

Chapter 1 : Android Adventures - Getting Started With Android Studio 2

Read Next: Java tutorial for beginners Android Studio was first announced at a Google I/O conference in and was released to the general public in after various beta versions.

I mean, think about it, solving the years-old problem of not being able to find a date with just an app, making it extremely easy for stress eaters to order in that hot chocolate fudge and eat their hearts out, allowing you to shop your favorite shoes with the tap of a click, Android is, in the truest sense, connecting people to each other and to their utilities. With the rampant emergence of technology and the growing number of Android users, Android app development is the new buzz in town. What is Android Studio? To develop an app, you need a text editor to write your code, a compiler to execute your code, and a debugger to test your app. An IDE is a software that allows a developer to perform these functions and to run her app on a mobile device in real-time. Android Studio largely focuses on accelerating your app development process and to ensure the same, it has a lot of cool features, some of which are listed below. Gradle based build system Android Studio uses a Gradle based system which is based on Groovy and Kotlin. Gradle supports the automatic download and reconfiguration of other libraries. It is a controller that allows one to use external libraries while writing an app. For instance, if you wish to improve the UI of an app, you can import external libraries on Android Studio using Gradle. Kotlin lint checks Since Google announced Kotlin as the official Android app development language, it has enhanced Kotlin language support in Android Studio. IntelliJ IDEA Developers are not unfamiliar with unseeable errors in a code like a missing semicolon or incorrect variable declaration. Much like that elder sister who saves you when your parents catch you talking to your girlfriend, IntelliJ saves you from the pain of scrolling down for hours by suggesting relevant changes in your code. It comprises various libraries, APIs, debugger, and emulator among many others. For running these tools so as to create an app, one needs an IDE. Now, you know what an Android Studio is. Setting up Android Studio a. Tool requirements Java Development Kit which you can download here. While downloading Android Studio, ensure that you have enough free space on your system and have already installed JDK. Download the latest version of Android Studio here. If you are using Windows, download a. You can also see this step-by-step installation guide for Android Studio. A project defines your workspace for app development from source code to build configurations. The very next step will be to select an activity, which represents a single screen with which users interact to perform a specific task like clicking a picture or sending an email. The next screen allows you to configure your activity by letting you add the activity name and layout name. The Android Studio opens with a text editor that has two windows: This name is MainActivity. The second phase involved in the app making process is build and run. This phase involves compiling the source code and app resources, packaging them into debuggable APKs, and running them on the emulator or a device powered by Android. Android Studio uses Gradle to manage the build process. The build files are named build. Connect the device on which you wish to run your app using USB. Make sure that you have enabled the USB debugging option from your Android device. If you do not have a device, you can run your app on an emulator. The Android Emulator allows you to develop and test your apps in a run-time environment without using any Android-powered physical device. The emulator uses Android Virtual Device AVD , a configuration that defines the characteristics of the Android device you wish to imitate in the emulator. The next phase of the Android app development is debugging and testing. This involves modifying and rewriting certain parts of the code so as to remove the existing bugs from the app and optimizing the app performance. You can know more about signing your app here. Well, this was just the surface of Android Studio. To learn Android Studio in and out, you can visit Android Developers the official Android tutorial for beginners. Android app development requires much more than the knowledge of Android Studio you need to be comfortable with programming in Kotlin or Java and XML. How to learn Android Studio and Android app development? What is it, you ask? From setting up JDK and Android Studio to building user interfaces using XML and adding functionalities using Kotlin, this Android training walks you through all the necessary T-points of the road to learning Android app development. With over 1 billion Android users and 2 million

DOWNLOAD PDF ANDROID STUDIO TUTORIAL FOR BEGINNERS

Android apps on Google Play, Android app development is the hottest industry. You may also like.

Chapter 2 : Learn Android Studio - The complete Android tutorial for beginners

Android tutorial for beginners - Learn Android Programming and how to develop android mobile phone and ipad applications starting from Environment setup, app.

Android represents a big potential market. It is also the most open of the "big" phone platforms. All the tools you need to create an Android App are free. All that stands between you and your Android app is your imagination and programming ability. In this book I will show you the fundamentals of Android programming. Not the tips and tricks, but how to think about what is going on. What do you need to know to get started? You need to be able to program in a modern object-oriented language. You might well need to look things up about the specifics of particular features of Java, but most of the time it should be obvious or obvious with the help of a few comments. That is, a lot of Android programming is just a matter of following the rules. However, if you hope to produce something unique and useful you will at some point have to add something of your own - and here creativity and skill are required. So you might not need to be an expert programmer to get started, but you need to become one by the time you create your amazing app. Fortunately practice is a good teacher and so learning to make the most of Android Studio will actually help you learn to code.

Getting Started This is a really good time to get started with Android because Google has released version 2. Put simply it is the way of the future and so worth your investment in learning it. With the release of Android Studio Google has stopped work on the Eclipse add-in and this means that Android Studio really is the only way to develop apps from now on. It is worth, however, going over the basic principles. Once you have Java, setting up Android Studio is easy: Follow the setup wizard to install Android Studio. Drag and drop Android Studio into the Applications folder. Accept any defaults that the setup program offers you - unless you have a good reason not to. It installs not only Android Studio, but the SDK and the virtual device system that lets you test your application. Download the compressed file and uncompress it to a folder of your choice. You will find the executable in the bin subfolder. It appears to start and then just vanishes. If you encounter this problem, you need to set an environment variable indicating the correct location of the Java JDK.

Android Studio is the official IDE (integrated development environment) for developing Android Apps by Google. It is based on JetBrains' IntelliJ IDEA software and has lots of amazing features which helps developer in creating Android App.

You will need these tools regardless of which version of Android you are targeting. These are what will actually create the APK – turning your Java program into an Android app that can be launched on a phone. These include a number of build tools, debugging tools, and image tools. The Build tools were once categorized under the same heading as the Platform tools but have since been decoupled so that they can be updated separately. As the name suggests, these are also needed to build your Android apps. This includes the zipalign tool for instance, which optimizes the app to use minimal memory when running prior to generating the final APK, and the apksigner which signs the APK surprise! The Platform tools are more specifically suited to the version of Android that you want to target. Generally, it is best to install the latest Platform tools, which will be installed by default. After first installation though, you need to keep your Platform-tools constantly updated. The tools should be backwards compatible, meaning that you will still be able to support older versions of Android. Anatomy of an app: It relies on Platform-tools in order to understand the Android version that is being used on said device and hence it is included in the Platform-tools package. You can use ADB to access shell tools such as logcat, to query your device ID or even to install apps. The Android emulator is what lets you test and monitor apps on a PC, without necessarily needing to have a device available. To use this, you also get an Android system image designed to run on PC hardware. I also recommend this resource on the build process that will help put the SDK into a little more context. They provide a kind of bridge between Android Studio and a physical device or emulator so that your app can be appropriately packaged and then tested as you develop. For the most part, you can leave the SDK alone: Android Studio will recommend necessary updates and it will call upon the required components when you hit Run or Build APK. That said, a few of the tools are also directly accessible, which will be used for things like updating the SDK, or directly monitoring and communicating with your Android device. If you are following along with an Android development tutorial, then you might sometimes get directed here in order to ensure that specific components are up-to-date. This lets you build your own emulators. This works with either an emulator or a connected device and will go a little deeper in monitoring the way your Android device and app are behaving. To do this, you will need to find your Android SDK installation folder and navigate to the platform-tools directory. On Windows, hold shift and right click anywhere in the folder to open a command line. On Mac, just open Terminal from Launchpad usually found in the Other folder. Now you can use a number of commands. You can find a list of the ADB commands here. Accessing the Documentation Looking for a specific Android development tutorial? There was a time when the Android SDK would also come packaged with a selection of useful sample projects. Today this is no longer the case, but you can find them instead by opening Android Studio and navigating to File – New – Import Sample. You may wish to use another IDE Integrated Development Environment , for instance if you want to streamline the process of making a 3D game in which case, you may wish to use Unity or Unreal , or if you are interested in cross platform mobile development in which case you might use Xamarin. You can also find the location of the Android SDK in Android Studio, in case you should ever need to move it, or just for your own reference. Just go to File – Project Structure. Be aware that this folder is hidden on Windows by default, so you might have a hard time finding it. This gives you access to certain libraries and can help to squeeze a little more performance out of a device – making it useful for game development, among other things. As mentioned, if it is just the SDK you are interested in, then you can download this on its own by visiting the downloads page and then choosing to include the sdkmanager. This will allow you to update the SDK through the command line. But for the vast majority of users, it makes a lot more sense to install the full suite and enjoy the graphical interface and other conveniences – even if you intend on using a different IDE for development. And this is the really good news: Android development is now easier than ever before thanks to the leaps and bounds that

Google has made with Android Studio. There was a time when setting everything up was considerably more complex. There has never been a better time to start Android development!

Chapter 4 : 23 Best Android Development Tutorials and Resource for Beginners in

The curriculum includes Installation, Activities, Layouts, List Views, SQLite and Services blog.quintoapp.comng Your First App, Android tutorial, Searches related to android tutorial for beginners.

This beginning Android development tutorial is now up to date with the latest version of Android Studio. Updates by Megha Bambra. Original tutorial by Matt Luedke. Previous updates by Darryl Bayliss. Download and install Android Studio. Set up testing for your app on devices and emulators. Import a sample project into Android Studio. Even if you follow the steps perfectly, you may have to troubleshoot a small issue or few. Your system configuration or product versions can make for unexpected results. You can find the Terminal app quite easily on a Mac: Once you have the Terminal open, type in `java -version`. You should see some output that mentions a version number, like below. Terminal might tell you `-bash: Google constantly updates this page, so the version you see may very well be newer than the screenshot above.` After reading these carefully everybody takes the time to fully read these, right? Once the download is complete, you can install Android Studio similar to how you install any other program. Once installation wraps itself up, go ahead and launch Android Studio! The setup wizard will greet you the first time it loads. Click Next to move to the Install Type screen. This whole process will probably take several minutes. Check the box for Standard and click Next. Click Finish to start downloading the SDK components. Once everything downloads, click Finish. Check whether any updates are available by clicking check for updates at the bottom of the welcome screen. If an update is available, a window like the screenshot below will appear. Select Update Now and let it do its thing. From the Android Studio welcome screen, click Configure. The menu will slide across and present the Configure menu. Select the SDK Manager option. Take note of the checkbox next to the SDK platform; it will be pre-selected if an update is available. By default, the SDK Manager installs the latest packages and tools. Select the SDKs as shown in the screenshot above. If you wish to install other SDKs, just select them for installation. The SDK Tools tab lists developer tools and documentation along with the latest versions. Each contains components that are designed to assist in the development of Android and work across multiple SDKs. Go with the default selection on this tab. For the purpose of setting up correctly, select the options that are checked in the screenshot above. Click OK to close out the window. The window will disappear and the SDK Manager will download and install the selected items. Now the fun begins! Creating Your First Project Android Studio has a nice little step-by-step tool to help you create your project. Identify Your Project Android Studio will present you with a project creation screen: Feel free to put your own name in the Company Domain text field. The Package Name is used to uniquely identify your app so that any work performed by a device is always properly attributed to the source, thus preventing confusion between apps. Click Next at the bottom of the window. This is where you select device types and operating systems to target. Selecting this value is simply a matter of balancing the capabilities you want and the devices you want to support. This is where developing for Android can get a little tricky. As you change the Minimum SDK in the drop down menu, the percentage in the text underneath reflects what percentage of devices currently run that version of Android. For more information on API versions and their uses, check out the Android Dashboards , which are updated every few days. Think of an activity as a window within your app that displays content with which the user can interact. An activity can take up the entire screen or it could be a simple pop-up. Your options on this particular template range from a blank activity with an Action Bar right up to an Activity with an embedded MapView. Select the Blank Activity option and click Next. This gives your activity a name to refer to in code. Android Studio takes this as a cue to go do a bunch of behind-the-scenes operations and create your project. You see your project name, which is familiar. You can customize your configurations to have development or production versions of the app that behave differently, or you can add dependencies for third-party libraries. Maven Maven is another project build tool, and it can also refer to the Maven Central repository of java libraries. After a brief moment, Android Studio will finish building your project. The project is pretty empty, of course, but it has everything it needs set up so that it can be launched on an Android device or emulator. Now to dress it up and work through building and running this project on an emulator. So how do

you run it? Android Studio comes with the ability to set up a software-based Android device on your computer and run apps on it, browse websites, debug and everything you would expect from a simulator. This capability is known as the Android Emulator. You can set up multiple emulators and set the screen size and platform version for each to whatever you like. Android Studio makes use of some useful software developed by Intel to ensure your emulator runs quickly. Up until recently, your computer would have to emulate everything an Android device would try to do, right down to its hardware, which runs an ARM-based processor. Most computers make use of x86 processors, meaning your computer would have to do computationally intense tasks that take a significant amount of time just to test your app. You still have the option to create an emulator that is as close to an actual device as you can, but be aware that the initial load times can drag a bit and have put off many an Android developer from using the emulator at all. Either way, work through the process of creating a new AVD so you know how to do it. Click **Create Virtual Device** in the bottom left to begin configuring a new virtual device. The first decision you need to make is what type of device. The Category list on the left shows all the types of devices you can emulate. In the middle, you see a list of specific devices. Take a moment to explore. For now, you just want to emulate a phone-sized device, but if you wanted to emulate an Android Wear watch or an Android TV then you have options to do so here. Select Nexus S in the list of devices available to you from the phone category and click **Next**. Select Lollipop and make sure the one selected has the value x86 in the API column so the emulator runs as fast as possible on your x86 computer. Go back to your selection and click **Download**. The last screen lets you confirm your choices and gives options to configure some other properties such as device name, startup orientation and RAM size. For now, use the defaults and click **Finish**. A new window will appear, asking you to choose the device you wish to test your App on. You currently have no devices running, so select the Nexus S you just created. See this thread on Stack Overflow for more troubleshooting tips. You just made your first Android app. **Running on a Device** If you have an Android device and want to run your app on it, follow the animated GIF on the right. It demonstrates how to enable developer mode on your device. Here are the step-by-step instructions to enable Developer Mode on an Android device: Go to Settings on your device. Scroll all the way down and select About phone. Scroll to Build number and tap in multiple times. Go back to Settings screen and scroll all the way to the bottom. Next, turn on the USB debugging switch under the Debugging section. Connect your device to your computer via USB. If this is a trusted machine, then check the Always allow from this computer option. The device you enabled the developer mode should now appear in this dialog. Select it and click **OK**. Go ahead and show it off to your friends.

Chapter 5 : Google Developers Training | Google Developers

9 Best Free Android Studio Tutorials for Beginners October 29, October 29, admin android, development, programming, tutorials Android Studio is an integrated development environment for Android platform that announced on at the Google I/O.

Android resource Here are 23 best Android development tutorials and resources for beginners, we will learn the most complete guide to the Android development and get a real job. According to a Google report , as of May , the number of active Android users has exceeded two billion, and this only continues to grow. However, Apple has established a foothold in the Chinese market. While we have no way of knowing which operating system will dominate in the future, it is clear that the number of Android users is exceeding iOS users. This is why more and more designers are becoming engaged in Android development. This article covers Android development tutorials for beginners, YouTube Android development tutorials, rich Android development resources, and Android app development software. Android app development for beginners 1. However, if you have some programming background, you will find it easy to follow this tutorial. Make sure you go through the entire tutorial as it offers some best practices at the end. Here are what you will learn from this tutorial: How to download and install Android studio How to test your app on different devices and emulators Create a simple "hello world" Android app that prints to your device and emulator How to import sample projects into Android studio 3. If you have a basic understanding of Java programming, it will be relatively easy to learn. This tutorial will teach you basic Android programming and then take you through some of the advanced concepts related to Android app development. After completing this tutorial, you will find yourself at a moderate level of expertise in Android programming. It can include everything needed to build an app, including source code, resource files, and Android manifests. It covers a lot of ground - from the basic introduction and installation of the Java JDK to installing Android studio and developing an app. You can learn everything you want to know about Android development from these 56 videos. From basic development knowledge to Android app creation, all the Android knowledge you need is covered in these 75 videos. After completing this series of tutorials, you will be able to create your own Android app and publish it in the Play Store. These tutorials were created in , so they are fresh and up-to-date. These days, Android development is done with a tool called Android Studio. It is user-friendly and is excellent in organizing projects. In this video, you will learn how to install Android Studio and create an app. Rich Android development resources 8. What Java is and how it works What to study after completing basic Java How to get a job with the skills you have acquired And more.

Chapter 6 : Android Development Studio Tutorial - Installation and Setup

The tutorial is a pretty detailed one - it starts with how to install Android Studio, how to create a new project, how to add functionality to it, how to run it, etc.

To make it easy for you and with no illusions that this list of Android tutorials is the best or complete, here are 12 Android tutorials to start with. Some of them start out for beginners and then delve into more advanced topics. If you encounter a hurdle, just spend more time with the tutorial, reading it a couple of times if necessary. If you are still not on friendly terms with it, there is no drama – just move forward and revisit it later. This is good because all the important content about the topic in one place and you just have to read it. This tutorial has more topics and information than the tutorial from Google, so if you are looking for an in-depth tutorial, this is one the. If you want to get the most from it, you will need quite a lot of time to read it from start to finish. It can be a great source if you need to consult a given topic in detail. Video Tutorials Series I find video tutorials less useful except when they teach design, animation, or any other visual topic but for many people they are the preferred way of learning. If you belong to this group, you will love this series of video tutorials. Similarly to the previous two tutorials, this series covers everything from absolute beginner level to advanced topics. The first two tutorials in this list are book-like but if you want something more authentic you could print them. Even better, a pdf tutorial, like this one , is a much better option. Similarly to the previous resource, this one might not be very up to date but it does cover the major principles of Android programming. This is one more general tutorial that covers Android development from beginner level to advanced. Game Development Series If you have some knowledge about Android but you want to delve into games development, this series of video tutorials is a great start. The series starts with the very basics of Android and Eclipse but my personal feeling is that if you are a total stranger to Android, the journey will be too hard. From what I saw, the series mentions general Android as well, not only game development. For some of these topics you can find information in the general tutorials as well but if you want more detail, this is for you. In this tutorial you will learn how to set up the action bar, how to add actions, how to split, hide, and overlay it, as well as how to add navigation. You will also learn about action bar interactivity, such as how to handle clicks on its items and to use action views. In such cases you need to know how to handle this data. This tutorial leads you step by step in the world of XML parsing. It also helps you create a parser that will look like the one shown in the next screenshot. Android for iOS Developers With the huge popularity of Android, even die-hard iOS developers are likely to consider switching or at least expanding to it. If you are an iOS developer, you are lucky because you are not new to mobile development as a whole. Of course, you could read the general Android tutorials I listed earlier but especially for you, here is a better tutorial. Unfortunately, some of the info in this tutorial might be outdated but with the rapid development of mobile programming technologies this is inevitable. This tutorial is great because it summarizes the differences between iOS development and Android development, thus making the change easier for you. The tutorial is a pretty detailed one – it starts with how to install Android Studio, how to create a new project, how to add functionality to it, how to run it, etc. Localizing Android Apps Android applications are popular all over the world. Your users speak different human languages, which means if you want to reach them, you need to think about localizing your Android apps. This tutorial explains it all. Getting Started with Android Library Projects At some point in your Android development career you will get tired of having to re-invent the wheel all the time and you will appreciate the advantages of reusable code. If you are already there, you will certainly want to know more about reusable code. In this case this tutorial will help you get started as quickly as possible. The first part warms you up with some basic concepts, while the other two delve into more detail about how and when to use Android Library Projects. So, if you have a spare minute, check the tutorials, learn something new and let us know your favorite tutorials. Meet the author Ada Ivanoff Ada is a fulltime freelancer and Web entrepreneur with more than a decade of IT experience. She enjoys design, writing and likes to keep pace with all the latest and greatest developments in tech. In addition to SitePoint, she also writes for Syntaxxx and some other design, development, and business sites.

Chapter 7 : Create an Android project | Android Developers

Android Studio for beginners, Part 4: For a few years now it's been clear that Android dominates the mobile OS landscape. This Java-based technology has sparked a new gold rush, with programmers competing to make money from their mobile apps.

There many paid courses and tutorials to learn the Android studio but there are also free and detailed tutorials too. There are step by step video tutorials and articles for absolute beginners too. Please feel free to use the comment section for your questions and advice. Getting Started " Android. You will find sets of lessons for each development category for Android. Android Development for Newbies Udemy. There are 75 lectures to learn everything you need to know to develop Android applications. Before starting this free course, you need to know some basic Java skills. If you are new to Java, you can click this link to check out the best Java tutorial for beginners. After that, you are ready to start learning with this awesome Android tutorial. This tutorial is great for beginner programmers to help them understand Android environment. When you complete it, you will find yourself at a moderate level of experince in Android programming. Basic understanding on Java programming will make things easier to understand. You can learn about basic controls, layout containers, UI graphics, menus and the action bar in Android Studio. After completing this tutorial, you should bookmark it because new lessons are on the way. Android App Development for Beginners Bucky Roberts is one of my favorite Youtubers because his tutorials are both entertaining and informative. In his Android development series, there are 77 videos covering all aspects of Android Studio. It describes how to create Android applications and how to use Android Studio. You will learn how to set up all the tools you need to become an Android developer and you will build your first Android app from scratch, by the end of this tutorial. There are 46 videos in this Android tutorial series to learn how to develop mobile apps step by step. If you want to subscribe to his channel, you can use this link. If you want to learn Android studio from a detailed video tutorial, you can try this.

Chapter 8 : 9 Best Free Android Studio Tutorials for Beginners

Android SDK tutorial for beginners. the Android SDK also comes bundled with Android Studio, the integrated development environment where the work gets done and many of the tools are now best.

This is the official IDE Integrated Development Environment for the Android platform, developed by Google and used to make the majority of the apps that you probably use on a daily basis. Prior to its release, Android development was handled predominantly through Eclipse IDE, which is a more generic Java IDE that also supports numerous other programming languages. Android Studio makes life significantly easier compared with non-specialist software, but it still has a little way to go before it can claim to be a completely intuitive and smooth experience. For complete beginners, there is an awful lot to learn here and much of the information available – even through official channels – is either out of date or too dense to make head or tails of. So just what is Android Studio? The programming language you will be using is Java and this will be installed separately on your machine. Think of this as an extension to the Java code that allows it to run smoothly on Android devices and take advantage of the native hardware. Java is needed to write the programs, the Android SDK is needed to make those programs run on Android and Android Studio has the job of putting it all together for you. At the same time, Android Studio also enables you to run your code, either through an emulator or through a piece of hardware connected to your machine. Google has done a lot of work to make Android Studio as powerful and helpful as possible. Setting up Android Studio is fairly straightforward and is easier than ever thanks to nearly everything being bundled into one installer. Remember, Android Studio is only really your window into Java! Follow the simple instructions during installation and it should also set you up with an Android platform that you will be able to develop with as well. Be sure to tick the checkbox to tell the installer that you want the Android SDK as well and make a note of where Android Studio itself and the SDK are being installed. These are the defaults that it selected for my installation: Pick a directory for the SDK that has no spaces in it. In some cases, this will be the entire app or in others, your app might transition from one screen to the next. This will include a menu in the top right corner, as well as a FAB button – Floating Action Button – which is a design choice that Google is trying to encourage. Pick the option that best suits the app you have in mind to build and this will impact on the kind of files you are presented with when you first start things up. What are all these files? To me, programming meant typing in a single script and then running that script. Android Development is rather different though and involves lots of different files and resources that need to be structured in a specific way. Android Studio exposes that fact, making it hard to know where to start! By default, this is MainActivity. Java but you may have changed that when you first set up the project. However, the actual layout of your app is handled in another piece of code entirely. Just to make things a little more complicated though, you can actually use any XML file to define the layout of any Java script called a class. This is set right at the top of your Java code, with the line: This also means that you could theoretically use the same XML file to set layouts for two different Java classes. A new empty activity, I love the smell of possibility in the morning! Your Java files are housed under java and then the package name of your app. Double click on MainActivity. Java and it will come to the fore in the window on the right. When you are editing XML files, you might notice two tabs down the bottom. In the Text view, you can make changes to the XML code directly by adding and editing lines. Everything in the resources folder needs to be lower case, which is why underscore is used a lot to separate file names into readable titles in the absence of camel case. This contains more XML files that hold the values of variables – things like app names and color values. You can create additional Java classes, XML files or entire activities at any point in order to add more functionality to your app. This is handy if you want to edit an image for example. Meet Gradle Android Studio tries to keep things nice and simple for users by providing all of the necessary tools and features in one place. Things only get more complicated once you need to interact with some of these other elements. You should be able to leave Gradle to do its thing most of the time, but you will occasionally need to jump into the build. One is to run it on your physical device and the other is to create a virtual device emulator to test it on. Running it on your device is simple. This is faster than ever right now

thanks to the Instant Run feature. Should something go wrong causing your app to crash or become unresponsive, then red text will appear and this will give you a description of the problem. It essentially saves you a ton of time versus blindly trying to guess what went wrong. Make sure to filter the types of messages you want to see here. You can also switch to the monitors tab and see useful information such as the CPU usage etc. The Android Device Monitor takes this monitoring a step further and lets you monitor everything at once, complete with handy UI. However, one of the biggest challenges for Android devs is fragmentation. This is essentially an emulator that you can use to mimic the look and performance of any other Android device, setting such things as screen size, power and Android version. To use the virtual device though, you first need to build one by downloading the required components and setting the specifications as you want them. For those wondering, you can treat this just like any other emulator and even access the Play Store to download your apps. The SDK Manager If you want to target a specific version of Android, or if you want to create a virtual device running a specific version, then you are going to need to download the necessary platform and SDK tools. Make sure to keep-up-to-date! Google has made this easy by building support right into the IDE itself. Likewise, you may find yourself needing to use GitHub, which lets you backup your apps online and handles version control for streamlined collaboration. While this might all sound like a headache, Google is taking huge strides to keep making these processes as simple and easy as possible. This tutorial would have been much more confusing a few years ago, even just the set-up stage! The best strategy is to get stuck in with a simple app project and to only learn the more advanced features as you need them. No Coding Experience Required. Whether you are an absolute beginner with zero coding knowledge or a veteran programmer, this course will guide you through the process of building beautiful, functional Android apps and bring you up to speed on the latest features of Android and Android Studio. The package includes over 6 hours of high quality videos and over 60 different lessons. Claim your discount now using exclusive promo code: This is your ticket to a lucrative future in Android App Development. What are you waiting for?

Chapter 9 : Android Studio tutorial for beginners - Android Authority

Android Studio - Learn Android Programming and how to develop android mobile phone and ipad applications starting from Environment setup, application components, activity lifecycle, service lifecycle, application architecture, publishing application, debugging applications, handling events, layouts, menus, user interface controls, styles and themes, handling rotation, fonts management, send.