Mobile Computing

This is a short semester course in which we will learn some basic android development using android Studio and We will also be using Git/GitHub as Version control.

Topic covered:

1. Git/GitHub (lecture 2-3)
   1. Installing Git
   2. Creating GitHub Account
   3. Creating a repository
   4. Managing local repo (clone, changes, commit, push)
   5. Pull Changes form website (GitHub.com)
   6. Remove files locally
   7. Branching the repository

**Let’s Start this short journey of ours starting with introduction of Git/GitHub…**

Git/GitHub

Git and GitHub are ***not***the same thing. Git is an open-source, version control tool created in 2005 by developers working on the Linux operating system; GitHub is a company founded in 2008 that makes tools which integrate with git. You do not need GitHub to use git, but you cannot use GitHub without using git. There are many other alternatives to GitHub, such as GitLab, Bitbucket, and “host-your-own”.

Following are some step-base instructional introductions in order to understand the basic and use the Git/GitHub throughout the course.

**Install Git**

The first two things you'll want to do are install git and create a free GitHub account.

* **Installing on Linux**
  + If you’re on a Debian-based distribution, such as Ubuntu, try apt:
  + $ **sudo apt install git-all**
* **Installing on macOS**
  + The easiest is probably to install the XCode Command Line Tools. On Mavericks (10.9) or above you can do this simply by trying to run git from the Terminal the very first time.
  + $ **git –version**
  + If you don’t have it installed already, it will prompt you to install it.
* **Installing on Windows** 
  + The most official build is available for download on the Git website. Just go to <https://git-scm.com/download/win> and the download will start automatically.

**Create GitHub Account:-**

* Go to <https://www.github.com>
* Sign Up to create the free account



**Create a local git repository:-**

1. *Click on the + sign on the left top corner*
2. *Click on New Repository*



1. *Give details of the repository*



1. *Click on Create Repository*

Repo will be created

1. *Copy The link of your repository for further use*



**Manage repository locally**

1. Clone repository locally by **git clone** command using URL copied in last step



1. Create a file and add data to it



1. Now push this html file to main repo using git add, commit and push commands



1. Changes will be applied at main repository at github.com too.



1. Now, lets make changes on web(github.com) and make it appear on PC using cmd **git pull**
   1. Now to edit from online GitHub editor click on edit button (pencil icon) at top right



* 1. Write something …



* 1. Commit the change



1. Now to update local repo we will use git pull command,



1. Local repo will be updated



**Git Remove:-**

1. Let us add a file in our repository



1. Now we check it on our main repository



1. Now we remove it using git rm “filename” command and commit changes and push changes to main repo



1. Now, we verify it on our main repo



**Git Branch: -**

1. Let us create a branch in our project so that we can make changes in it separately and marge when required. We use git checkout to navigate among branches. Now we add some text to it.



1. When we checkout to other branches we can see that other branches is at different version (before creating txt file)

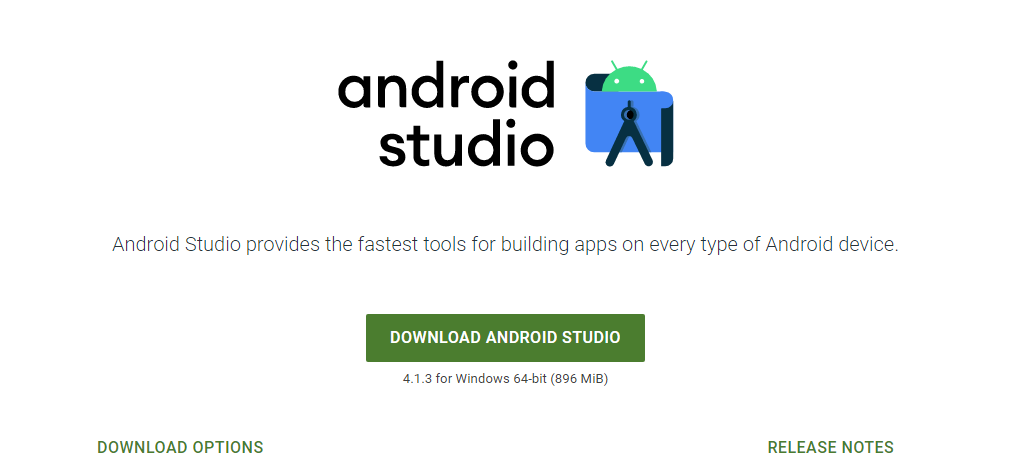


Mobile App Development

**Introduction**

Installing Android Studio:

1. Go to <https://developer.android.com/studio>. Download installer

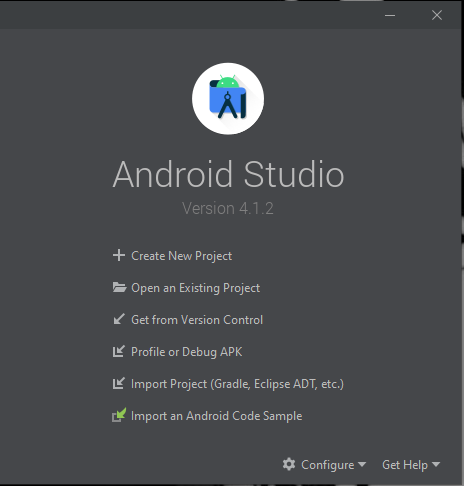


1. Use installer to install on your device

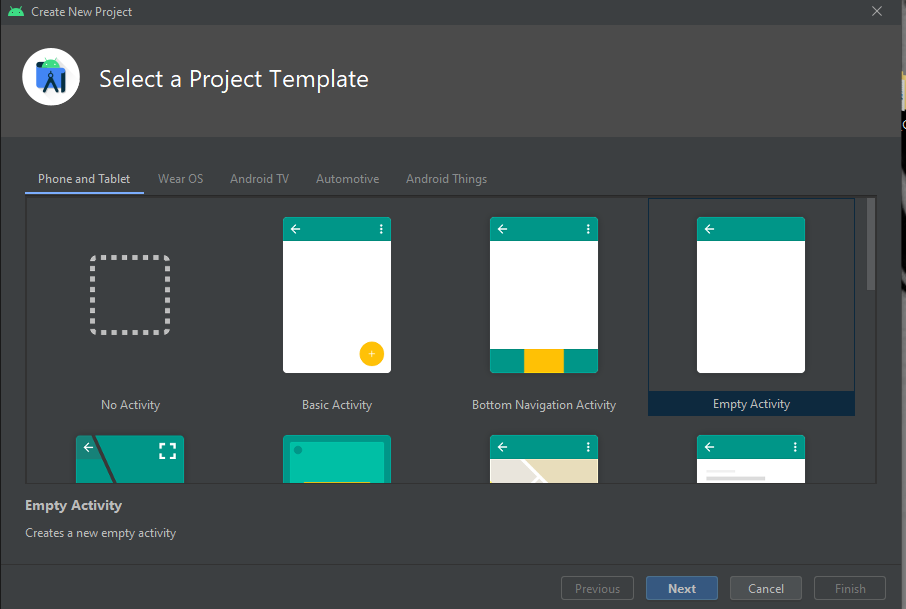


**Creating empty Application:**

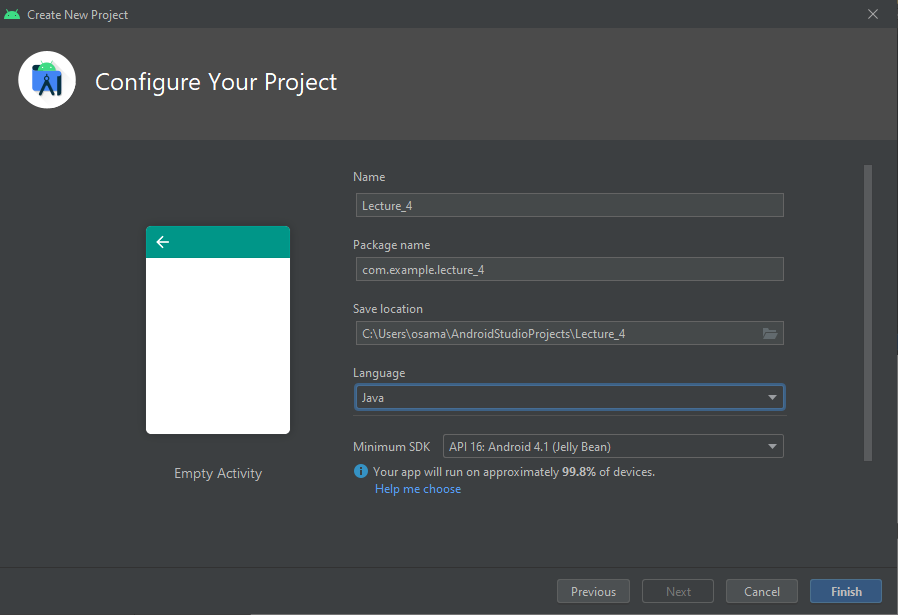
1. Select New Project



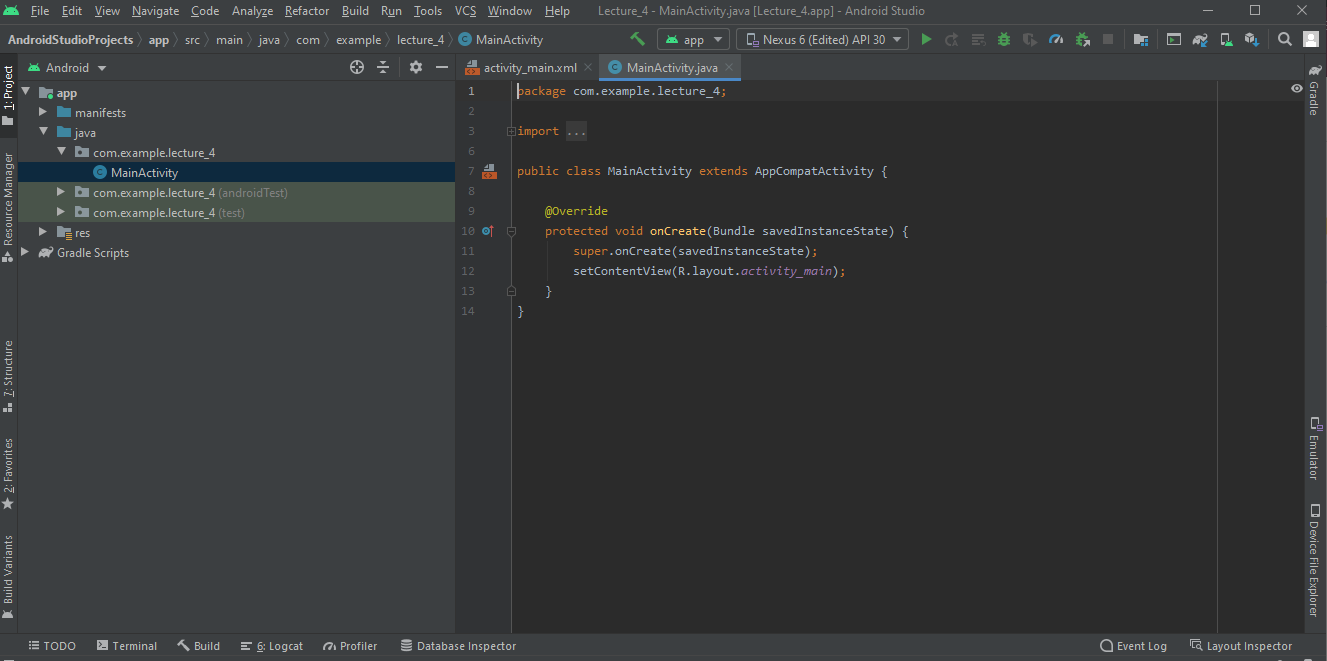
1. Select empty Activity as template of Project



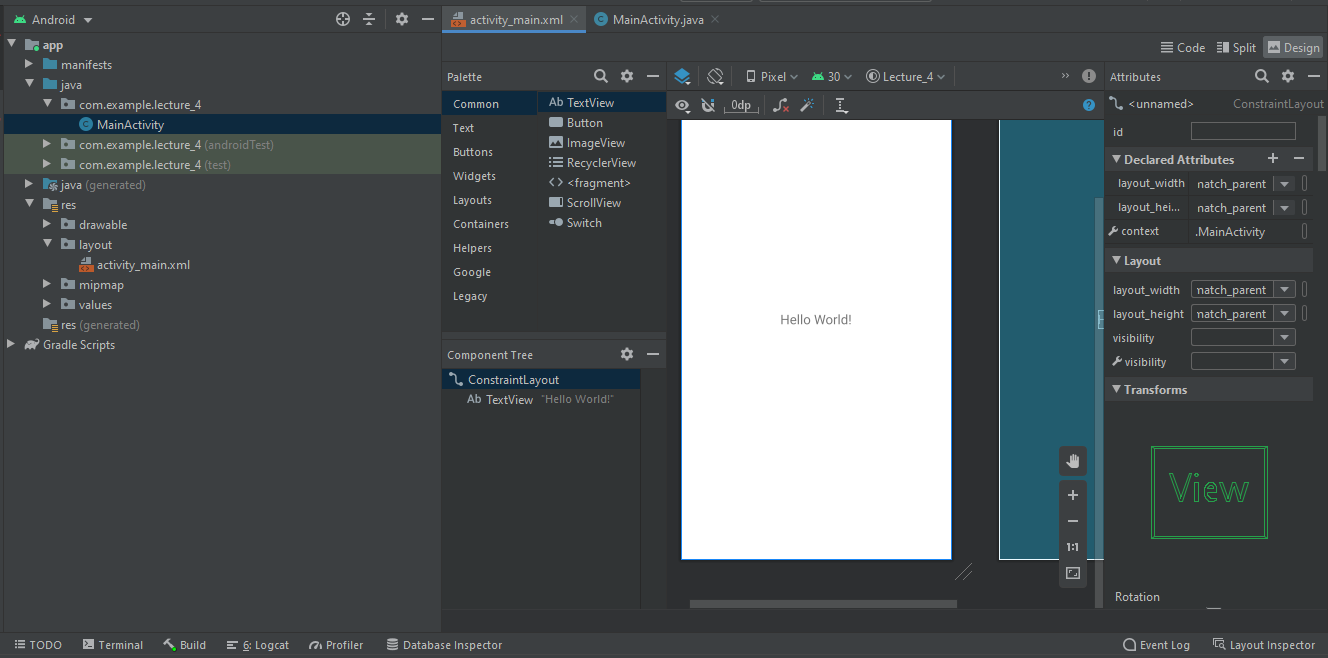
1. Give name to Project
2. Select language of Project and press finish.

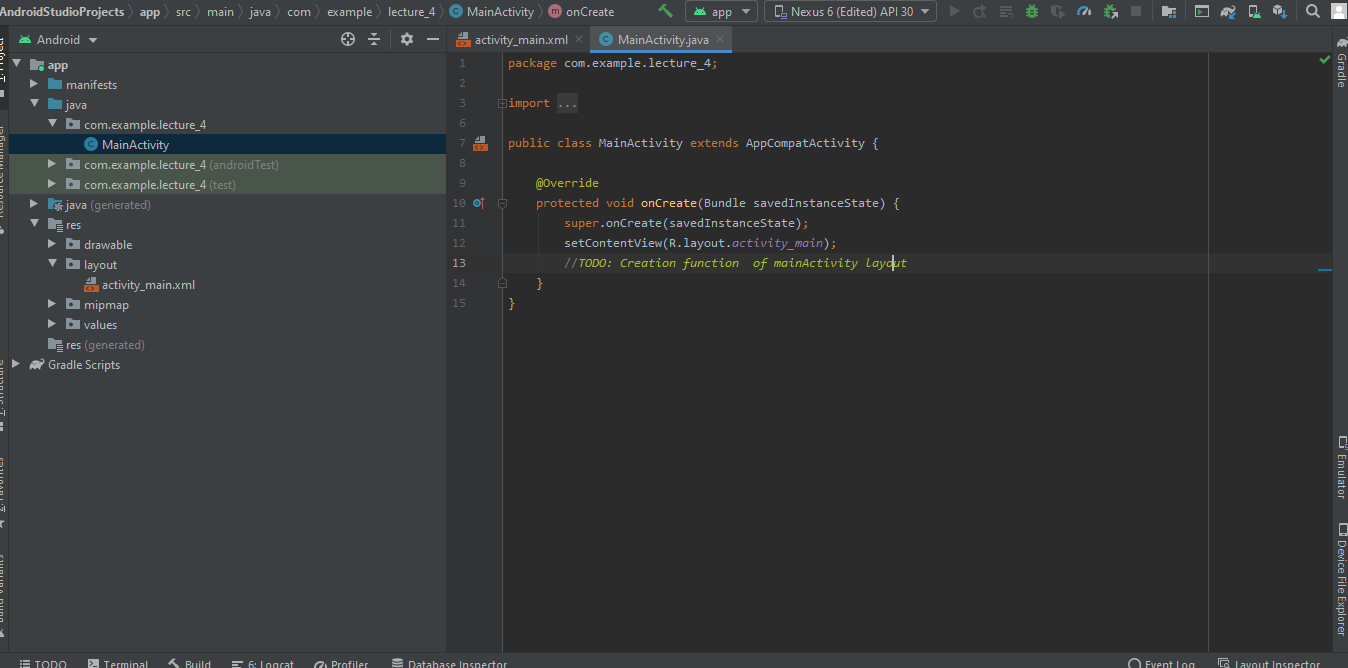


1. Wait for Empty activity to start.

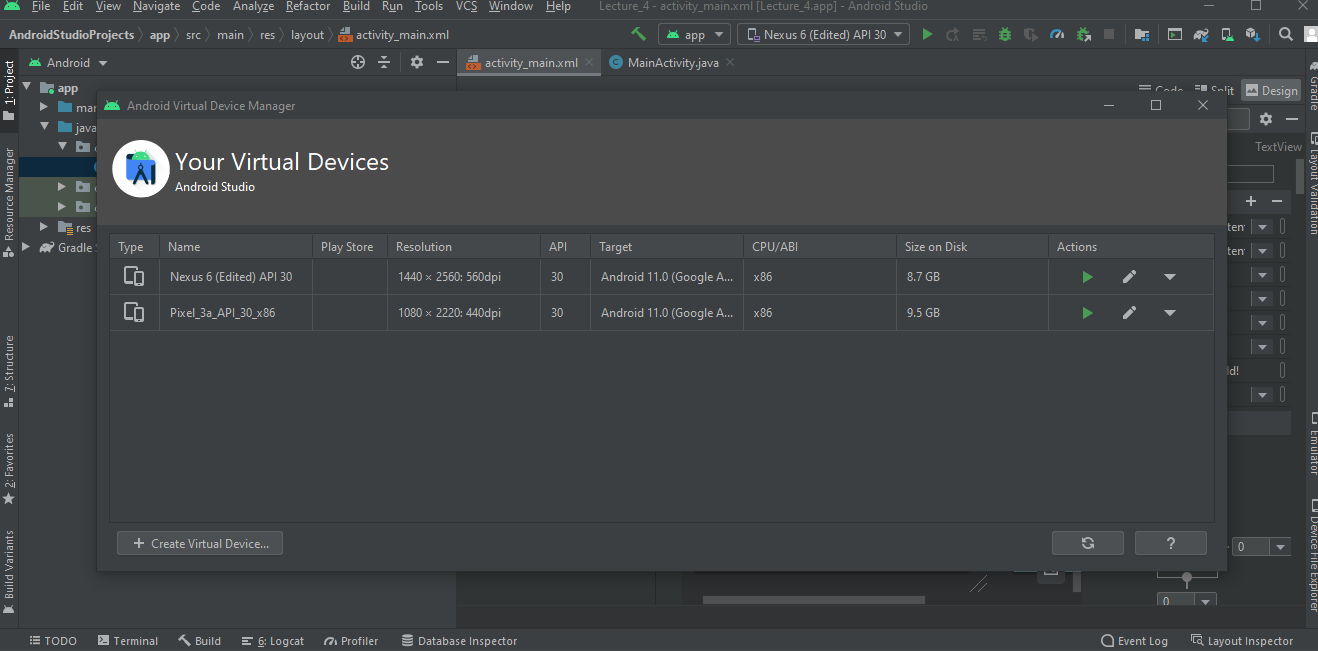


1. You can Add anything either by design it on main\_activity.xml design or directly add it through java code





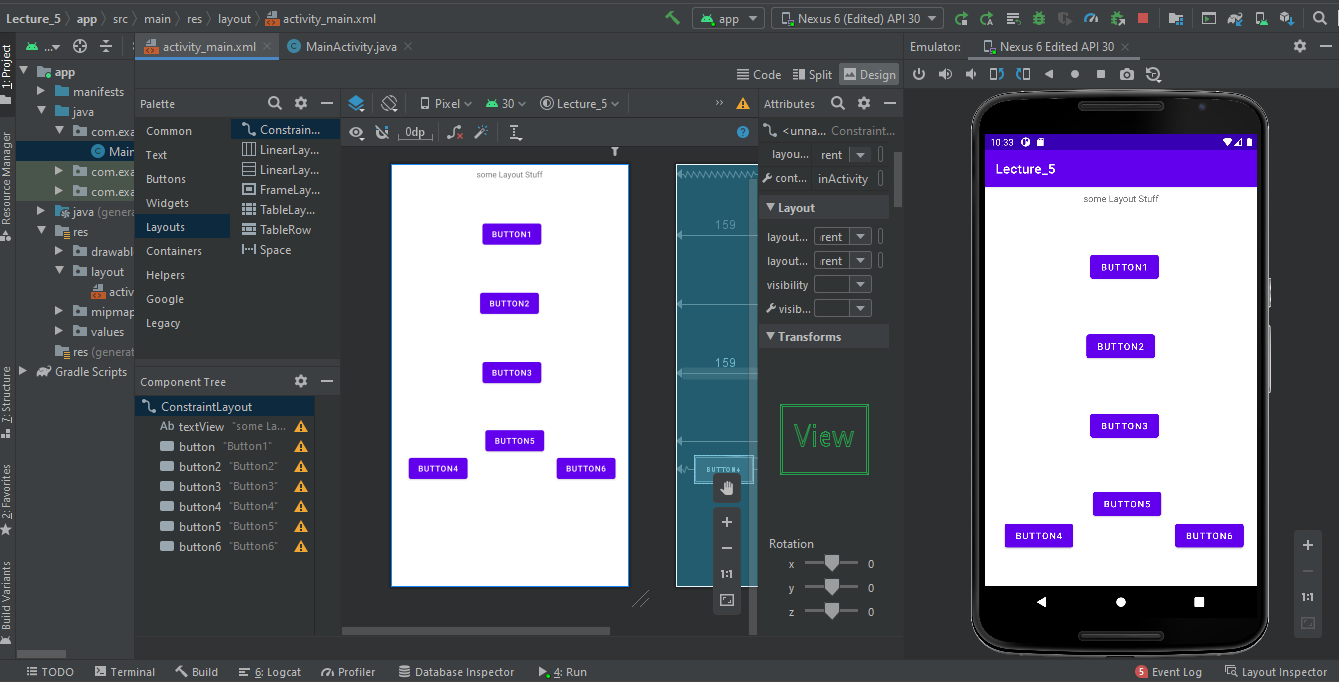
1. Add a virtual device from AVD manager which can found in tool bar



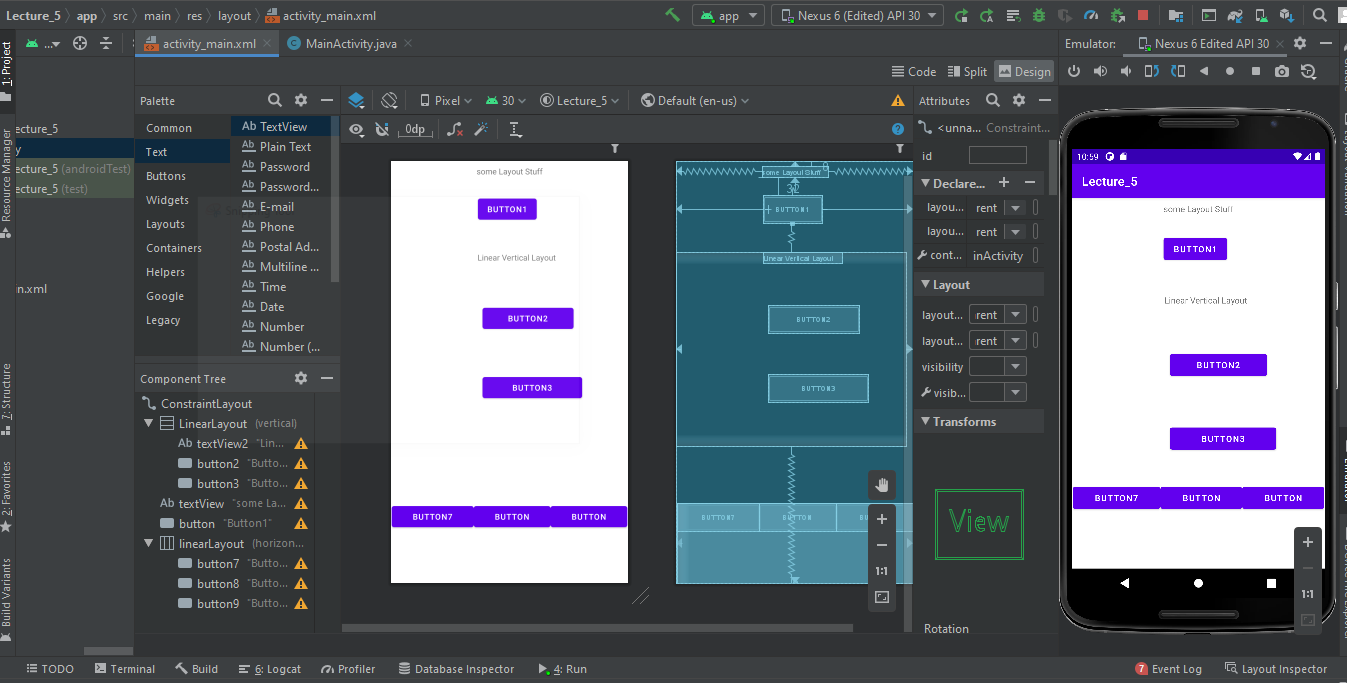
1. Now you can select any device and run the app.

**Layouts Practice**

1. Add some buttons
2. Add a vertical and Horizontal Constraint to each of button1. We can either add constrain of an element to another element or to layout itself



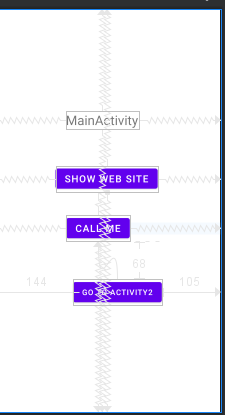
1. Adding different layouts in layout1



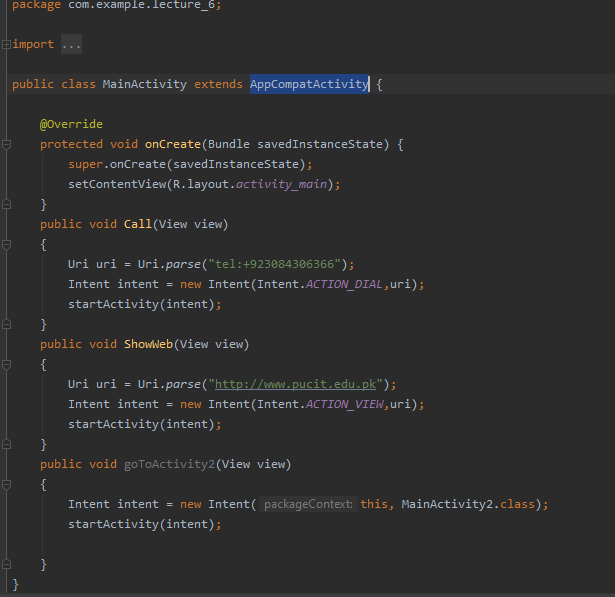
1. Add another layout in layout2
2. So, we have 2 linear layouts in constraint layout and one vertical layout in horizontal layout.

**Intent and Activities**

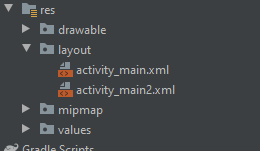
1. Add three button and text view in main activity and label them as show website, call me, gotoActivity2 and MainActivity. And apply some constrains



1. Add functions in MainActivity.java in java folder named as showweb and call to direct them to other places.



1. Create another layout in layout folder under res folder named as main\_activity2.xml



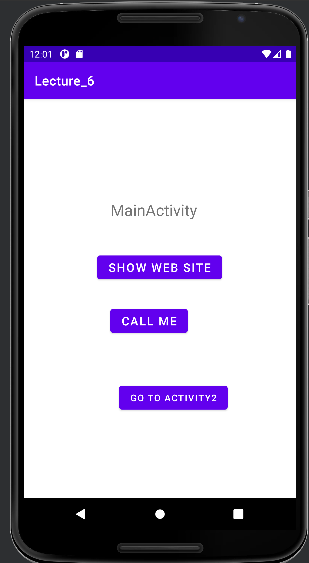
1. Similarly add three buttons and text view and add a class MainActivity2 in java folder

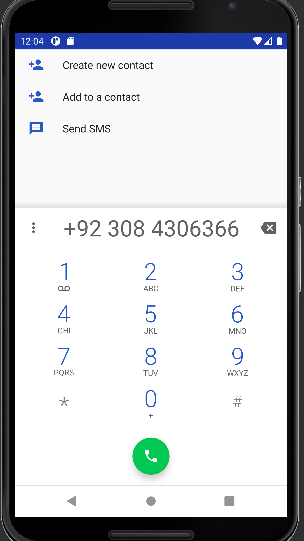


1. Add functions in MainActivity2.java



1. Run the app







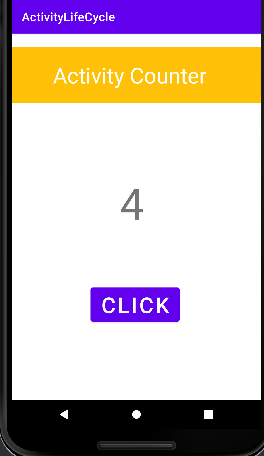
**Activity Life Cycle(Application)**

Normally when we rotate our screen (virtual Device ), a new activity is started which resets the values previously stored in that activity. But this could be removed by using the concept of Activity Life Cycle

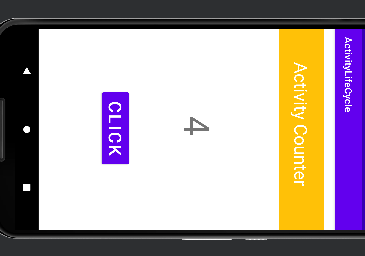
Normal Cycle of Activity





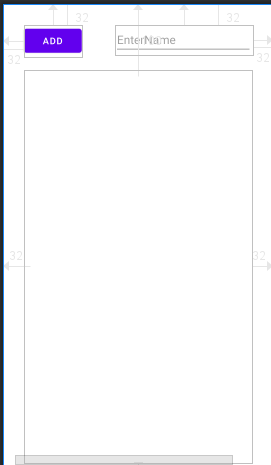


Now if we rotate then value will remain the same



**ListViews:**

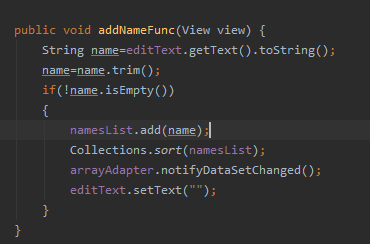
1. Add a editable text view and an button at the top in main\_activity and at bottom add a listview to show data.



1. Create an array in mainActivty.java and add some dummy data to display it.



1. Create a function to add the given data in the array. So that data coming from input field could be added in the array.



**Now lets run the app….**

