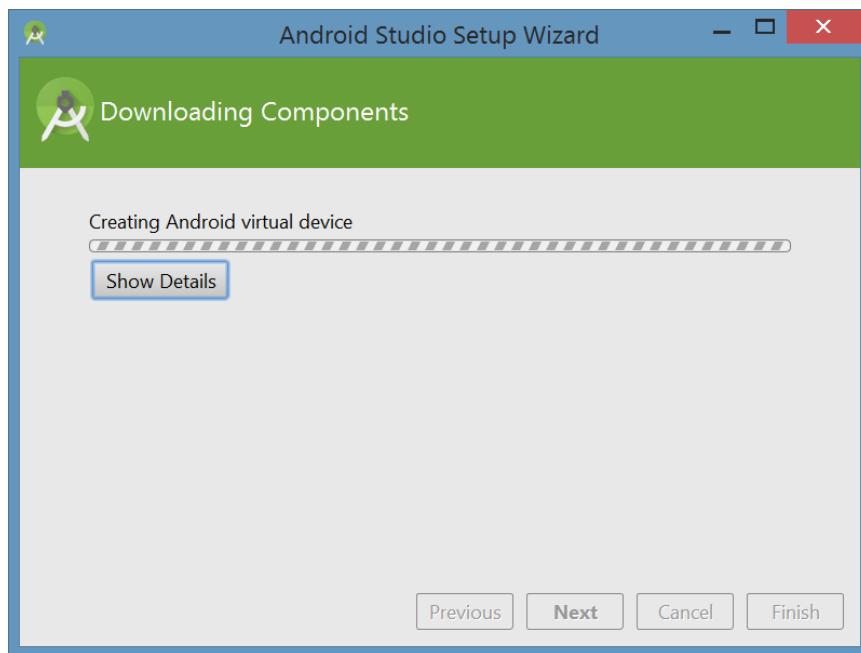


Step 18: Choose 'Next' to select the default UI Theme.

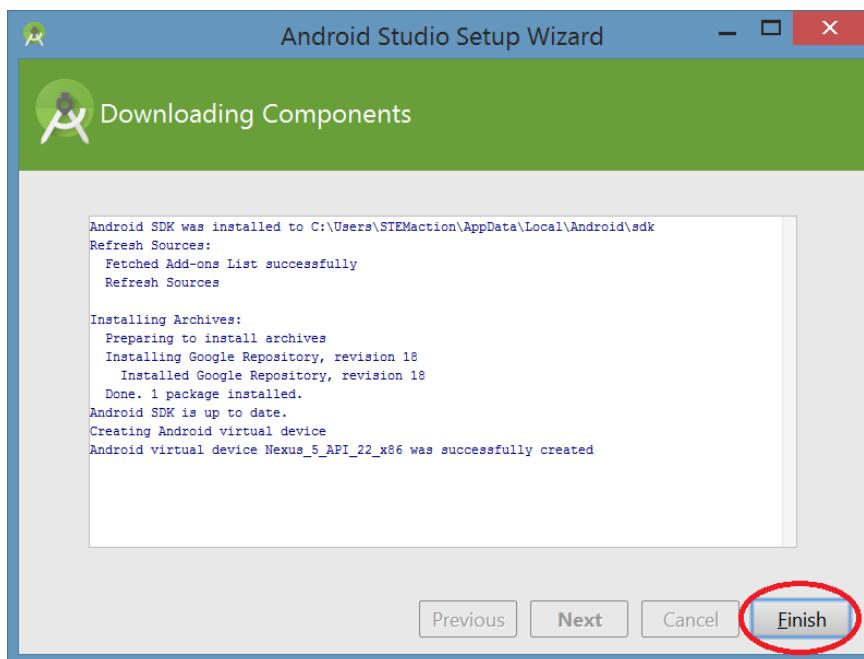


The following windows will be displayed while the setup wizard downloads necessary components.

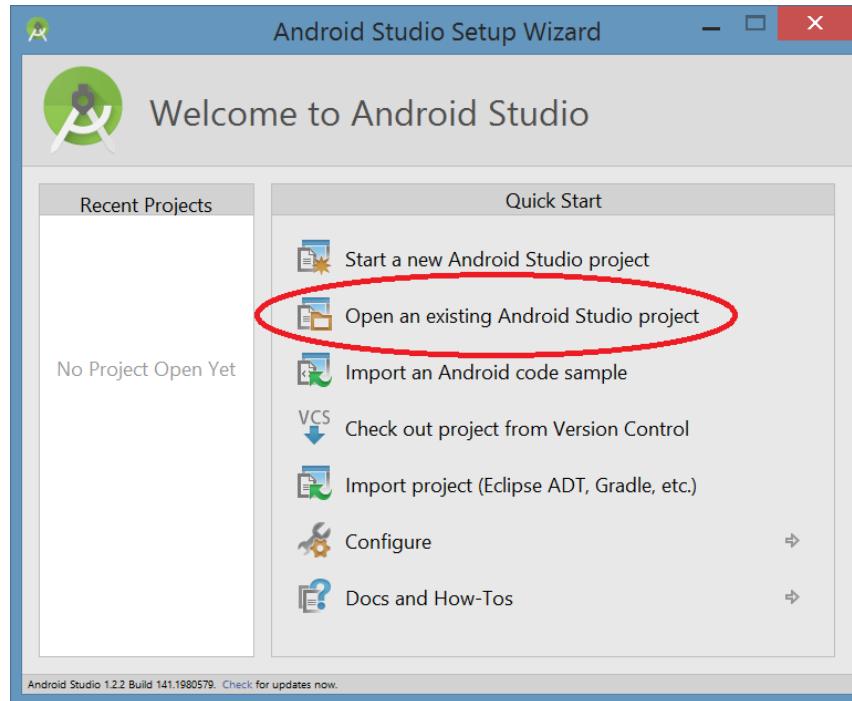




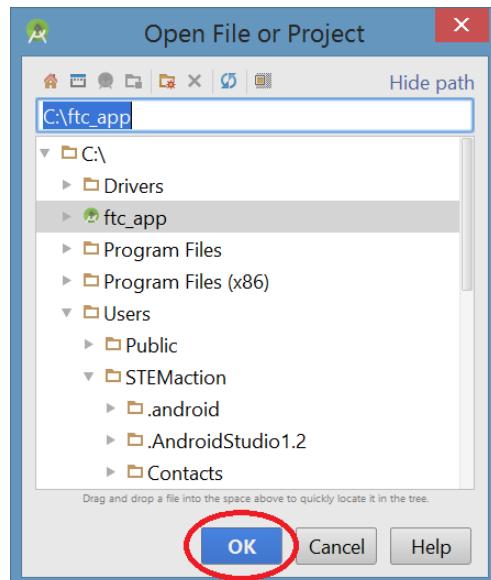
Step 19: Select 'Finish' and the Android Studio Welcome page will be shown.



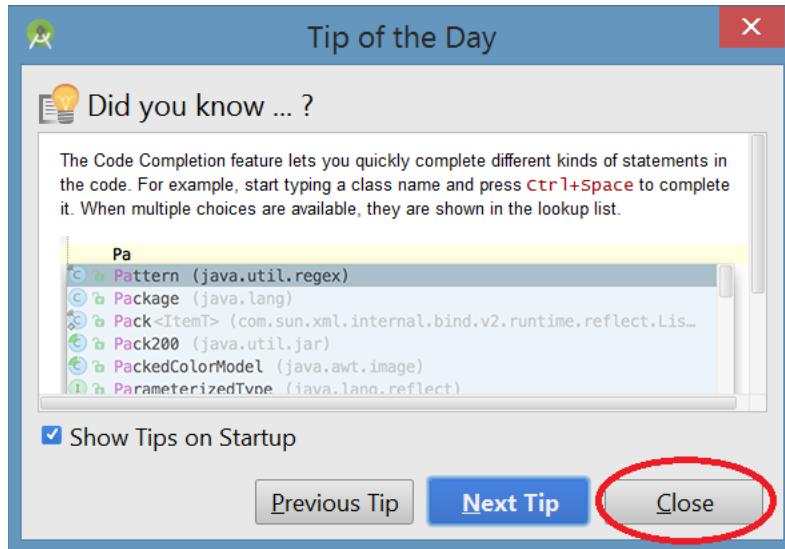
Step 20: Open the FTC SDK by selecting ‘Open an existing Android Studio project’.



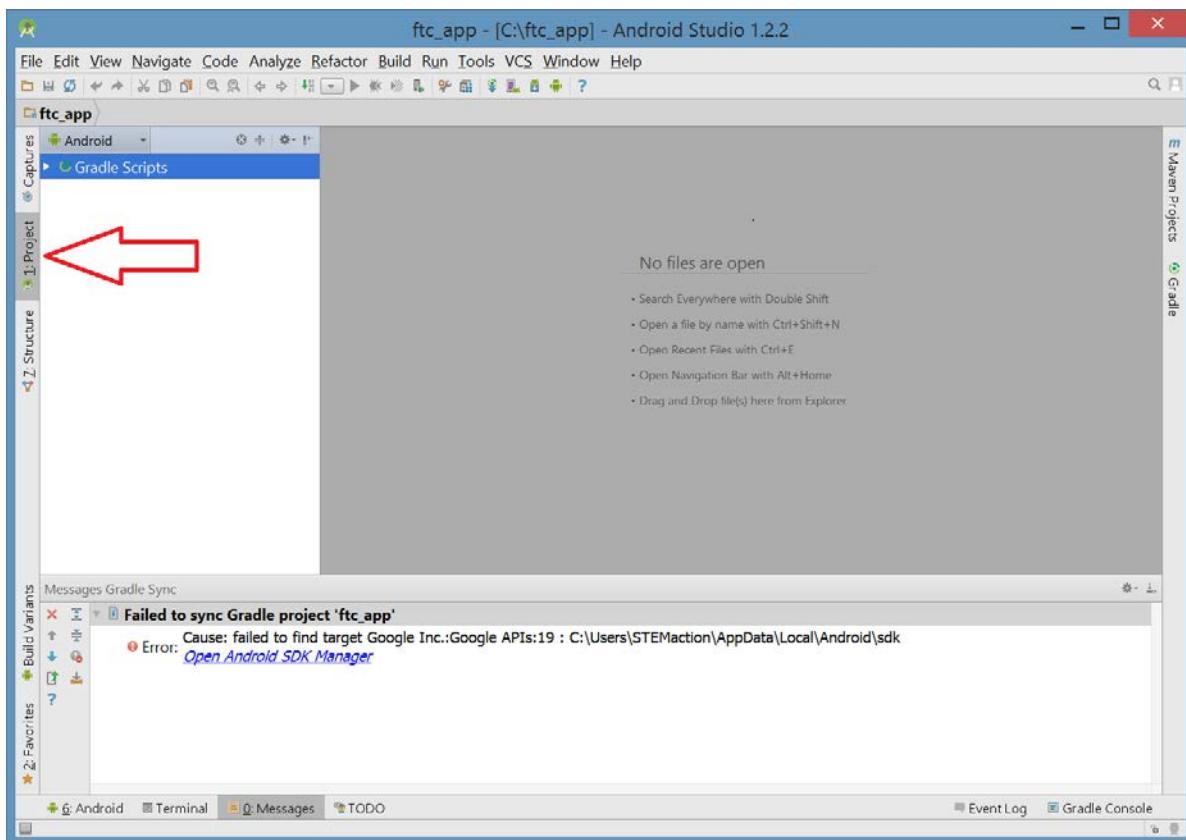
Step 21: It might take a while for the dialog to completely load. If ‘ftc\_app-master’ doesn’t show in the list after a long while, close the “c:\” folder by clicking the triangle to the left of ‘C:\’ and reopen the folder by clicking the triangle again. Single click the ‘ftc\_app-master’ folder. Select OK. Android Studio will attempt to build the project and several windows will be displayed before the IDE opens.



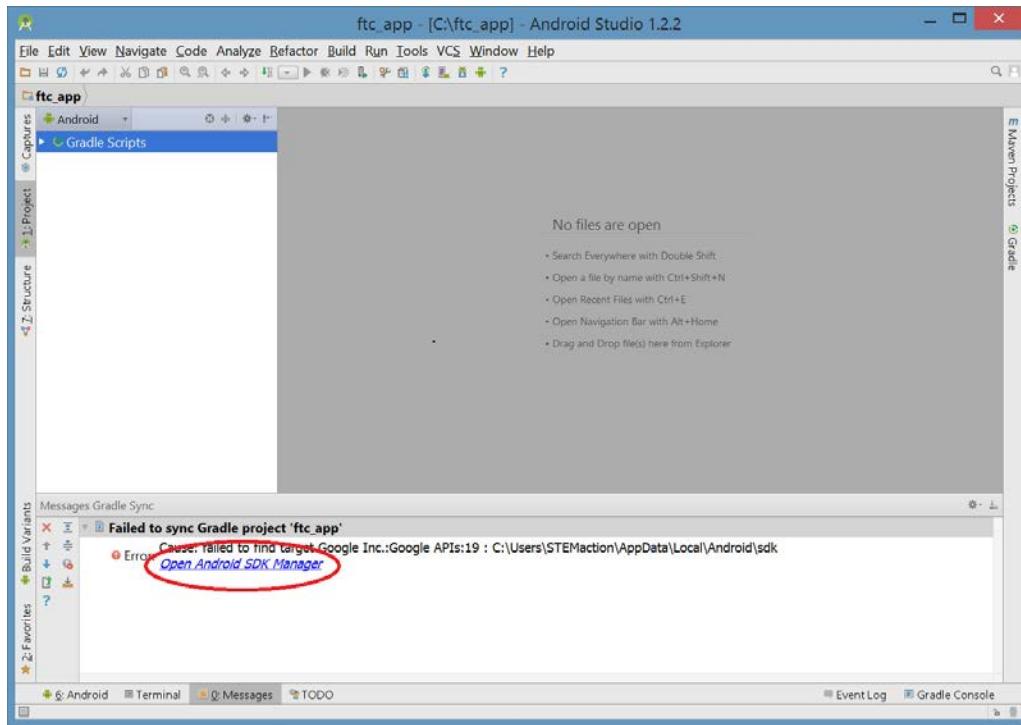
Step 22: Close the Tip of the Day window.



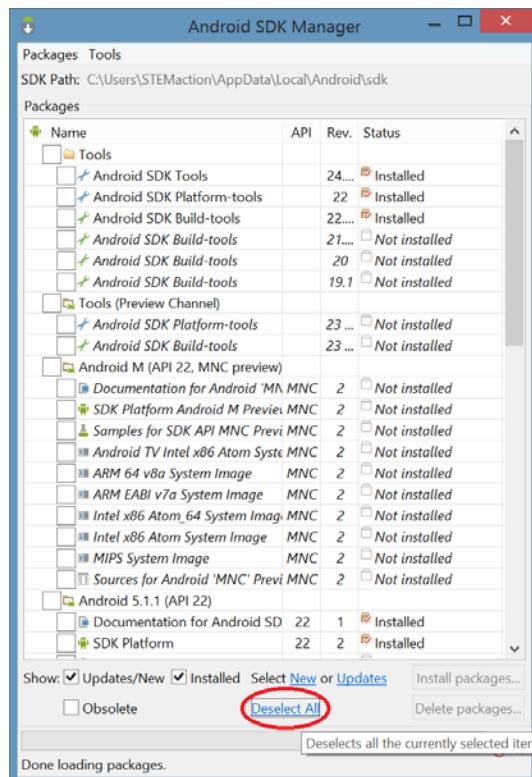
Step 23: If the ftc\_app-master project isn't shown, click on the triangle next to Android (near the top left portion and select Project from the drop down list. The image below is sometimes how the project opens (i.e. without the SDK showing).



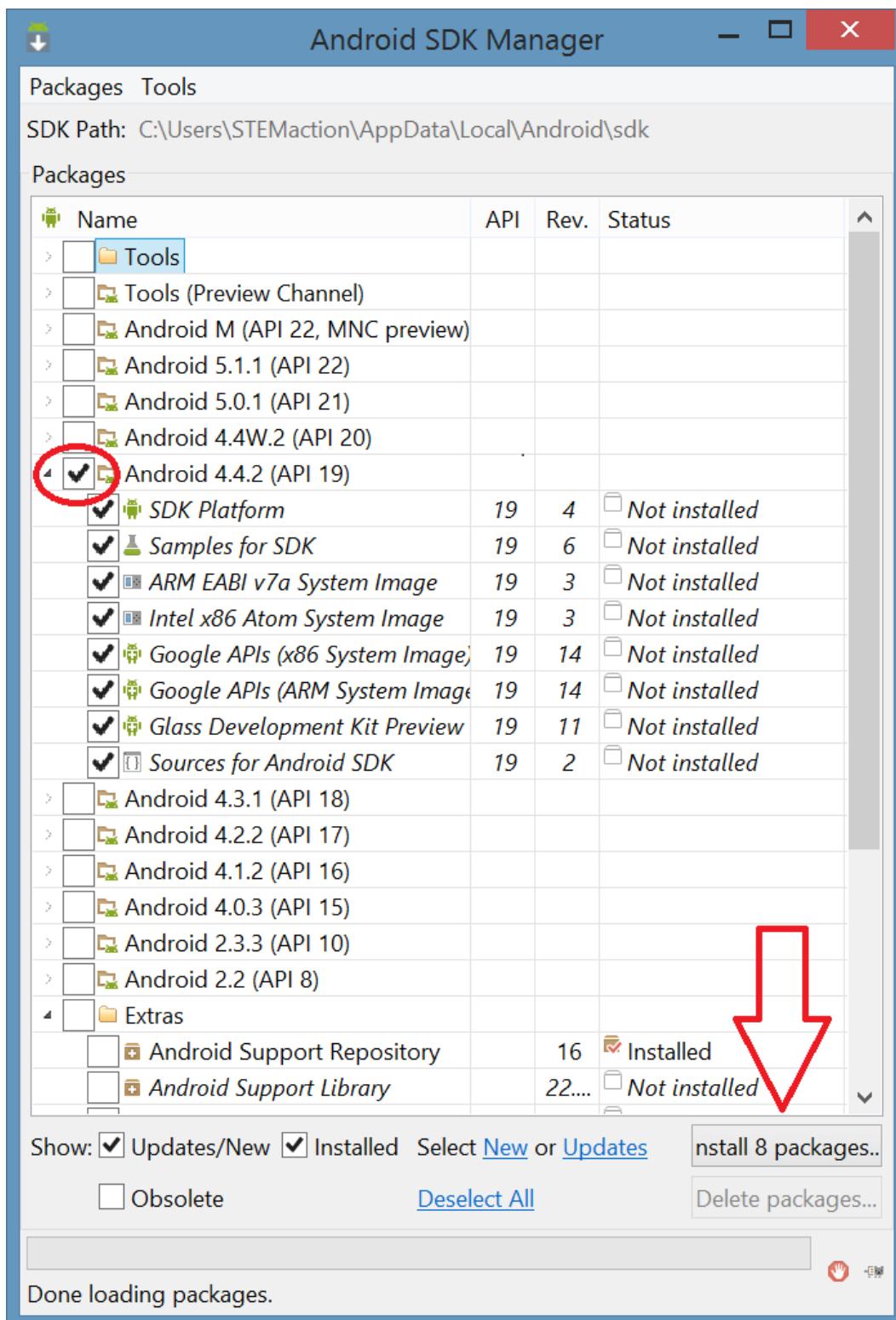
Step 24: If an error message is displayed at the bottom of the window, then install the Android SDK Manager by selecting the link near the bottom of the page. If there is no message that indicates “failed to find Build Tools...”, then the installation is complete and proceed to the next section “Installing the Push Bot Op modes”.



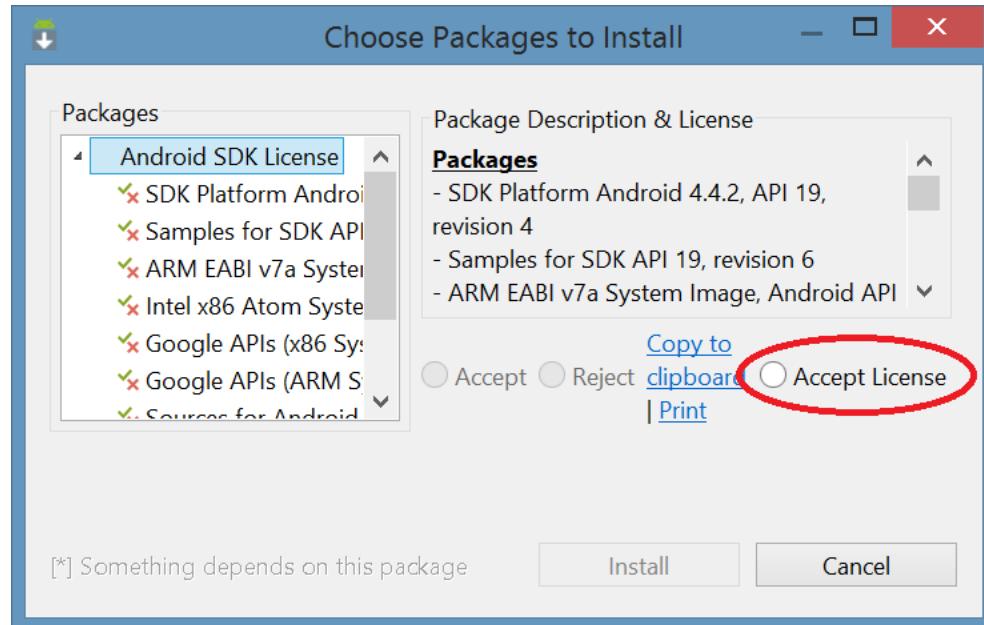
Step 25: Deselect all components using the ‘Deselect All’ button near the bottom of the window.



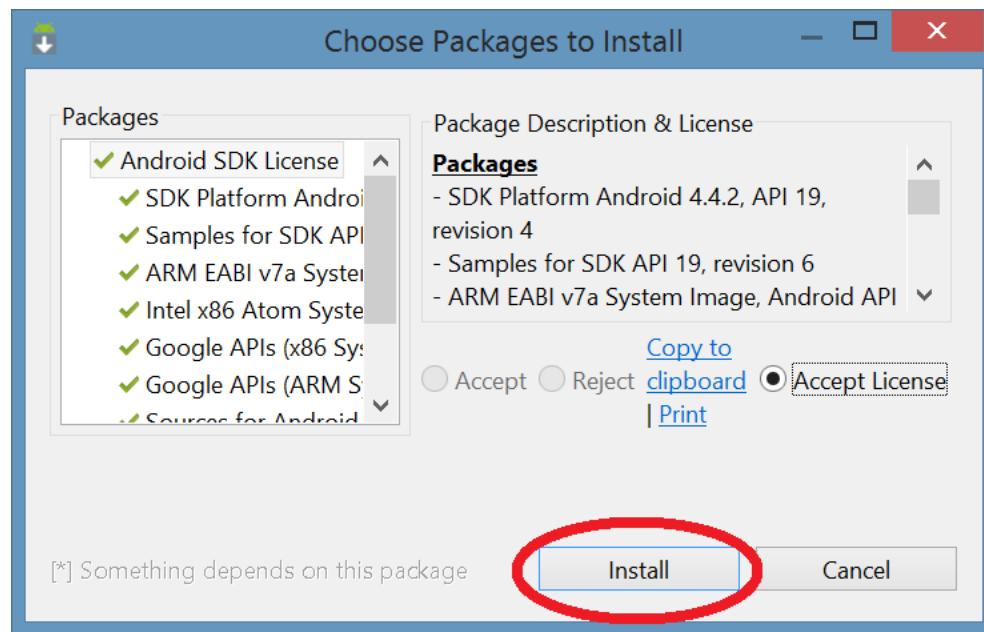
Step 26: Select the box (red circle) next to Android 4.4.2 (API 19). All components beneath that element will become checked. Install the components (red arrow).

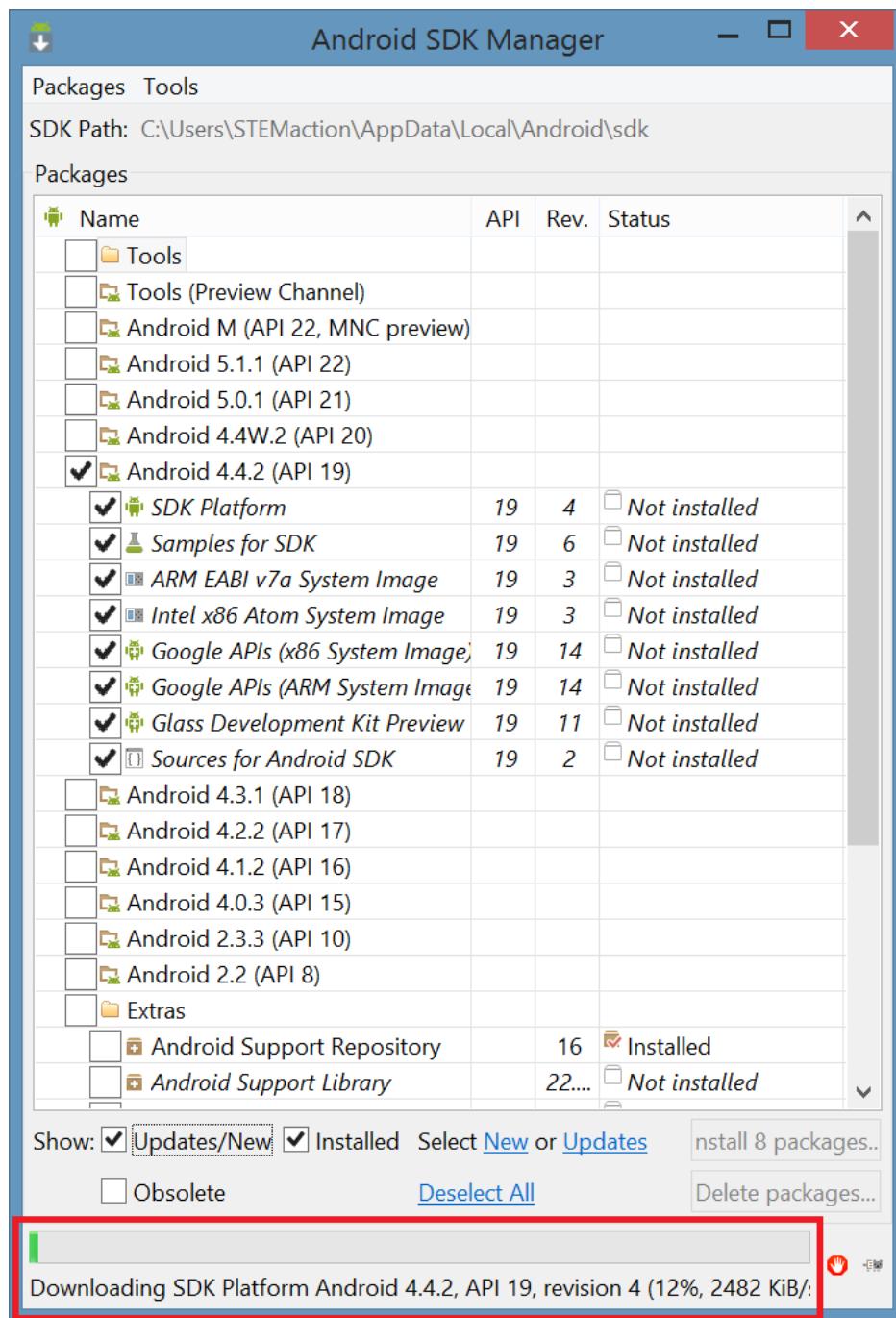


### Step 27: Accept the License Agreements.

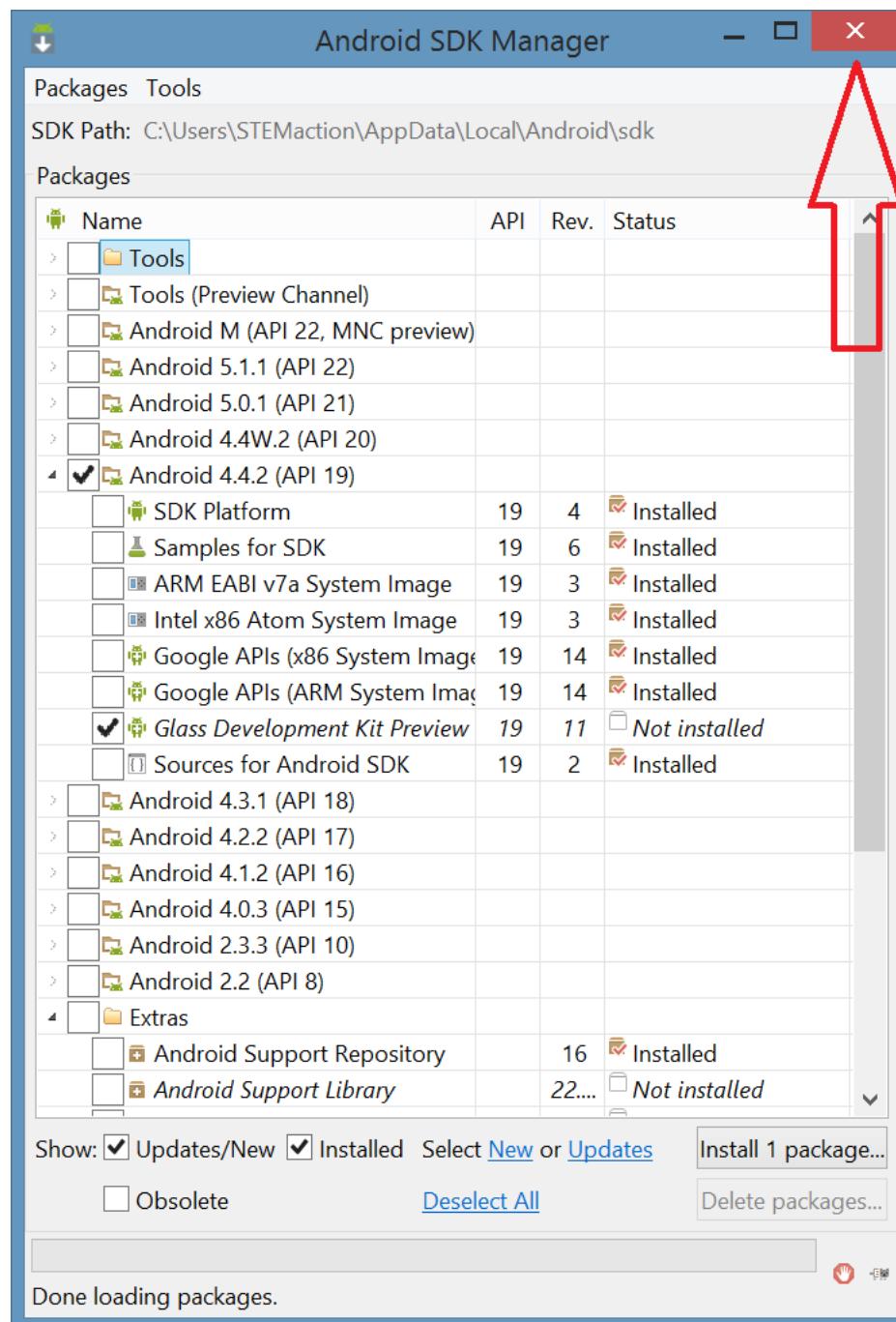


Step 28: Install the packages. While the packages are being downloaded and installed, the status bar at the bottom of the Android SDK Manager will update (image below the following image).

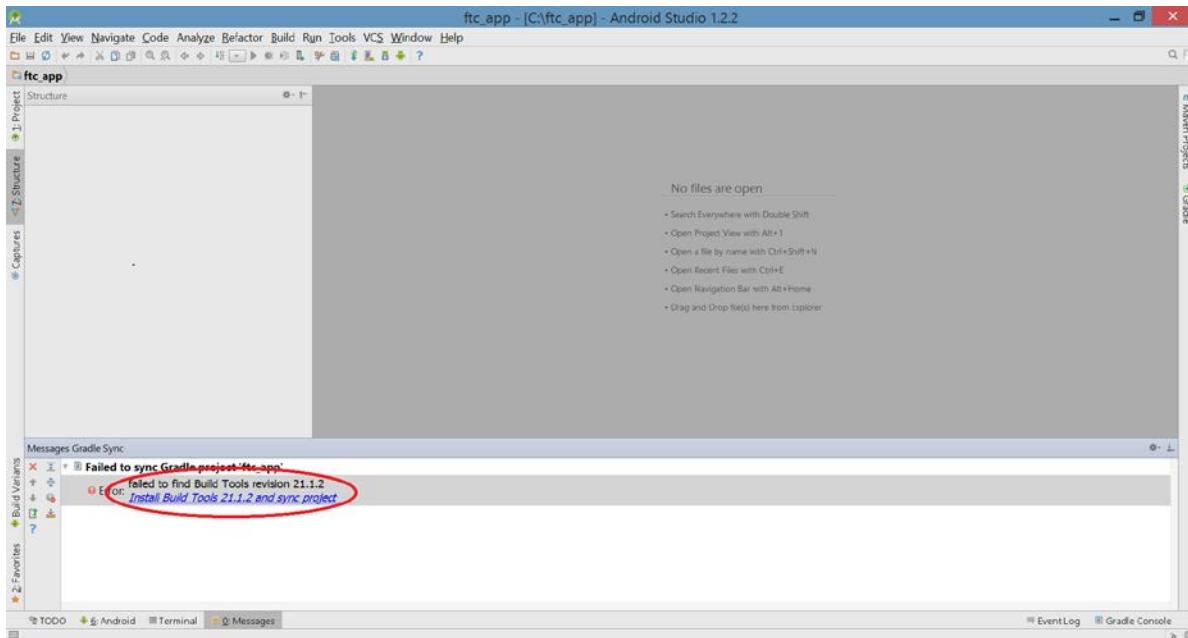




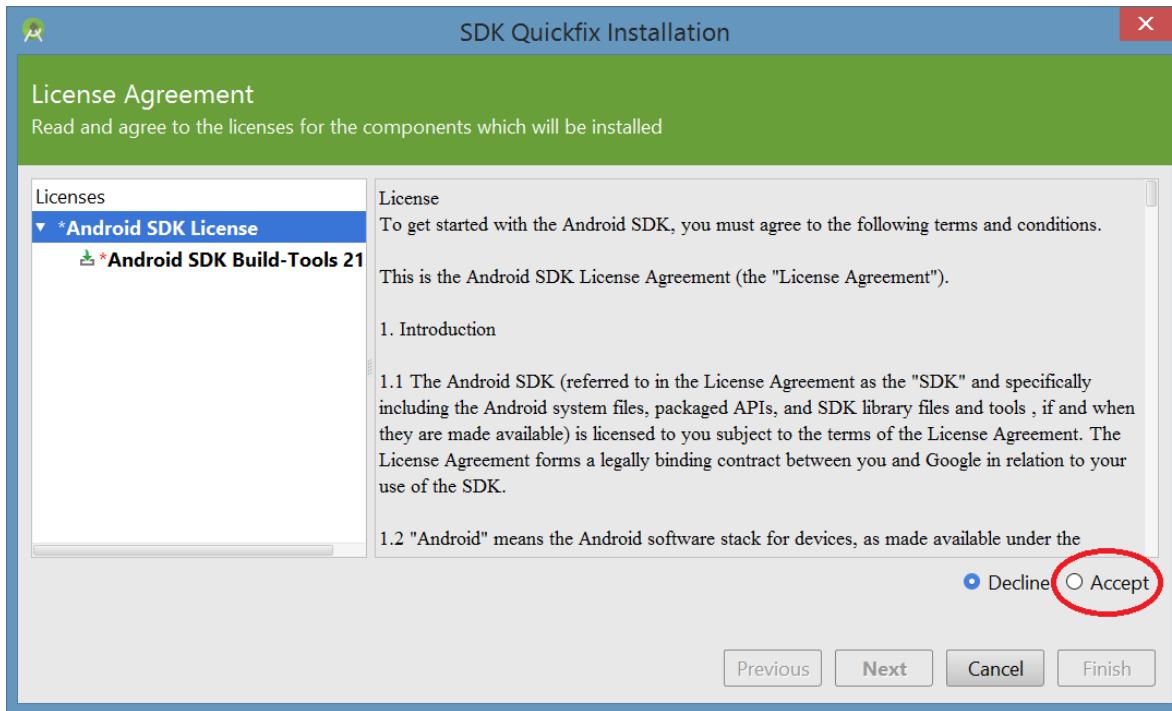
Step 29: After the download has completed, close the dialog.



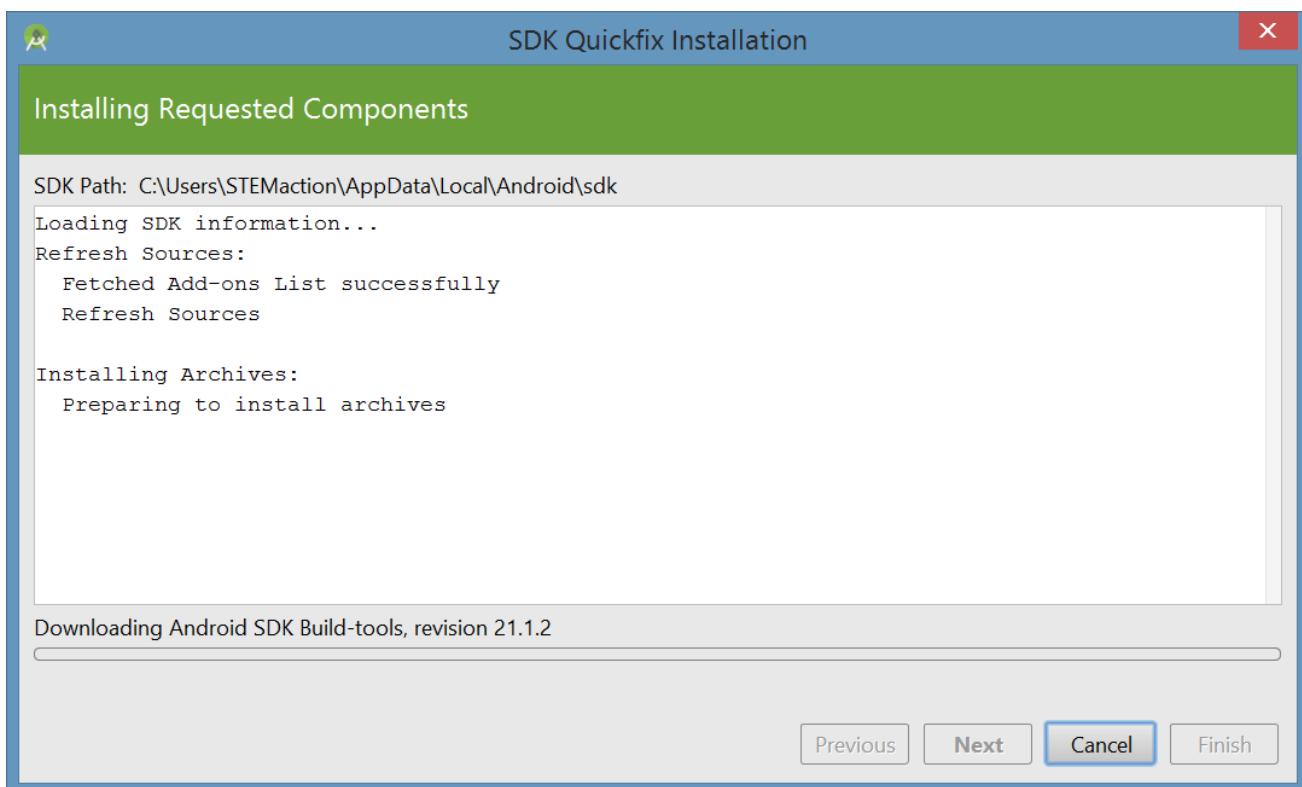
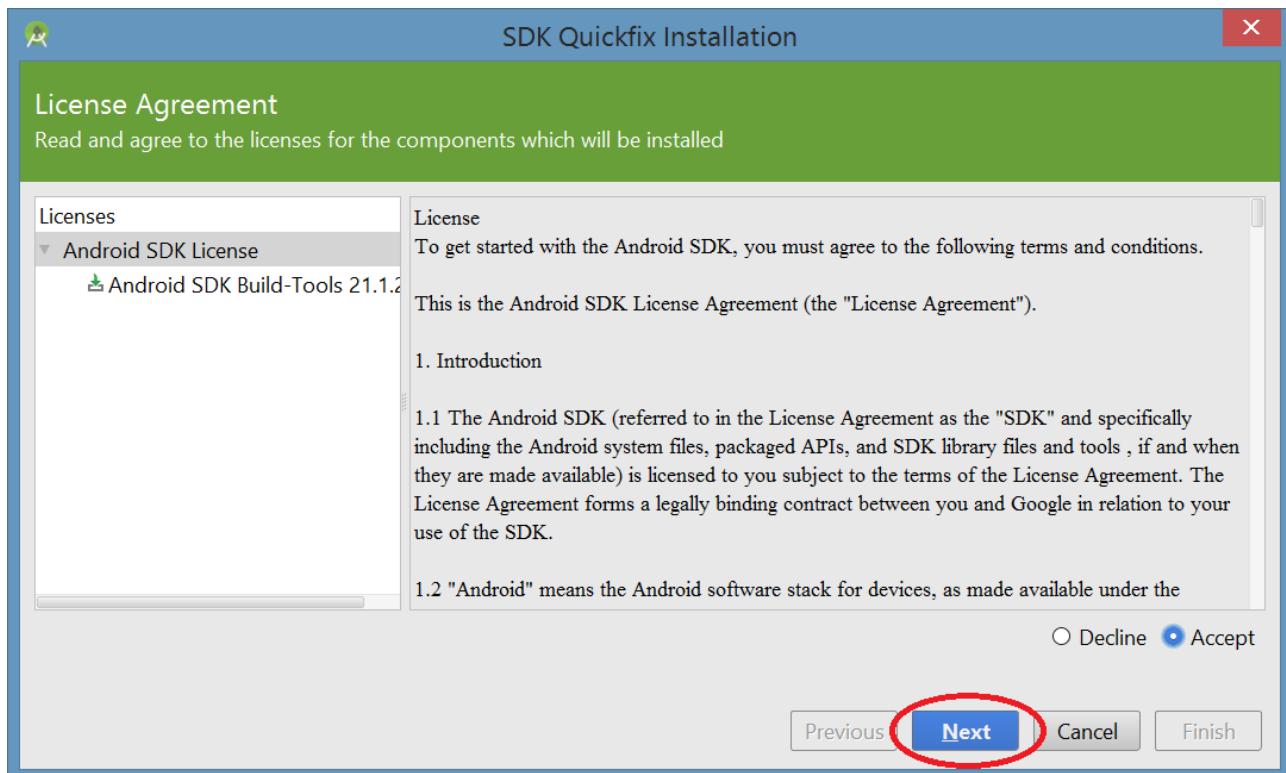
Step 30: If Android Studio doesn't automatically attempt a rebuild, close Android Studio and restart it. If a different error message than the previous error message, select the blue link again to install Build Tools 21.



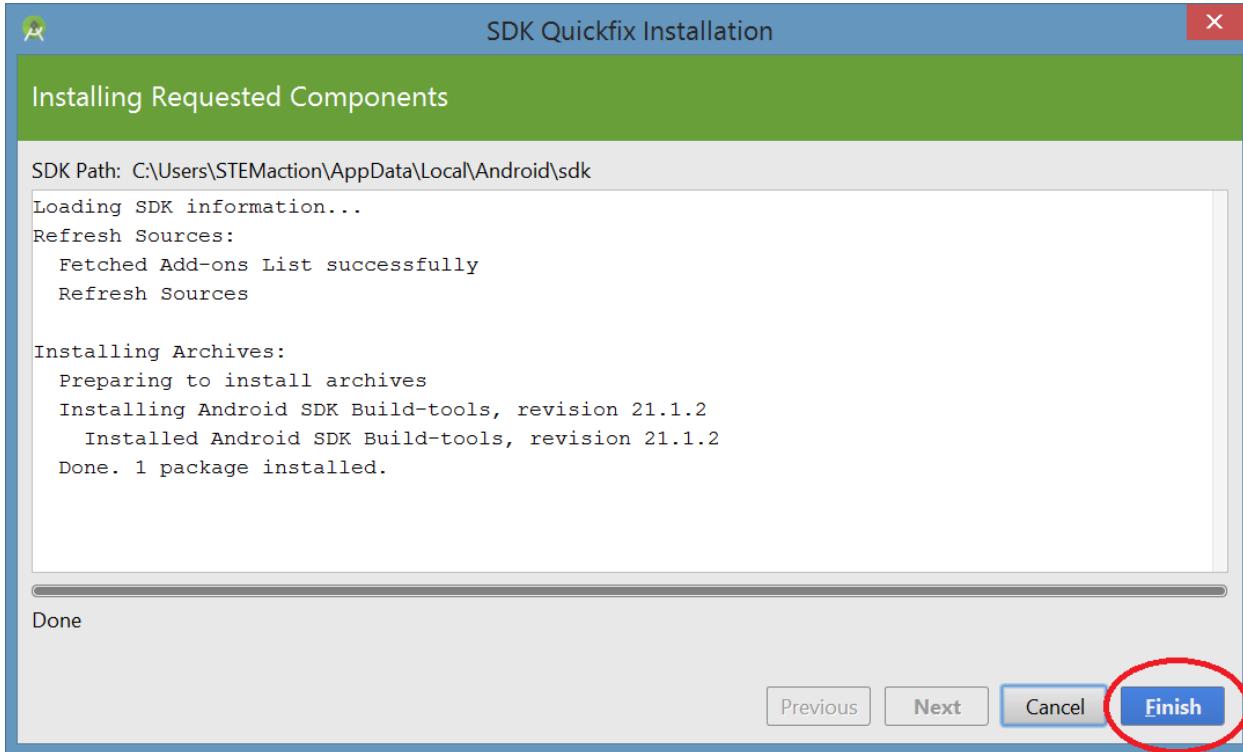
Step 31: Accept the quick fix license.



Step 32: Click 'Next' to install the components. The progress of the installation can be seen in the next dialog (shown below the following image).



Step 33: Return to Android Studio by selecting 'Finish'.

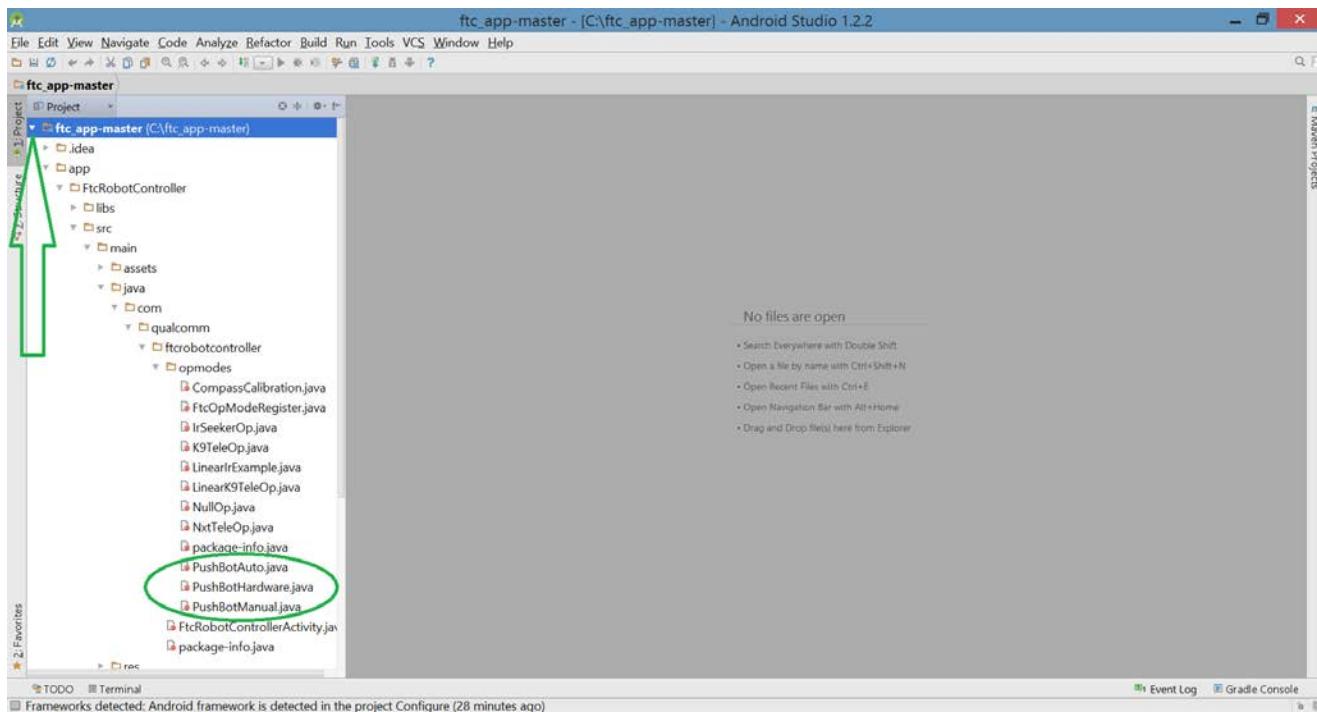


Step 34: Installation should be complete at this point. If an error message continues to be displayed, refer to the Android Studio website or USFIRST forum for more information.

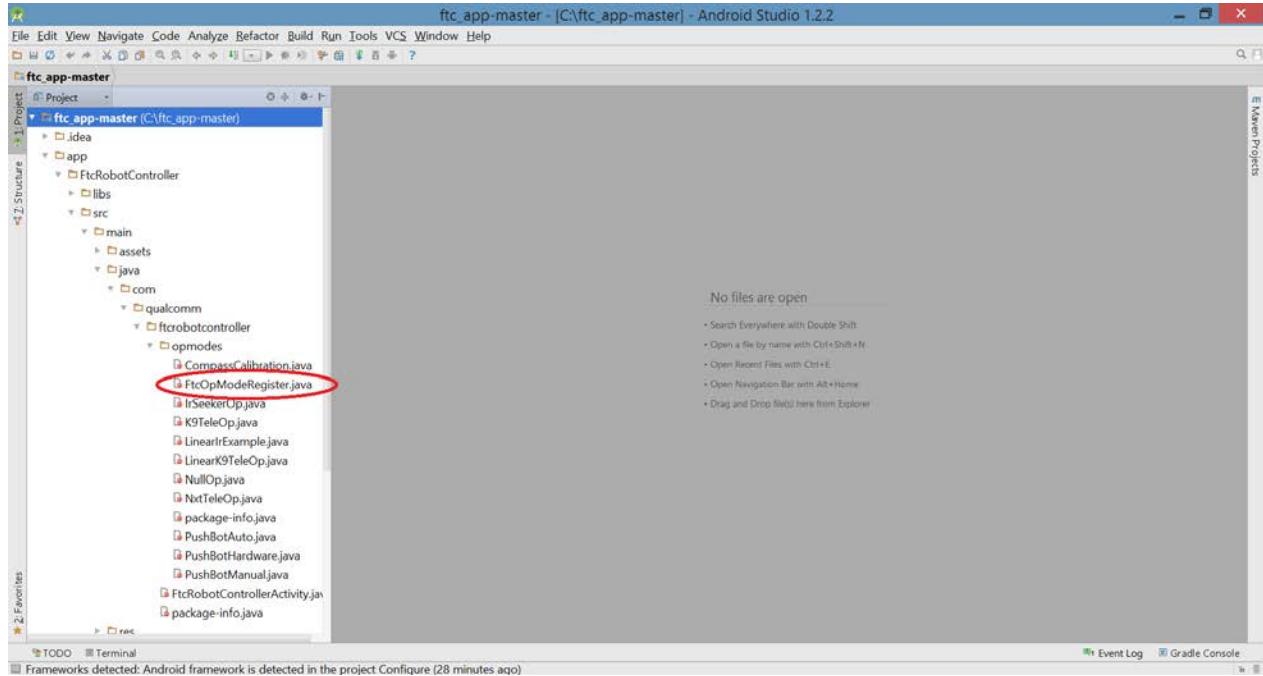
## Registering the PushBot Op modes

Before the Driver Station will display an Op mode, it must be registered (i.e. added to the FTC Robot Controller application's list of Op modes). You should verify that the PushBotAuto and PushBotManual Op modes are registered so that they will be listed as an available Op mode on the Driver Station.

Step 1: If Android Studio isn't still open, open it now. Open the FtcRobotController tree using the small arrow to its left. Continue opening the tree until the Op modes item is shown (i.e. open FtcRobotController, then Java, then com.qualcomm.ftcRobotcontroller, then Op modes). Verify that the PushBot Op modes are in the tree. Note that there are three entries in the tree that begin with 'PushBot': PushBotAuto, PushBotHardware and PushBotManual (green circle).



Step 2: Open the FtcOpModeRegister file (red oval) by double clicking it.



Step 3: Scroll to the bottom of the file until the ‘manager.register’ lines are visible; they will have green (strings) and blue (class names) text. Using the existing entries as a guide, type “**manager.register (“PushBotAuto”, PushBotAuto.class);**” and “**manager.register (“PushBotManual”, PushBotManual.class);**”. Do **NOT** create a line for the PushBotHardware class. The new lines are shown in the red rectangle.

```

ftc_app-master - [C:\ftc_app-master] - FtcOpModeRegister.java - Android Studio 1.2.2
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
ftc_app-master > app > FtcRobotController > src > main > java > com > qualcomm > ftcrobotcontroller > opmodes > FtcOpModeRegister.java

/*
 * Register Op Modes
 */
public class FtcOpModeRegister implements OpModeRegister {

    /**
     * The Op Mode Manager will call this method when it wants a list of all
     * available op modes. Add your op mode to the list to enable it.
     *
     * @param manager op mode manager
     */
    public void register(OpModeManager manager) {
        /*
         * register your op modes here.
         * The first parameter is the name of the op mode
         * The second parameter is the op mode class property
         *
         * If two or more op modes are registered with the same name, the app will display an error.
         */

        manager.register("K9TeleOp", K9TeleOp.class);
        manager.register("LinearK9TeleOp", LinearK9TeleOp.class);
        manager.register("LinearIRExample", LinearIRExample.class);
        manager.register("NullOp", NullOp.class);
        manager.register("IrSeekerOp", IrSeekerOp.class);
        manager.register("CompassCalibration", CompassCalibration.class);
        manager.register("NxtTeleOp", NxtTeleOp.class);
        manager.register("PushbotAuto", PushbotAuto.class);
manager.register("PushBotManual", PushBotManual.class);
    }
}

Event Log Gradle Console
65:56 LF: windows-1252: 

```

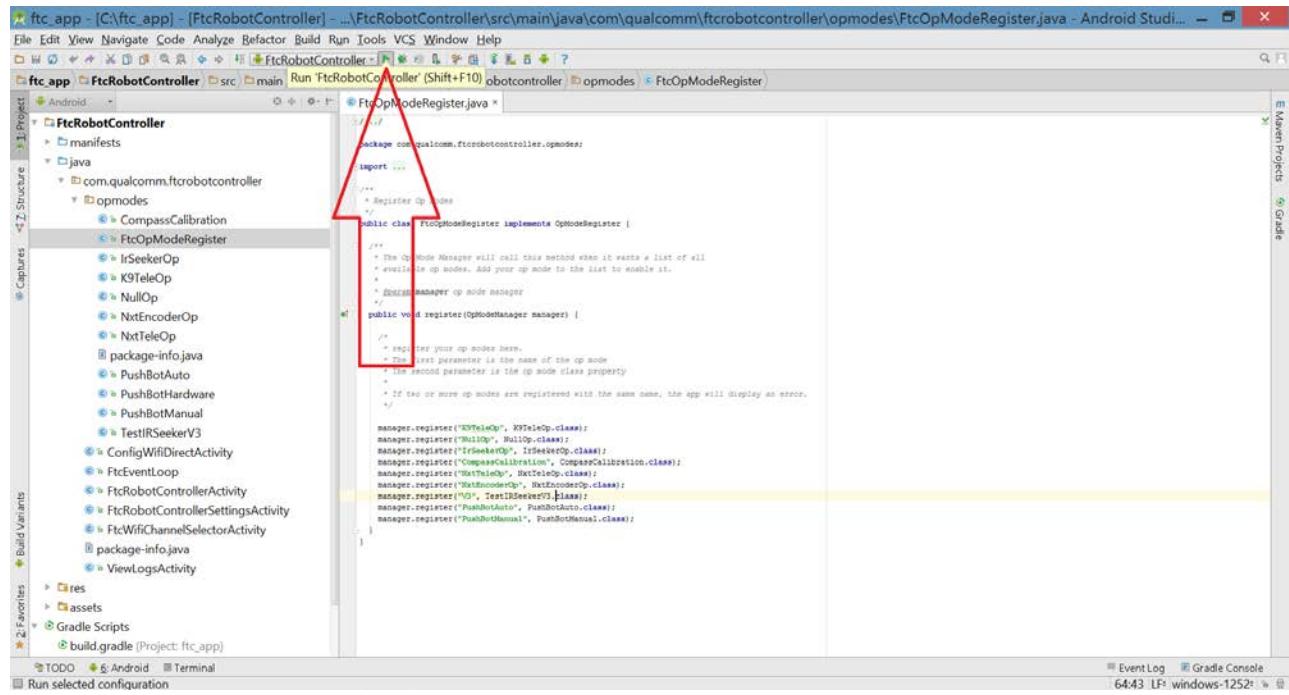
The screenshot shows the code editor for FtcOpModeRegister.java. The cursor is positioned at the end of the file, just before the closing brace. Two new lines of code have been added, both of which are highlighted with a red rectangle:

```

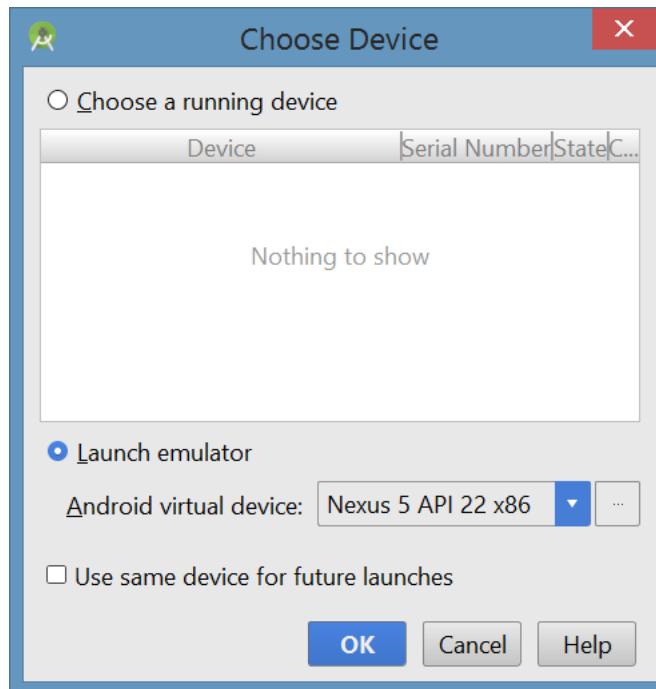
        manager.register("PushBotAuto", PushBotAuto.class);
manager.register("PushBotManual", PushBotManual.class);

```

Step 4 Build by selecting the green triangular button near the top of the window (red arrow). Note the results of the previous step (green oval). This may take a while. Status is displayed at the bottom of the window.



Step 6: Observe that the build completed successfully and the Choose Device window opens.



### **Preparing the Robot Controller and Driver Station cell phones**

To prevent the phones from attempting a connection with a mobile network, the SIM cards should be removed from both the driver station phone and the Robot controller phone.

Step 1: Gather both phones and both cases. If the phones are on, then turn them off. The power button is on the top right edge of the phone.



Step 2: Remove the back covers from the phones, leaving the phones face down. There is a gap between the front and back covers located on the lower right hand corner of the device that can be used to remove the back cover. Inserting a fingernail in the gap and then sliding the fingernail all around the case will cause the case to pop apart.



Step 3: The batteries are on the left and the SIM cards (circled) on the right. Gently push the SIM card INTO the phone.



Step 4: Let the spring gently push the SIM cards from the phones. Pushing the SIM cards activate the spring mechanisms; releasing pressure allows the springs to push the SIM cards out of the phones.



Step 5: Store the SIM card in a safe place. In the image below, the cards have been taped to the inside back cover of the phones.



Step 6: Replace the back covers.



Step 7: Place the protective cases' back covers onto the back of the phones.



Step 8: Turns phones face up.

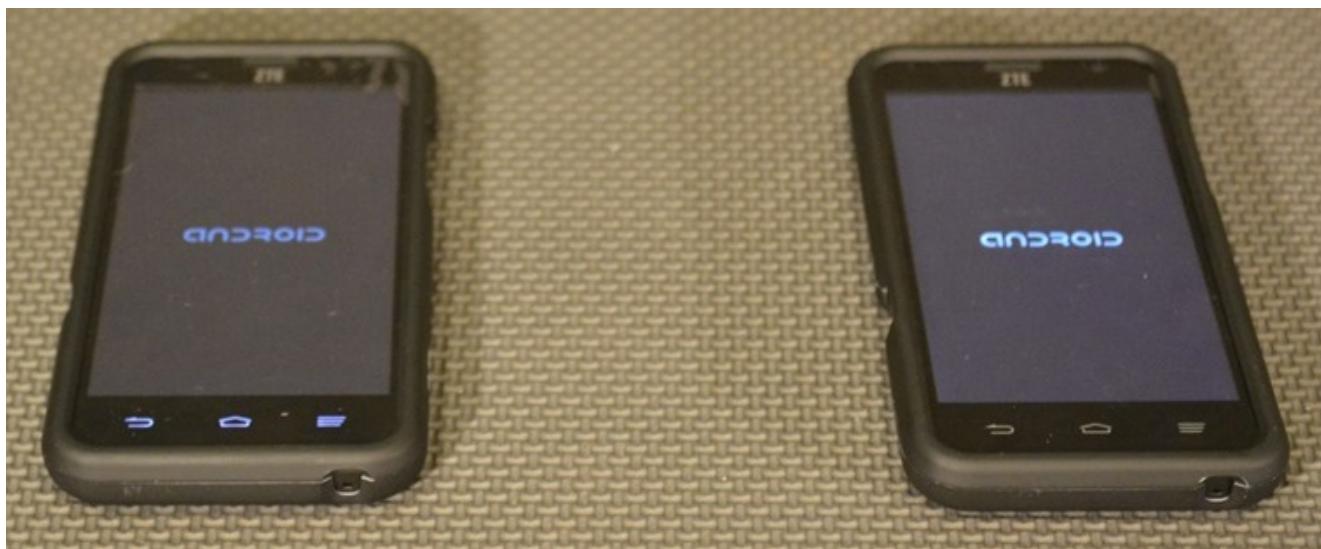


Step 9: Place the protective case front covers into the front of the phones.

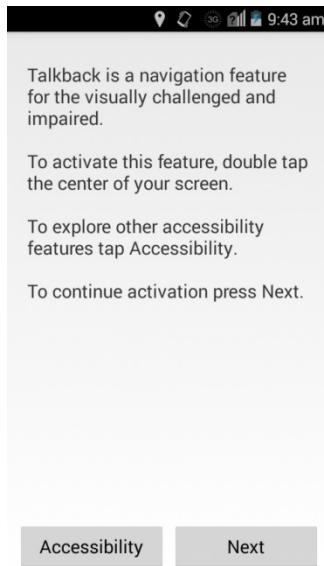


Step 10: Turn phones on. The circle is where the power switch is located. The phones will begin booting. The sequence of images shows the booting sequence.

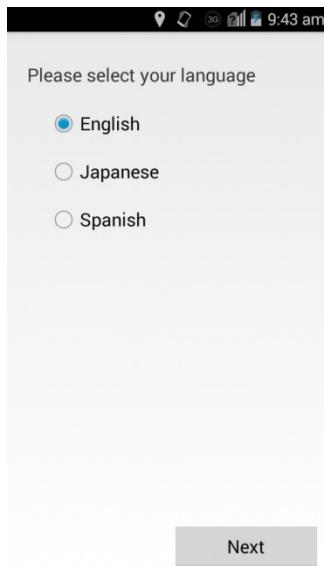




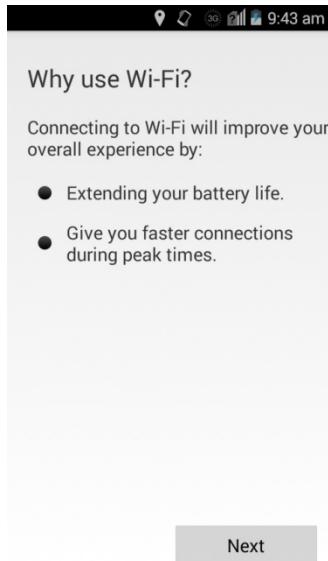
Step 11: The above images show instructions for both phones. This and all following steps may only show one image, but apply the step to both phones. Choose ‘Next’ to accept the default talkback setting.



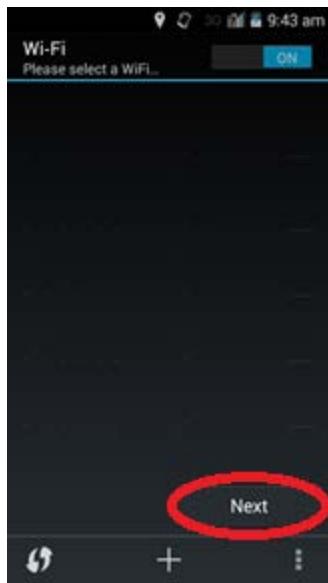
Step 12: Select ‘English’.



Step 13: Choose 'Next' to configure Wi-Fi.



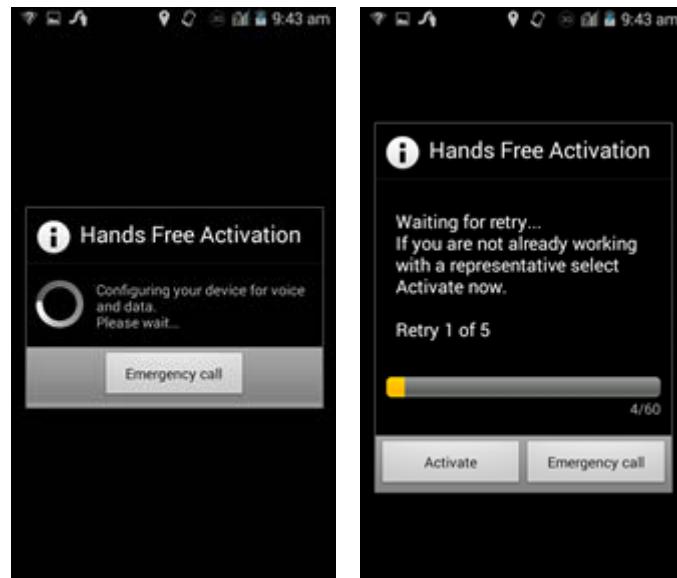
Step 14: Do not connect to Wi-Fi at this time. Select 'Next'.



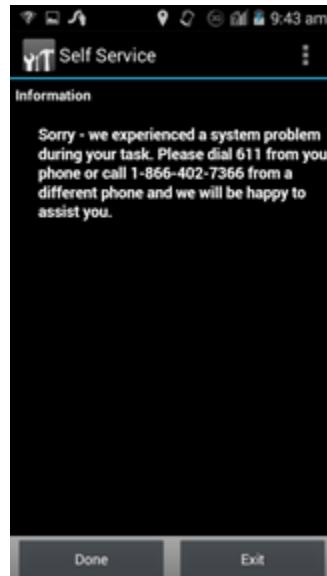
Step 15: When the UICC Error screen is shown, select 'OK'.



Step 16: Select 'Activate' when the Hands Free Activation activity provides the option. When the activity first starts, it provides the ability to make an emergency call, but then will provide an 'Activate' button.



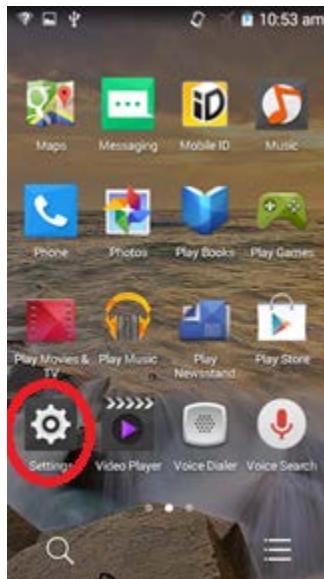
Step 17: The activation should fail. Select ‘Done’.



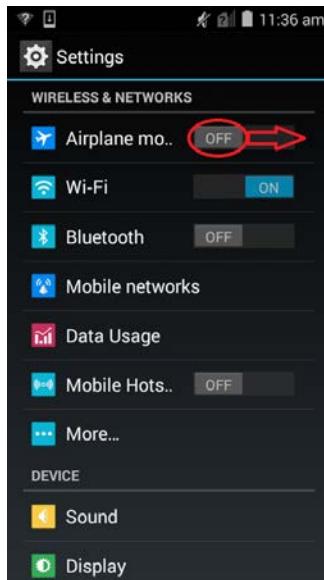
Step 18: Android will return to the home screen. Open the application activity by selecting its icon (shown in the red circle).



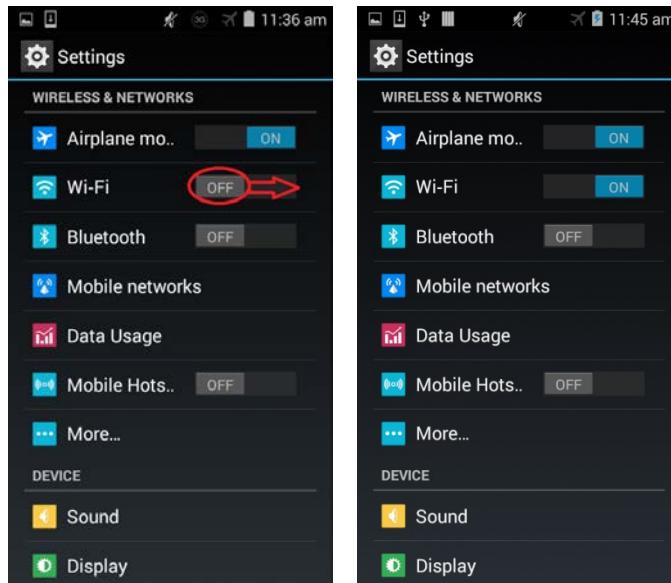
Step 19: Open the settings activity by selecting its icon (shown in the red circle). It might be necessary to swipe the screen left or right.



Step 20: Put the phones into Airplane Mode by sliding the button from left to right until the icon comes into view.



Step 21: Turn Wi-Fi on, if it's off, by sliding the button from left to right. The phones' screens should appear as the image on the right.



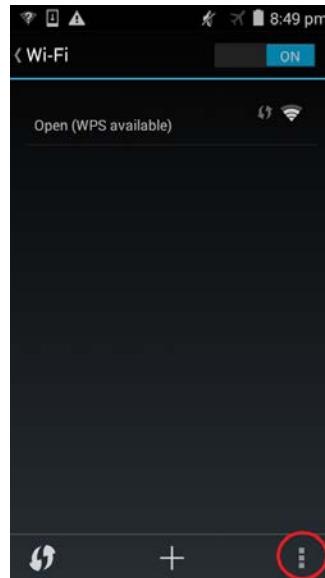
### Naming the Cell Phone

According to the Robot software rules in the game manual, the phones need to be named properly. The first portion of the name will be the Team's number. The Driver Station phone's name will be postfixed with “-DS”. The Robot Controller's will be postfixed with “-RC”.

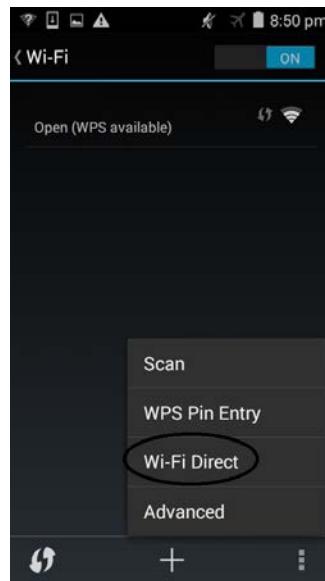
Step 1: (Continuing from the previous section – the settings page is displayed) Select Wi-Fi.



Step 2: Select the overflow settings (three dots on the bottom right).



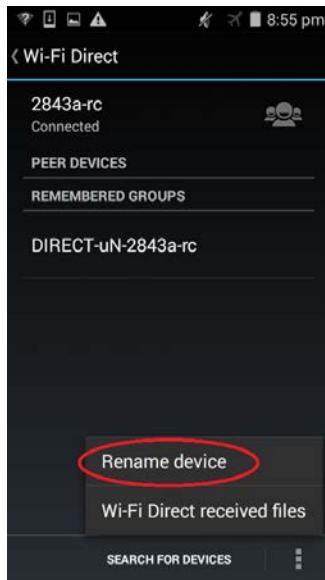
Step 3: Select Wi-Fi Direct



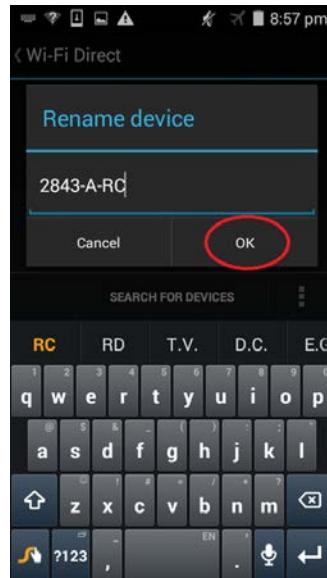
Step 4: If “RENAME DEVICE” is not shown at the bottom of the screen, then select the overflow settings and then select “Rename device”. In the image below, the phone had already been named. New phones will have a generic name.



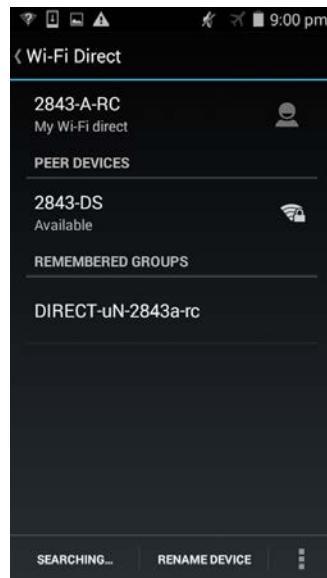
Step 5: Enter the new name according to the rules in Game Manual Part I.



Step 6: Select OK.



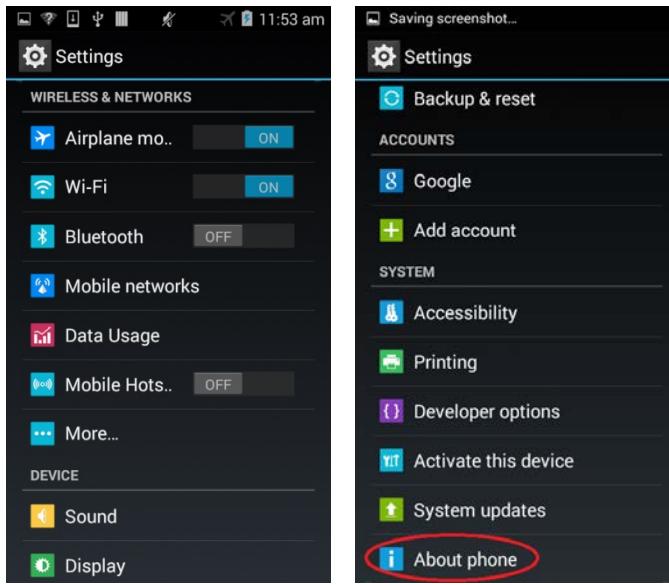
Step 7: From the page shown below, return to the settings activity by pressing the back button. It may take several presses to return to the settings activity.



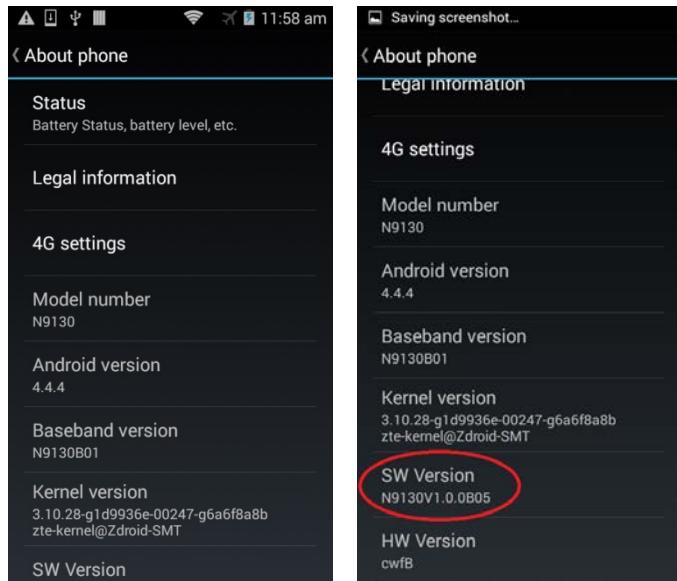
### **Enabling Debugging Mode on the Robot Controller Cell Phone**

For this section choose one phone to be the Driver Station and the other to be the Robot Controller. The Robot Controller will host the Robot Controller app, which implements the PushBot Op modes. Android Studio can't deploy the Robot Controller app until debugging is enabled on the phone.

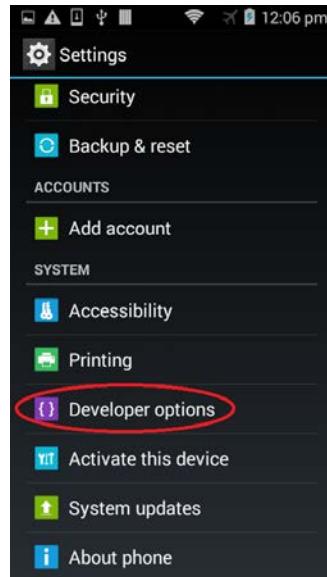
Step 1: From the previous section, the Settings activity should be shown (left image). Scroll downward until 'About phone.' Is visible (right image) – it should be at the very bottom. Select 'About phone'.



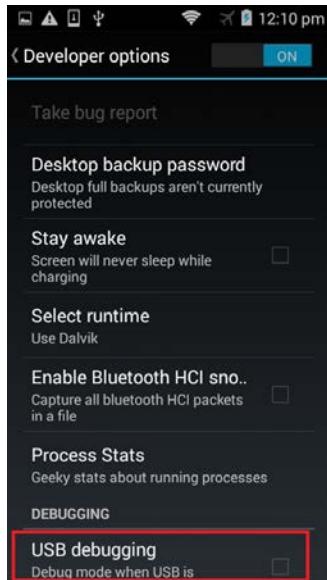
Step 2: The About phone activity (left screen) allows a developer to enable debugging. Scroll downward until ‘SW Version’ is shown (right image) - on some Android devices the text says ‘Build number’. Tap ‘SW Version’ 7 times. A toast (a message that is displayed for a short amount of time) should display as the number of taps increase and finally should indicate that debugging is enabled.



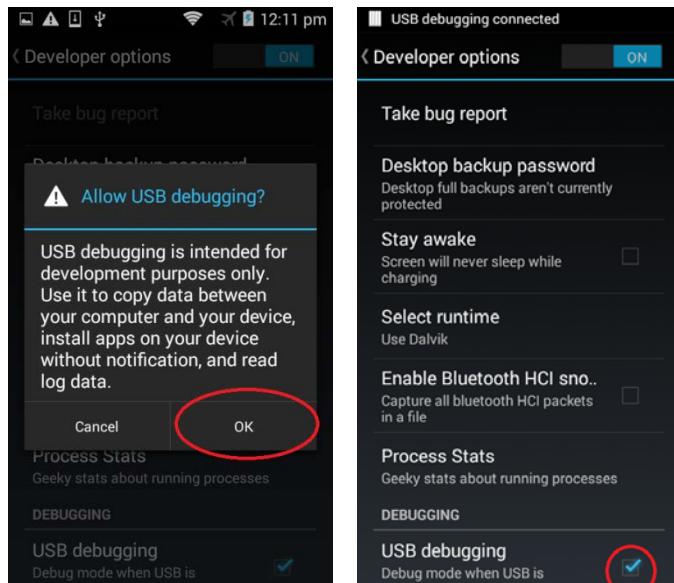
Step 3: If Android hasn't already switched to the Settings activity, then do so now. Scroll to the bottom of the page and select ‘Developer options’. The developer options button is invisible until debugging is enabled (see previous step).



Step 4: Select the USB debugging checkbox.



Step 5: Select 'OK' to allow USB debugging between the phone and a computer when the phone is connected via USB to the computer hosting Android Studio. Note that the box has a check in it (right image).



### **Installing the USB Device Driver**

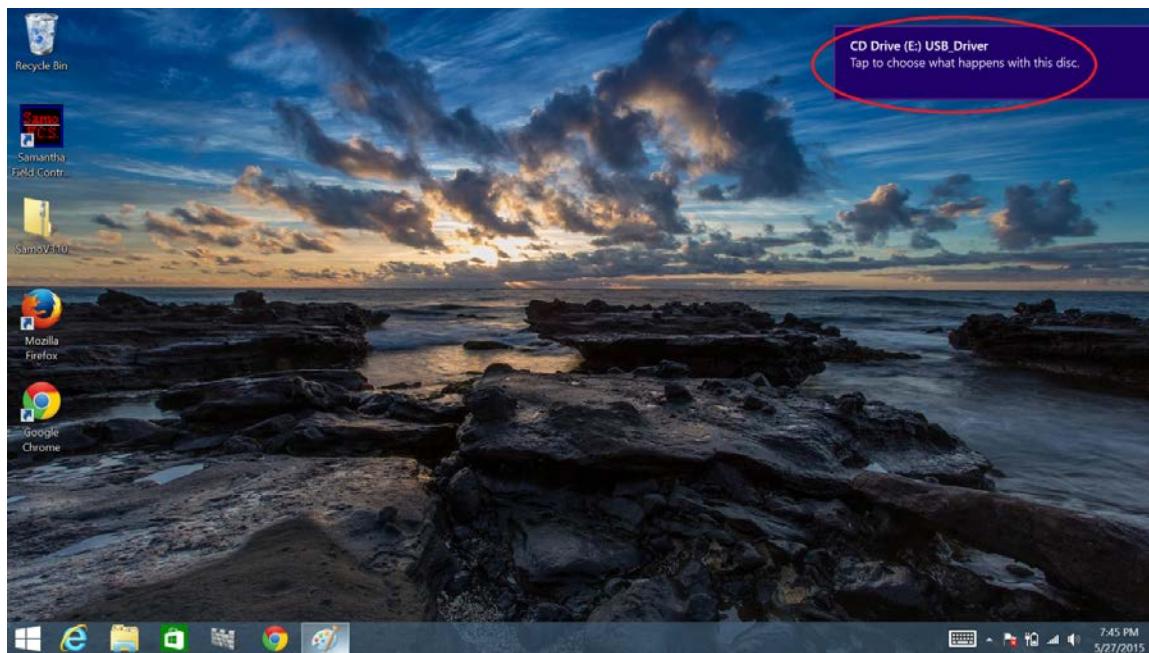
This section applies only to the Robot Controller cell phone. The correct USB driver will need to be installed onto the Windows computer hosting Android Studio. Without the correct driver, Android Studio can't connect to the cell phone when it is connected via a USB cable. Also the computer will treat the device as a media player or storage device (like a FLASH drive). File Explorer (aka Windows Explorer) might be able to access files on the cell phone in this mode, but Android Studio will not recognize the device as an available Android device.

Step 1: Using the cell phone's power cable (with the charging block removed so one end is a USB connector), plug the USB connector into a USB port on the phone.

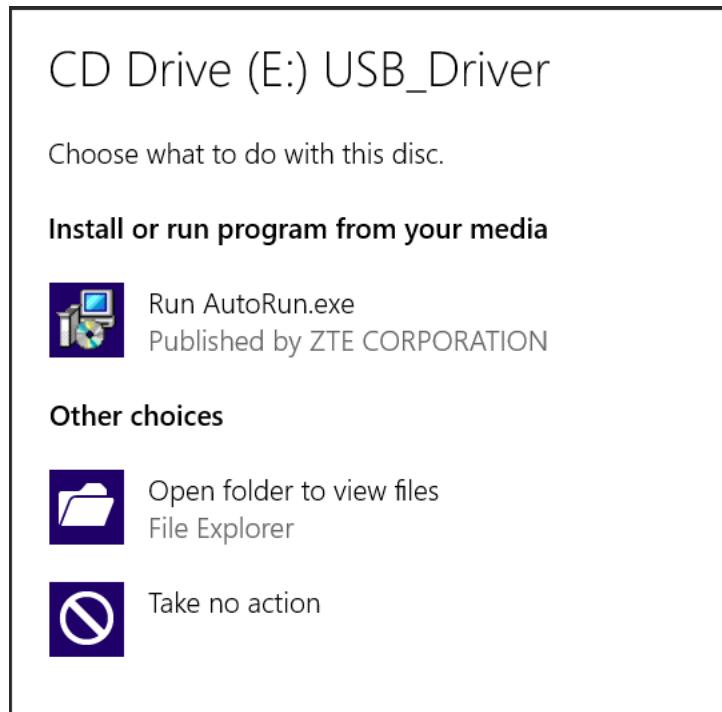
Step 2: Select 'Install driver' on phone.



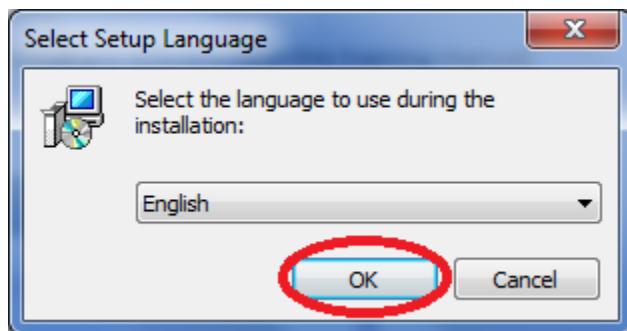
Step 3: When 'Install driver' is selected on the phone the computer will detect the connection. Confirm the computer's prompt.



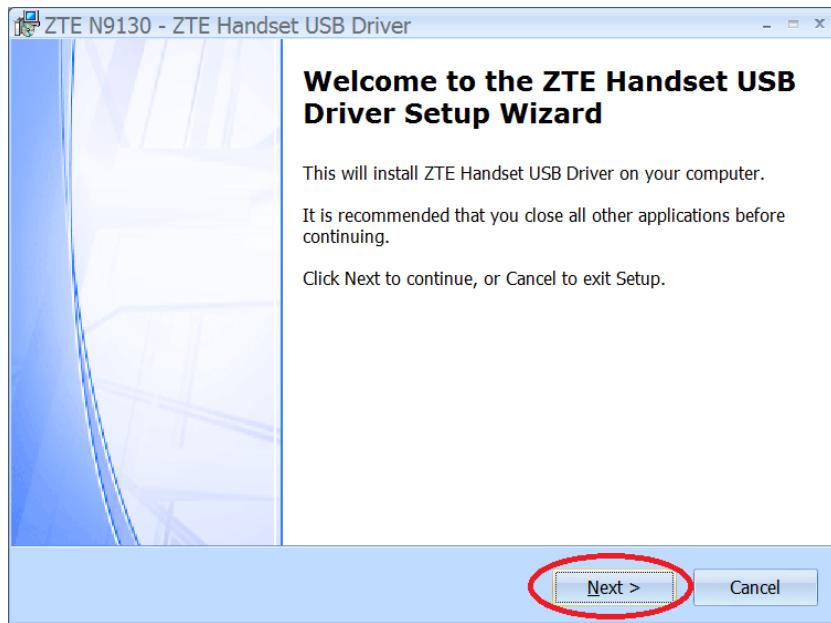
Step 4: After confirming the prompt, another prompt will be displayed. Select ‘Run AutoRun.exe’. If the screen darkens and a dialog asks “Do you want to allow the following program to make changes to this computer?”, select ‘Yes’.



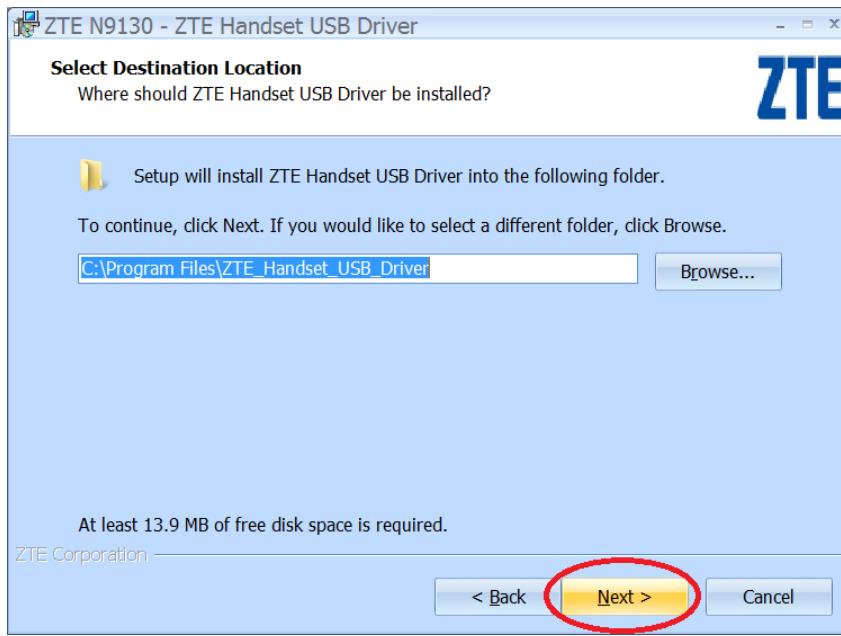
Step 5: The Select Setup Language dialog should be displayed. Select ‘OK’.



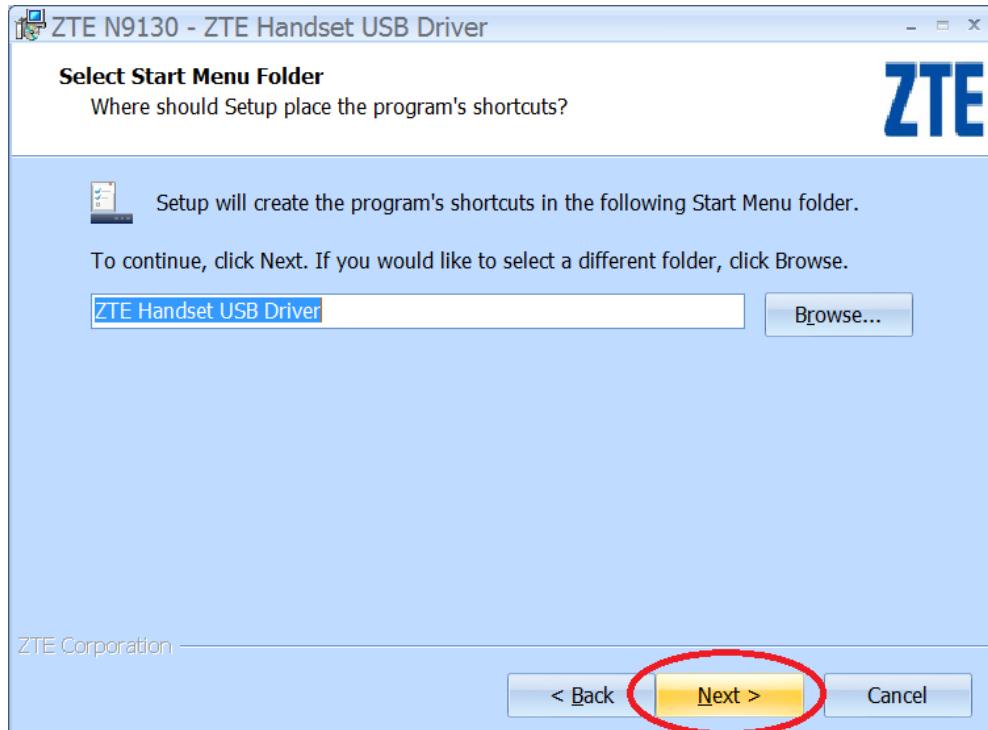
Step 6: Select 'Next' to continue with the installation.



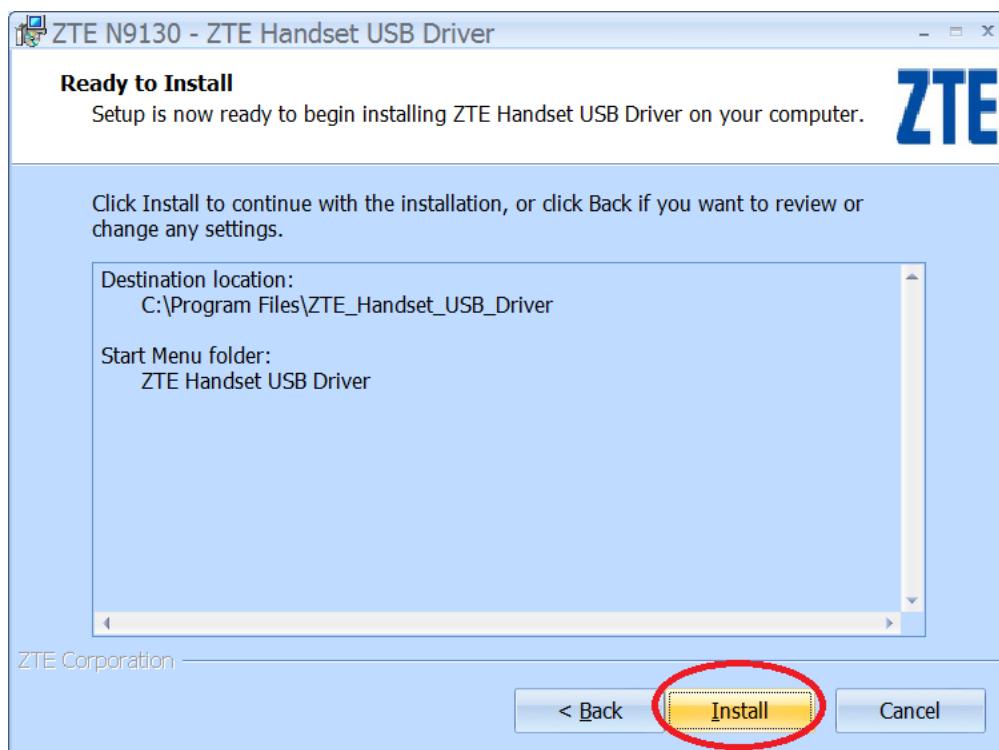
Step 7: Select 'Next' to accept the default location for the driver.

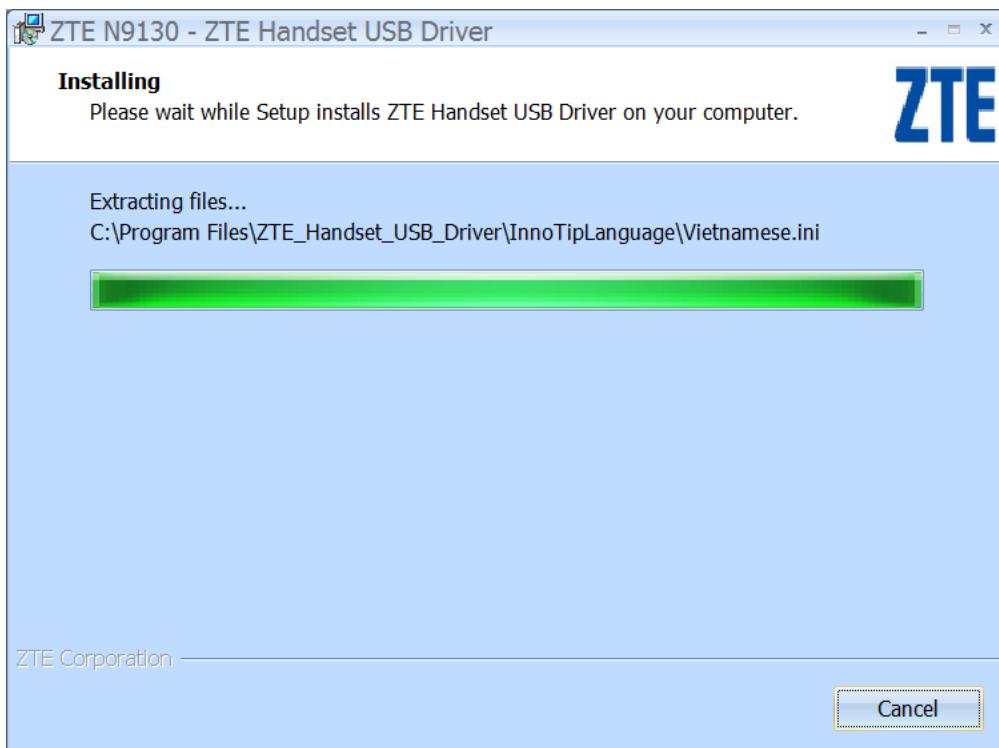


Step 8: Select 'Next' to accept the default location of the driver shortcuts.

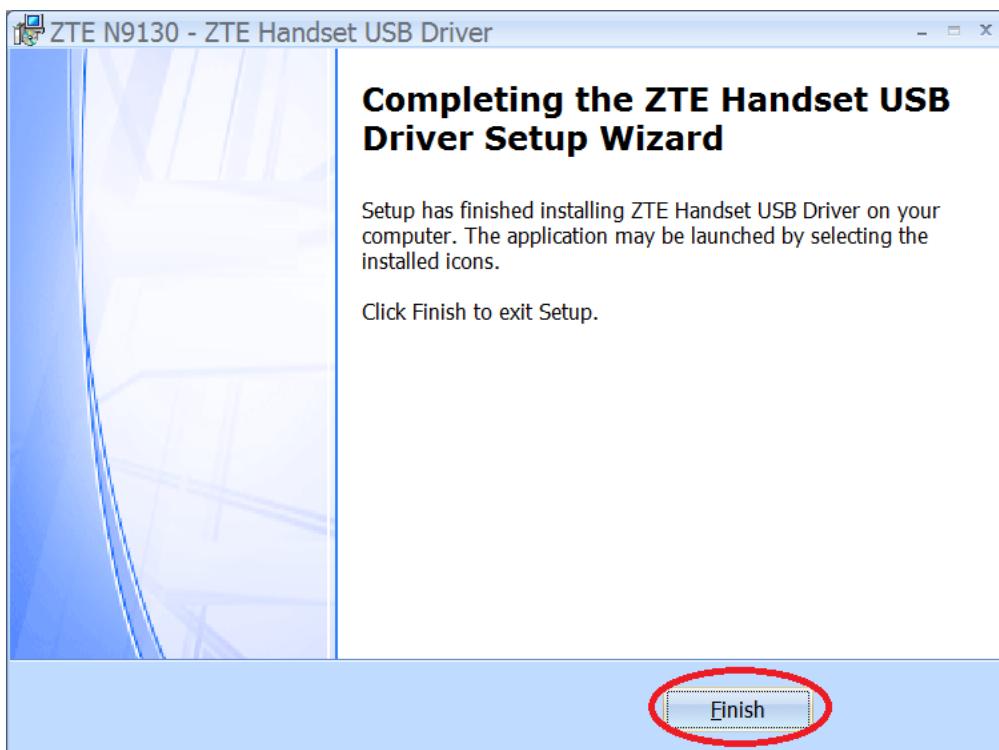


Step 9: Select 'Install'. The image below the following shows status as the driver is being installed.





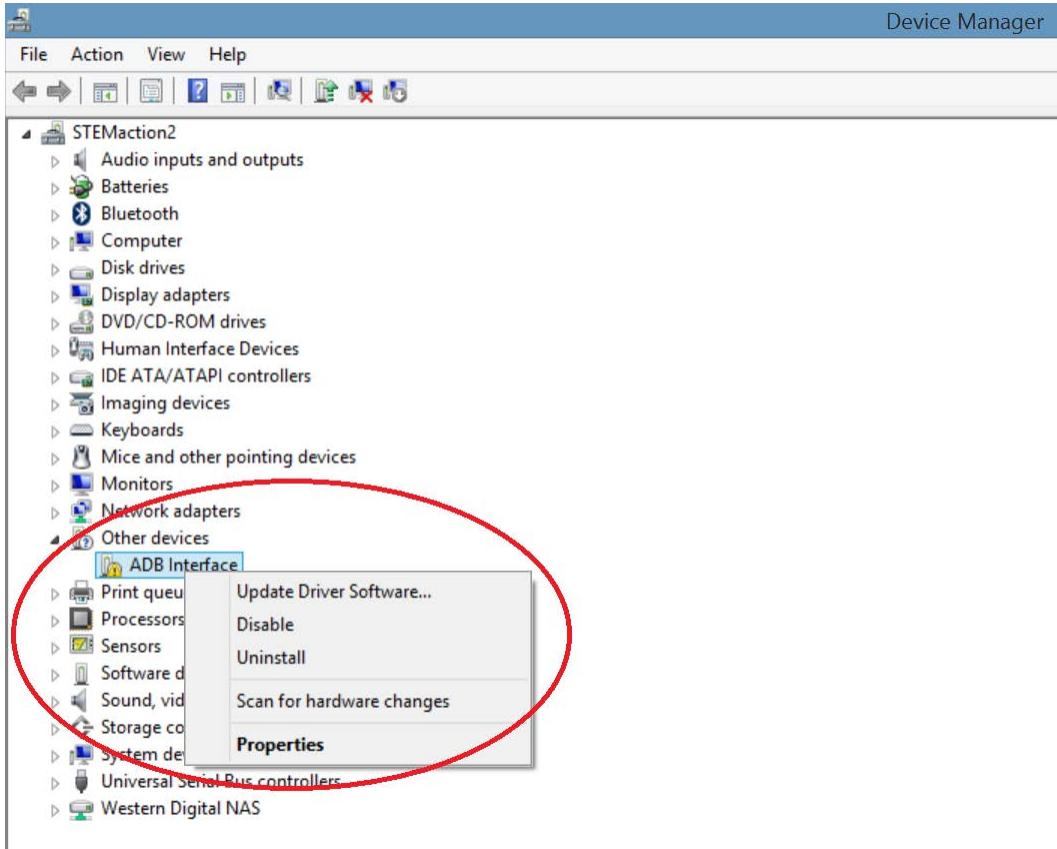
Step 10: Select 'Finish'.



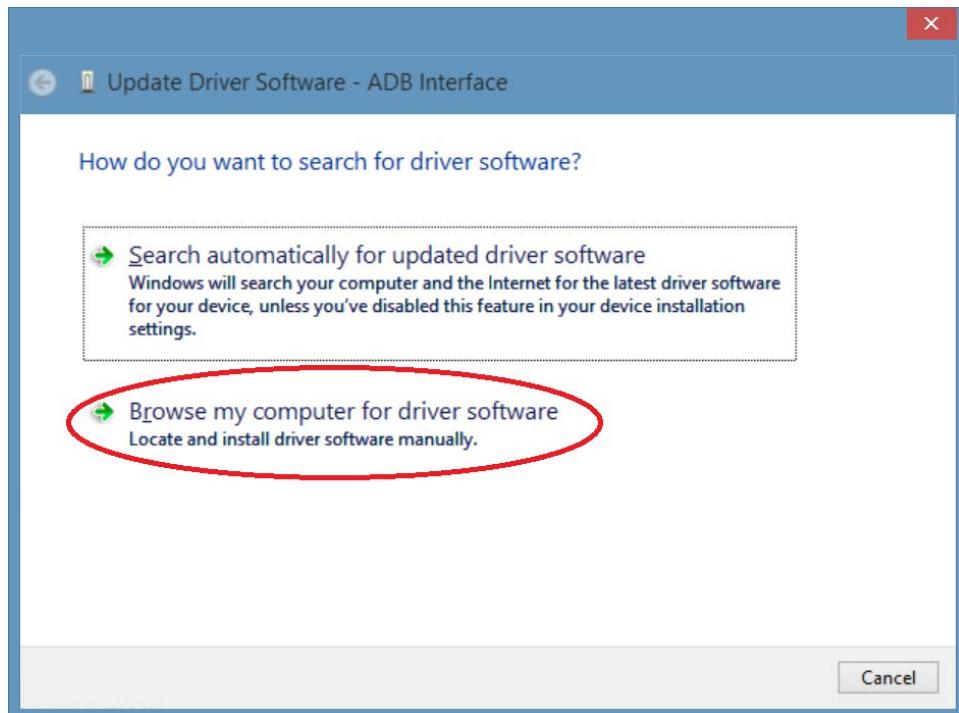
Step 11: Press the Window key and type 'device manager' and press return (similar to opening Internet Explorer).



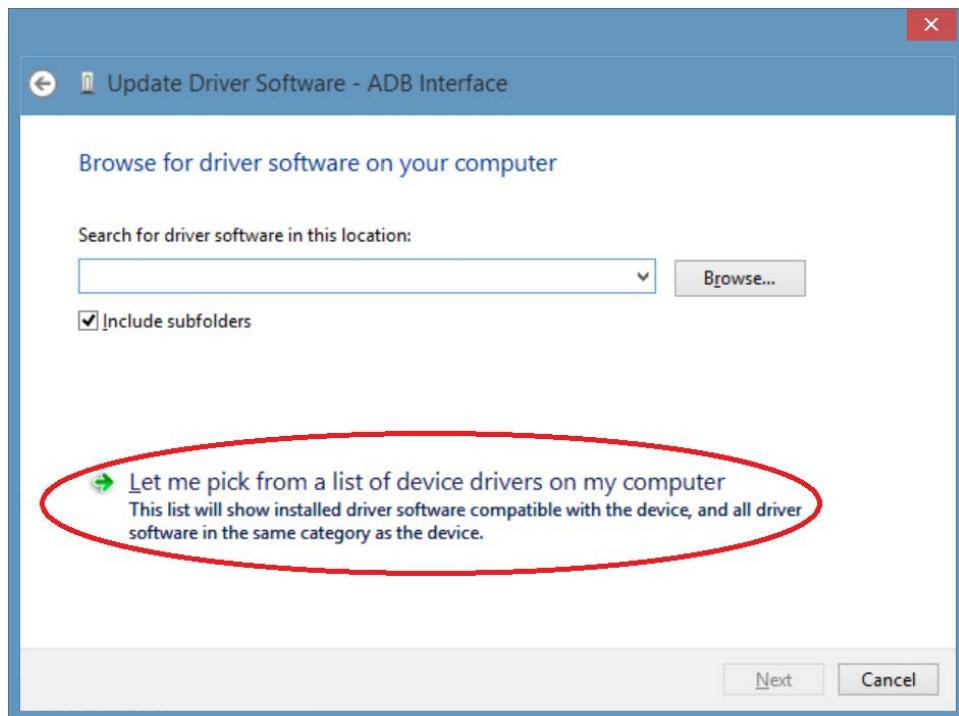
Step 12: Right click on ADB Interface and select Update Driver Software.



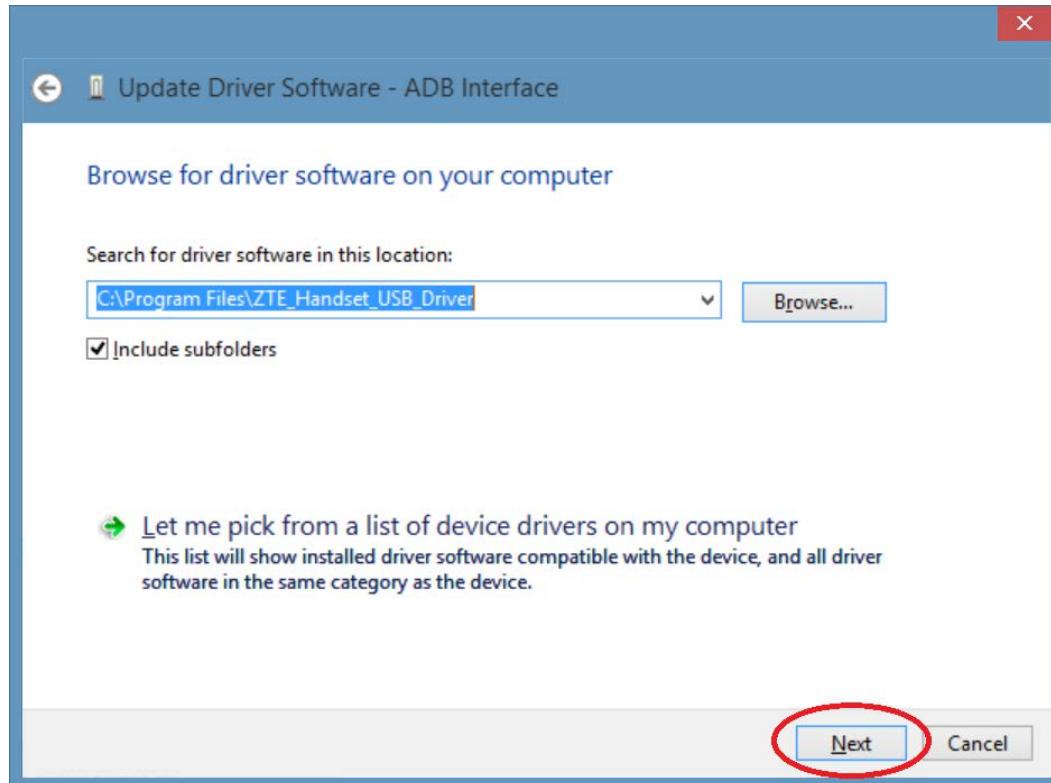
Step 13: Select 'Browse my computer for driver software'.



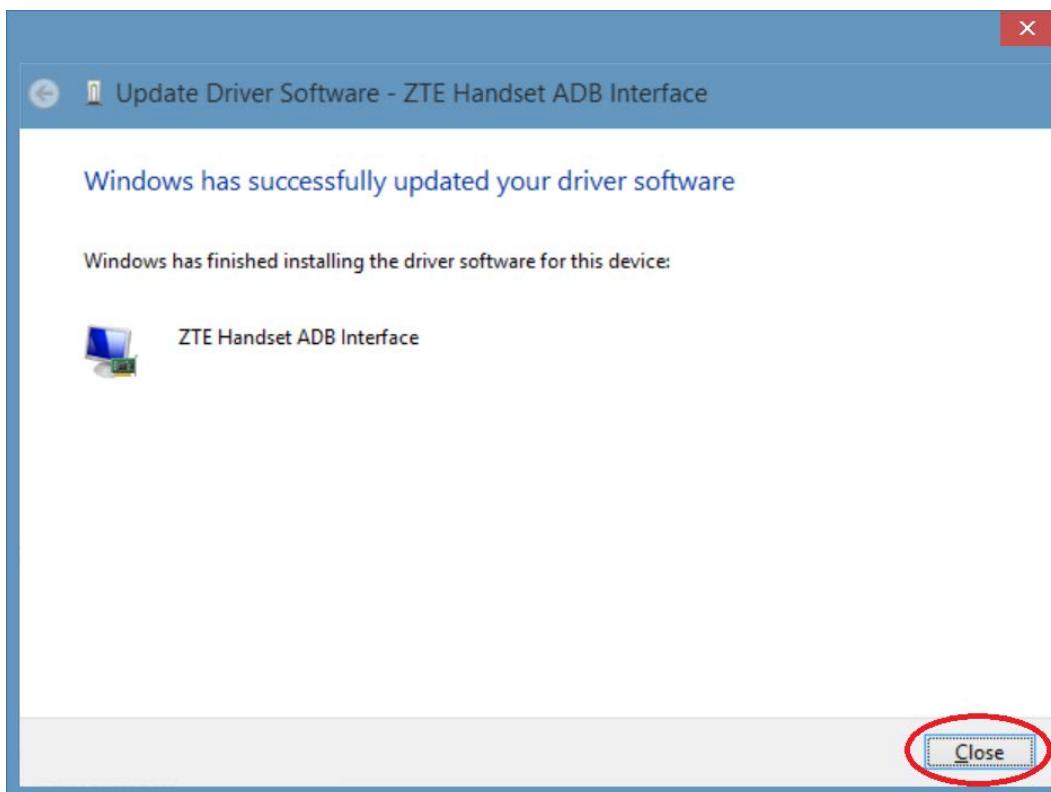
Step 14: Select 'Let me pick from a list of device drivers on my computer'.



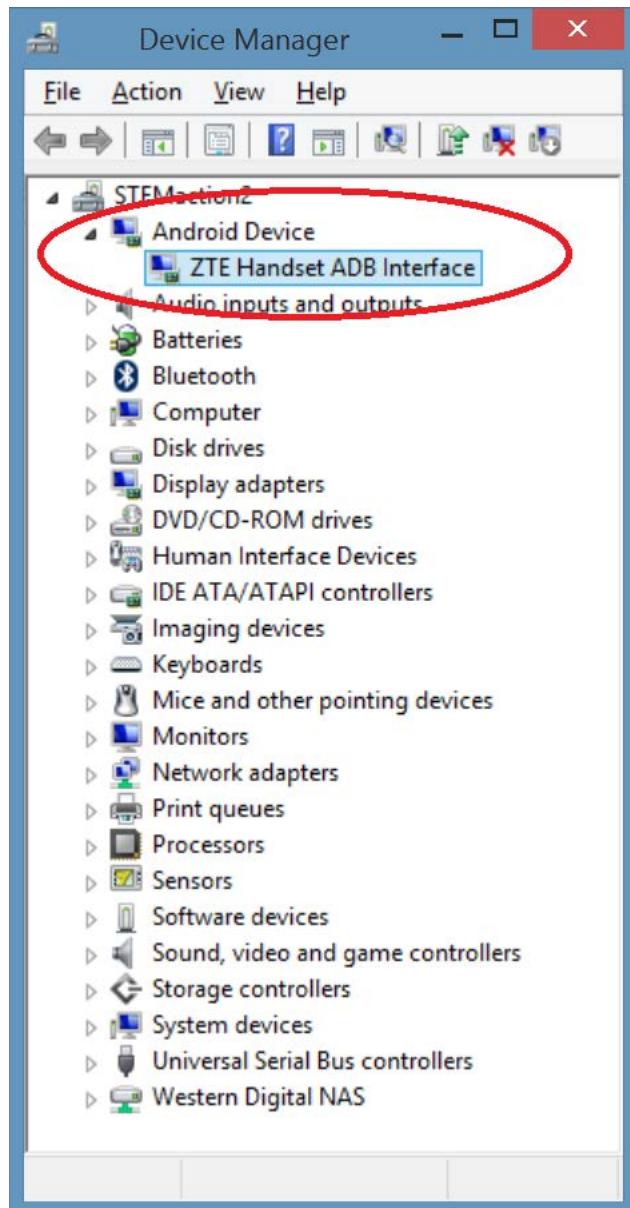
Step 15: Select 'Next'.



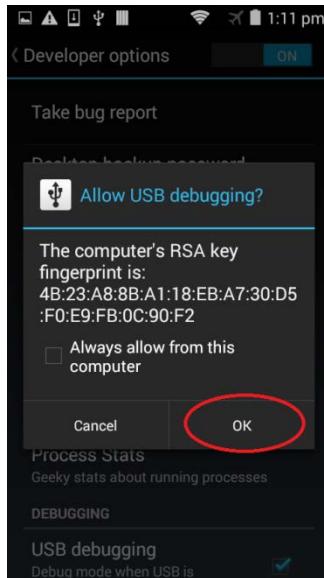
Step 16: Select 'Close'.



Step 17: Observe that the device is now connected. This window can be closed.



Step 18: Observe that the cell phone is displaying the ‘Allow USB debugging’ window. Select ‘OK’.

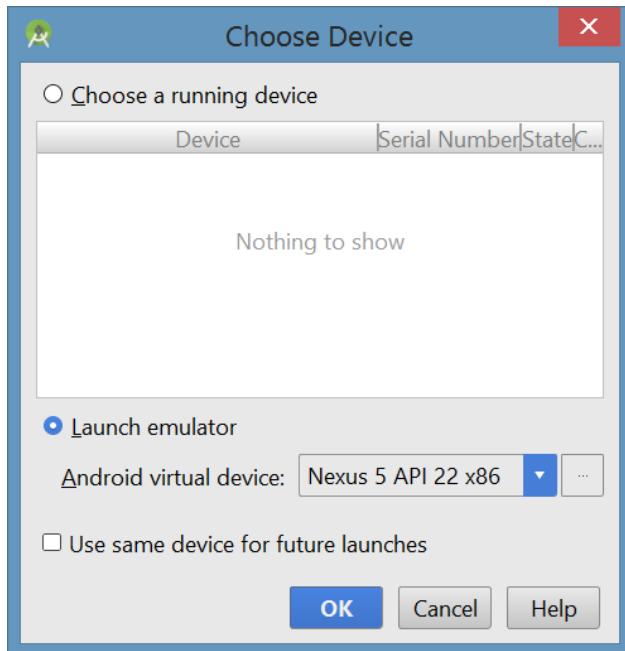


### ***Deploying the Robot Controller***

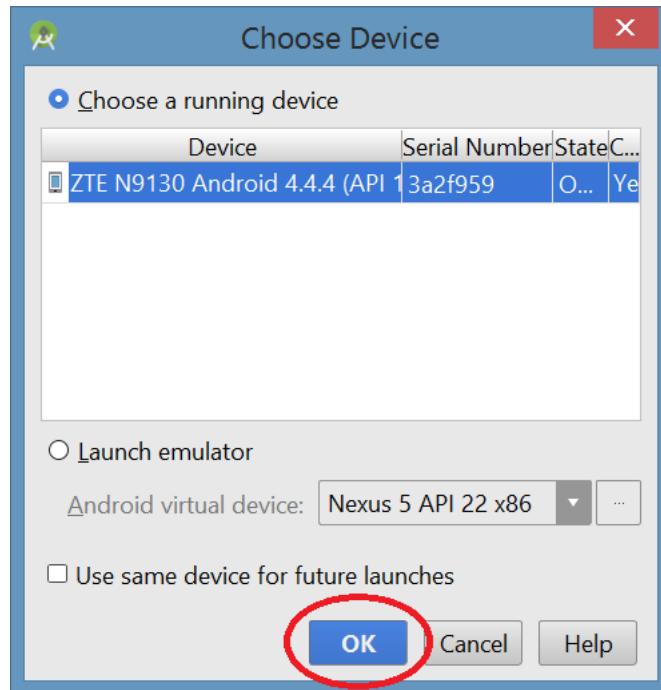
For this section choose one phone to be the Driver Station and the other to be the Robot Controller. The Robot Controller will host the Robot Controller app, which implements the PushBot Op modes. Android Studio can't deploy the Robot Controller app until debugging is enabled on the phone, so that will be the first task.

The correct driver for your Android device will need to be installed onto your Windows computer. If your Windows computer does not have the correct driver for your phone installed, then when you connect your device to the computer via a USB cable, the Windows computer will think that the device is a media player or storage device (like a Flash drive). You might be able to access files on your device in this mode, but your Android Studio software will not recognize the device as an available Android device that it can talk to.

Step 1: From a previous section, Android Studio may still be open and displaying the Choose Device dialog. Close the Choose Device dialog if it is still open and the device list is empty (shown below). If Android Studio isn't open, then start it. Build the project (small green triangular button –shown in the aforementioned section).



Step 2: The device should be shown in the list. Select 'OK'.



Step 3: Observe that the application starts and the Active Configuration File activity is displayed.



### ***Deploying the Driver Station***

The driver station is the Android device that sits near the Team drivers and is connected to one or two USB gamepads. Teams use the touch screen and the gamepads to operate their Robot remotely. The driver station Android device runs a special app called the FTC Driver Station. The instructions in this section are directed at the Driver Station cell phone unless noted otherwise.

Step 1: Open the application activity by selecting its icon (shown in the red circle).



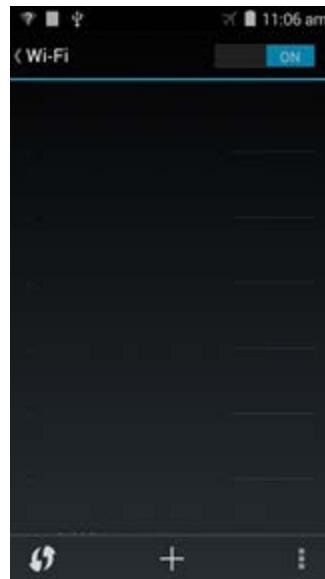
Step 2: Open the settings activity by selecting its icon (shown in the red circle). It might be necessary to swipe the screen left or right until the icon comes into view.



Step 3: Click on the word ‘Wi-Fi’, not the on/off button.



Step 4: Connect to a Wi-Fi network that is able to reach the Internet and to which the login information is known. The image below shows no networks. If networks are available, then they will be shown in the middle of the window. Select a Wi-Fi network and enter any required login information.

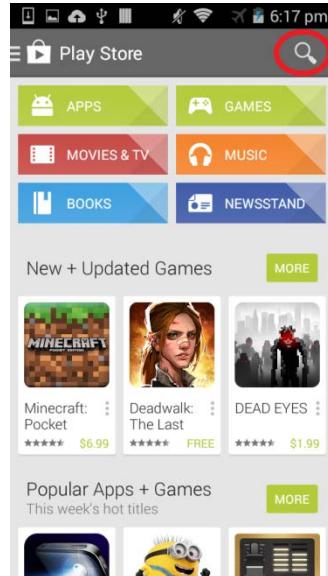


Step 5: Return to the home screen by pressing the home button (the center icon on the bottom of the phone's screen).

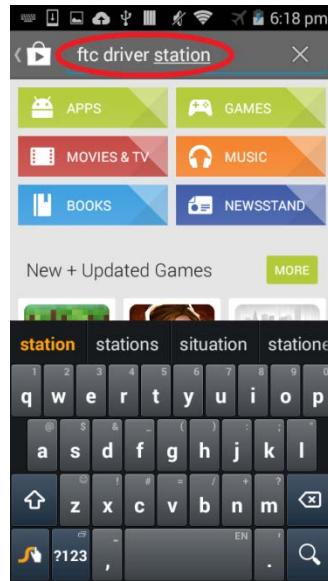
Step 6: Launch the Google Play Store app (orange circle). The Play Store app might prompt you to either login to an existing Google account or create a new one. Follow the onscreen instructions to either create a new (free) account or login to your existing account. The Driver Station app is free and no payment method is required. If the application requests a credit card number or some other method of payment, select the “Skip” button.



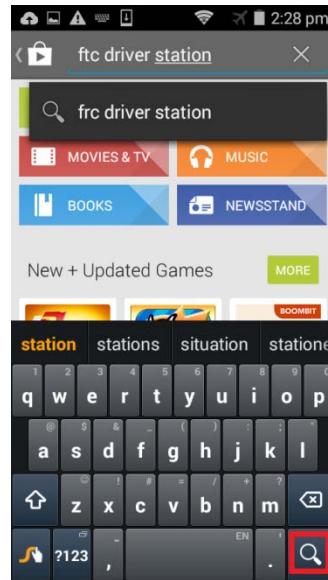
Step 7: Click on the search icon.



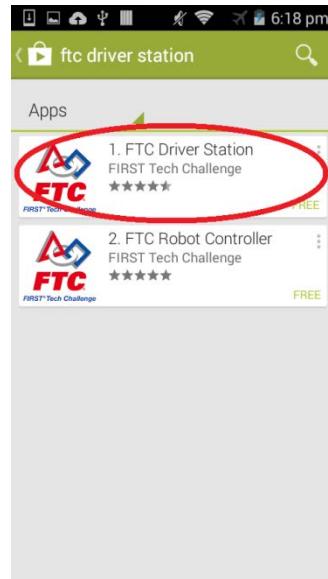
Step 8: Enter the phrase “FTC Driver Station”.



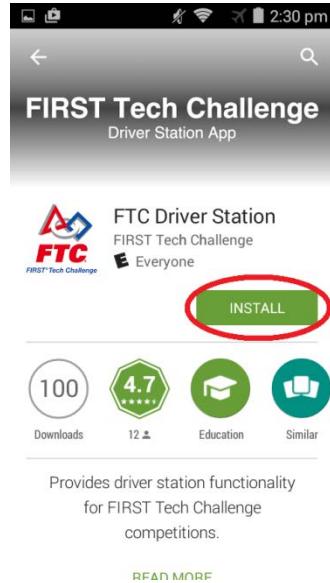
Step 9: And press the magnifying glass in the bottom right corner (on the keyboard).



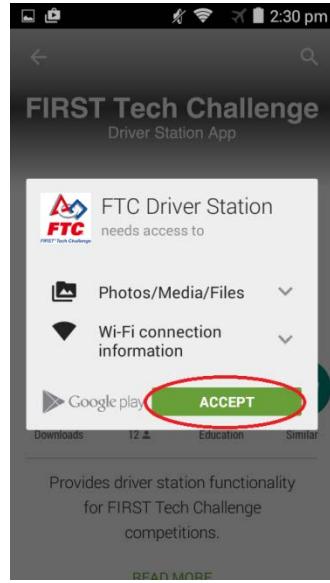
Step 10: Select the FTC Driver Station block.



Step 11: Select ‘INSTALL’.



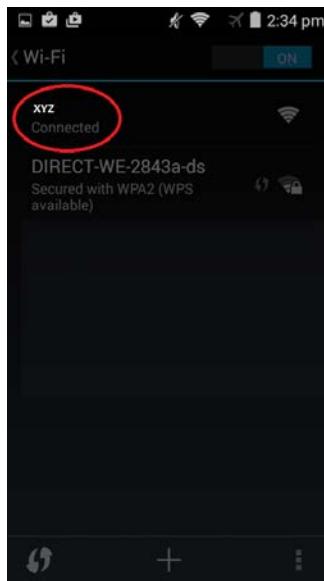
Step 12: Select ‘Accept’.



Step 13: Return to the home screen. Observe the FTC Driver Station icon is displayed there. If it isn't shown on the home screen then search for it in the app drawer.



Step 14: (IMPORTANT) The network needs to be forgotten to prevent connection issues between the Driver Station cell phone and the Robot Controller cell phone. Open the Wi-Fi setting activity as shown earlier in this section. Long hold the name of the connected network.



Step 15: The following screen should be displayed. Select ‘Forget network’. In general, the Driver Station and Robot Controller cell phones should not be connected to any wireless networks with the exception of the Wi-Fi Direct connection between the two cell phones.

