Make an interactive UI App:

In this section you will be learning how to build apps which will interact from the user i.e. take data from user and give a feedback to user.

The application will be using concept of:

1. Layout
2. TextView
3. Buttons
4. Toasts

Let’s start with making of our app:

Our app should look like this:

Let’s give our App any name I have given it Higher lower game you could give it a name **Game:**

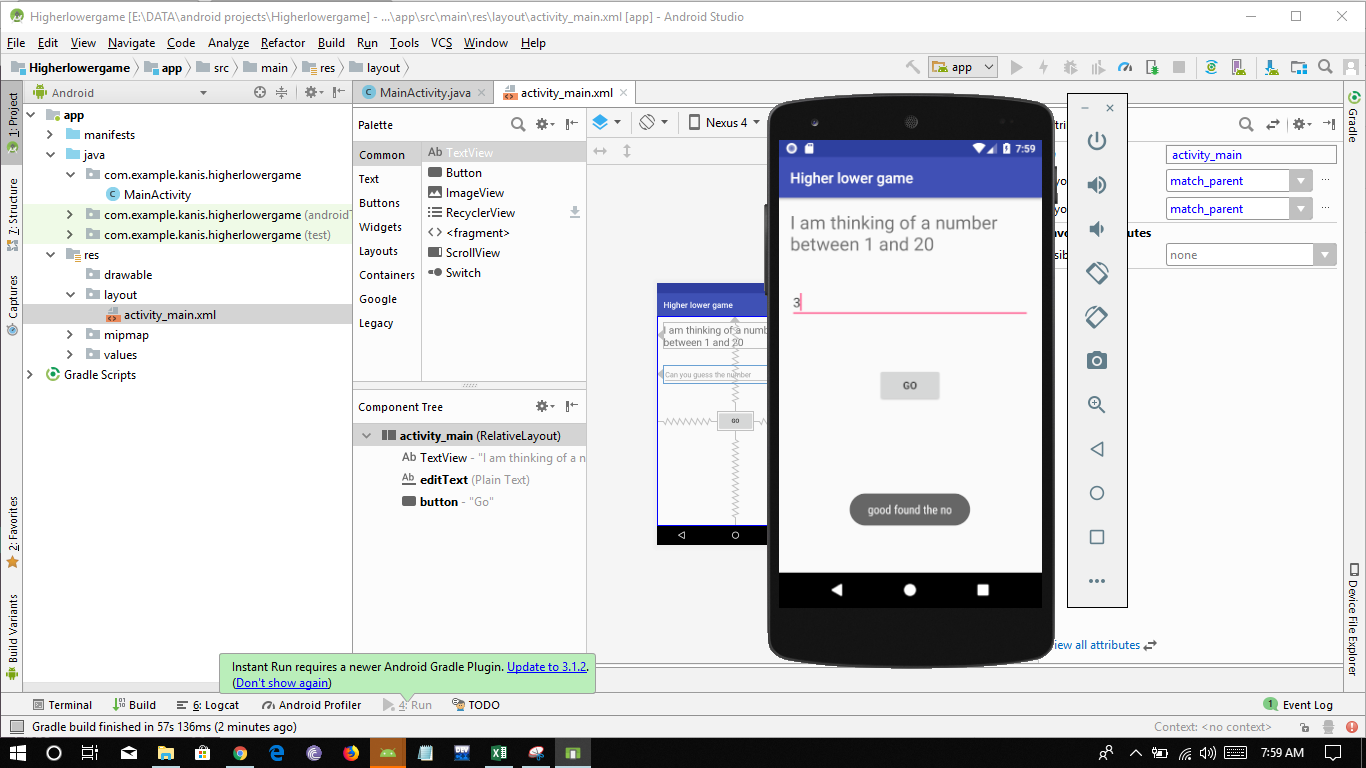
So let’s start with your Game app

About the app

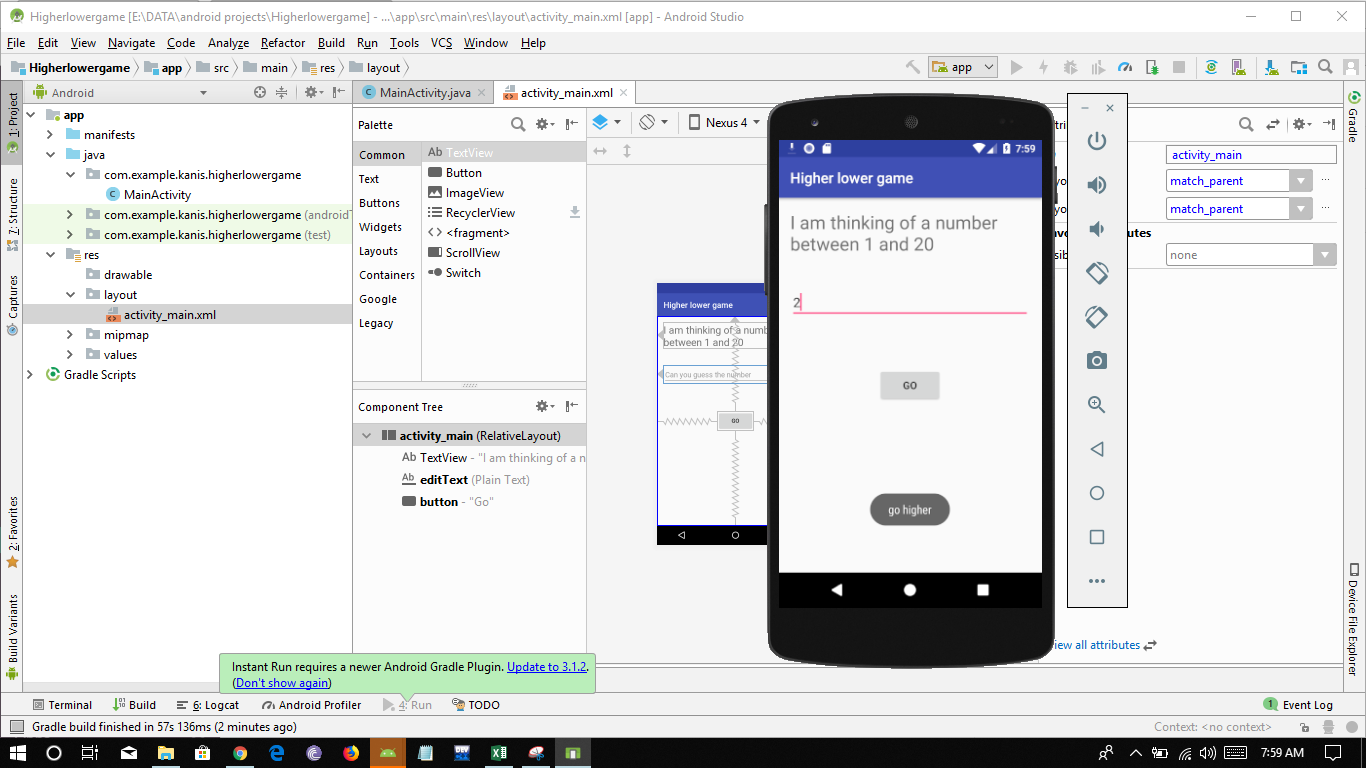
This is a simple interactive app which takes input from the user as a number between 1 to 20 and generates a random number between 1 to 20.

Which has three cases:

1. If the entered number matches with the number generated than User gets a message in form of a toast:

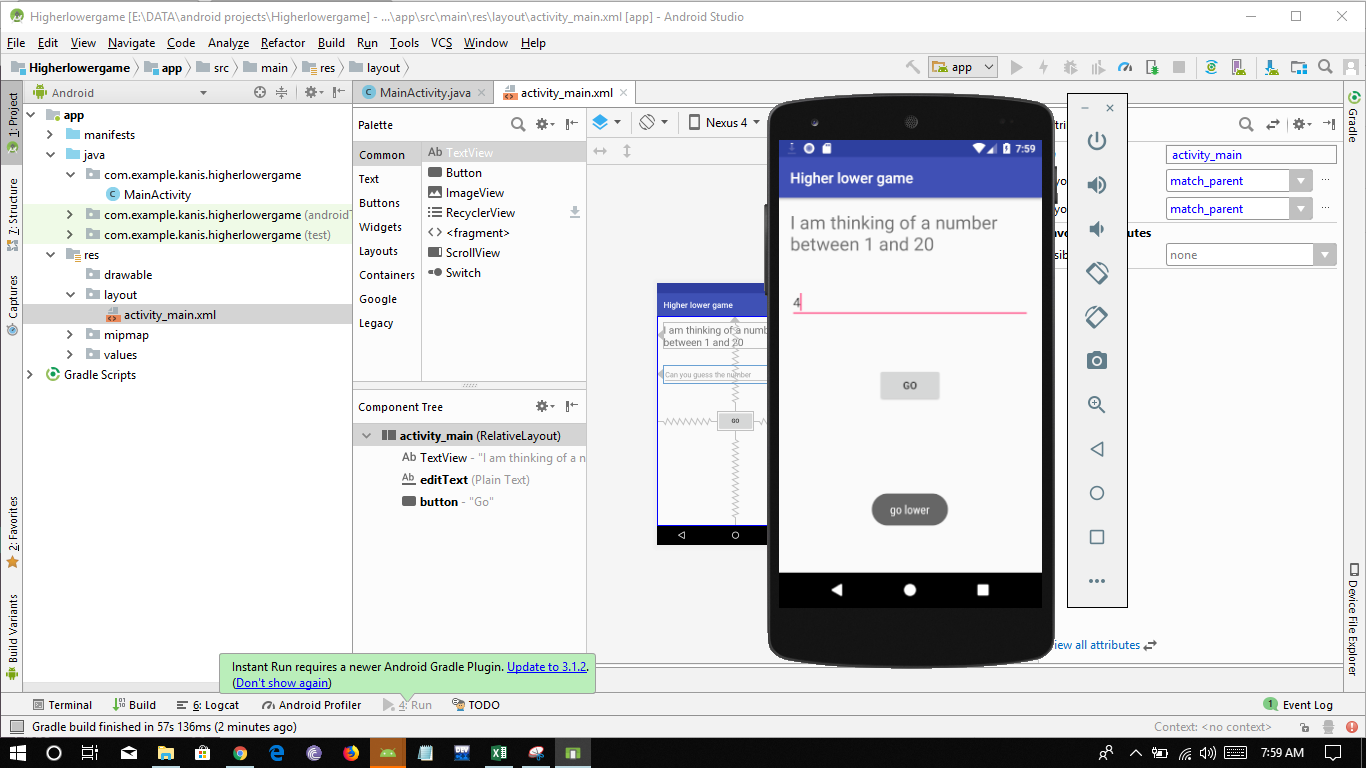
“Good found the no. “

1. Second case is if the user enters number lower than the number which is generated than user gets a message as toast:

“Go higher”

1. The third case is when user enters a number higher than the generated number than user gets a message in form of toast:

“Go lower”



Now to know the concept of random number you must know the use of function Random() in java.

Random Function:

Random function is used to generate random number either between a specific range or any random number.

This function comes under the math class.

Syntax:

<data type> variable\_name=new Random().<data type conversion>(maximum – minimum) +1;

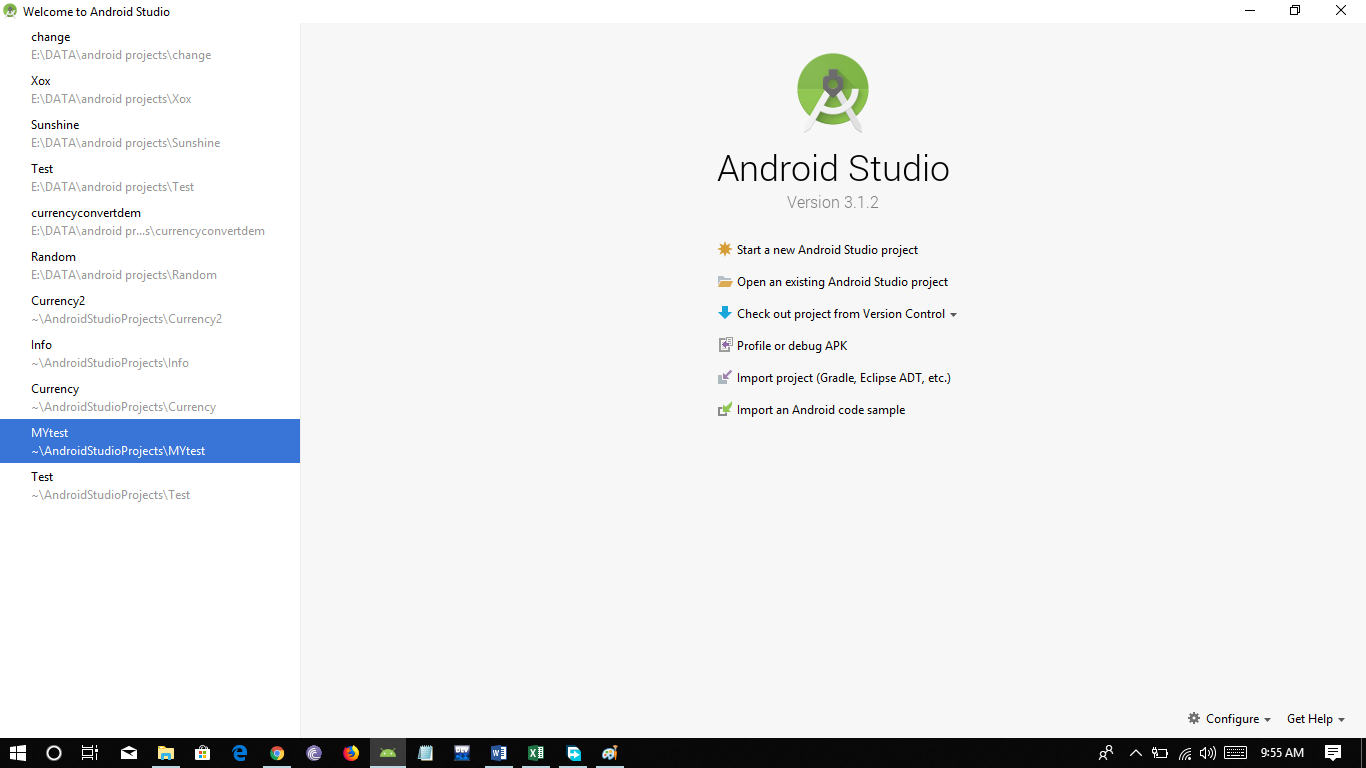
suppose we take a range of 1 to 20 of int nature than we will write:

**int rn=new Random().nextInt(19)+1;**

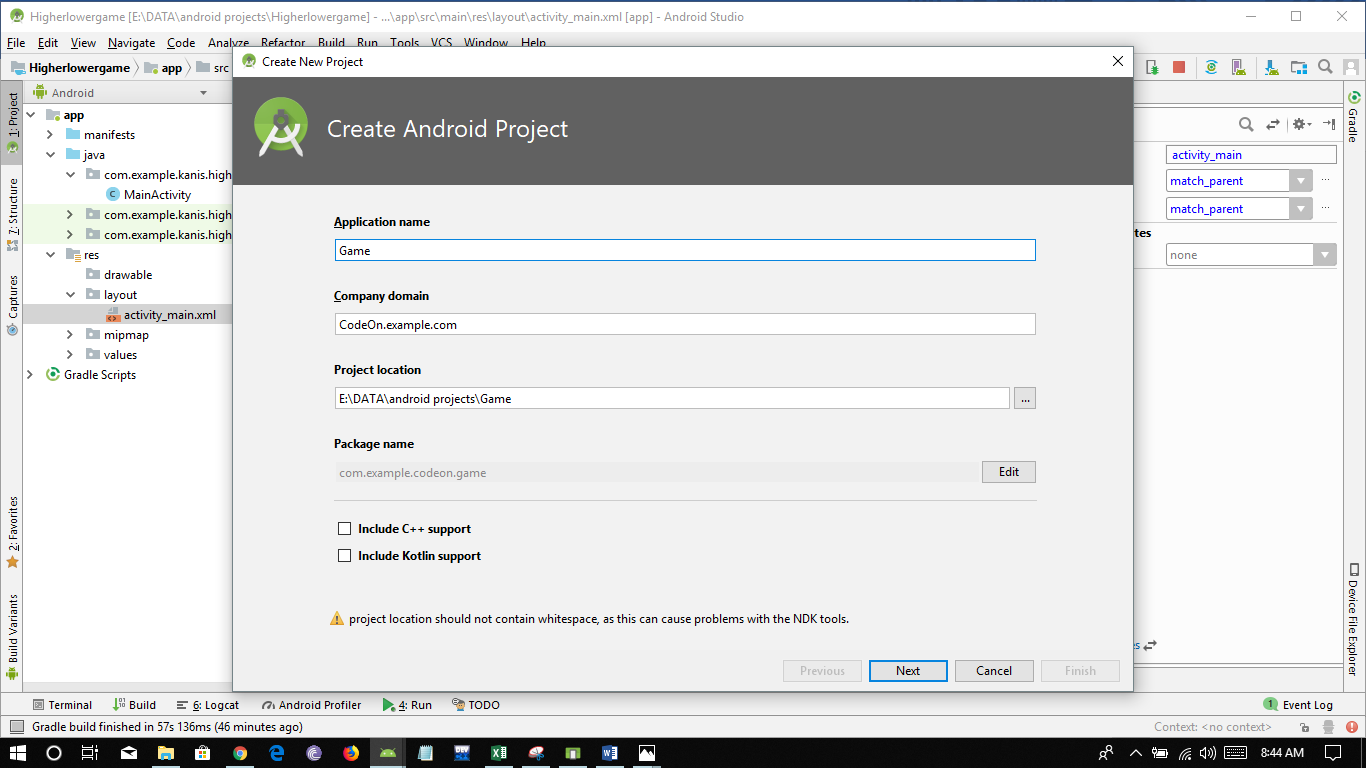
Now let’s start with building the game app

For this you first need to create a new project so first open the Android Studio Application on your pc and then click on new project option

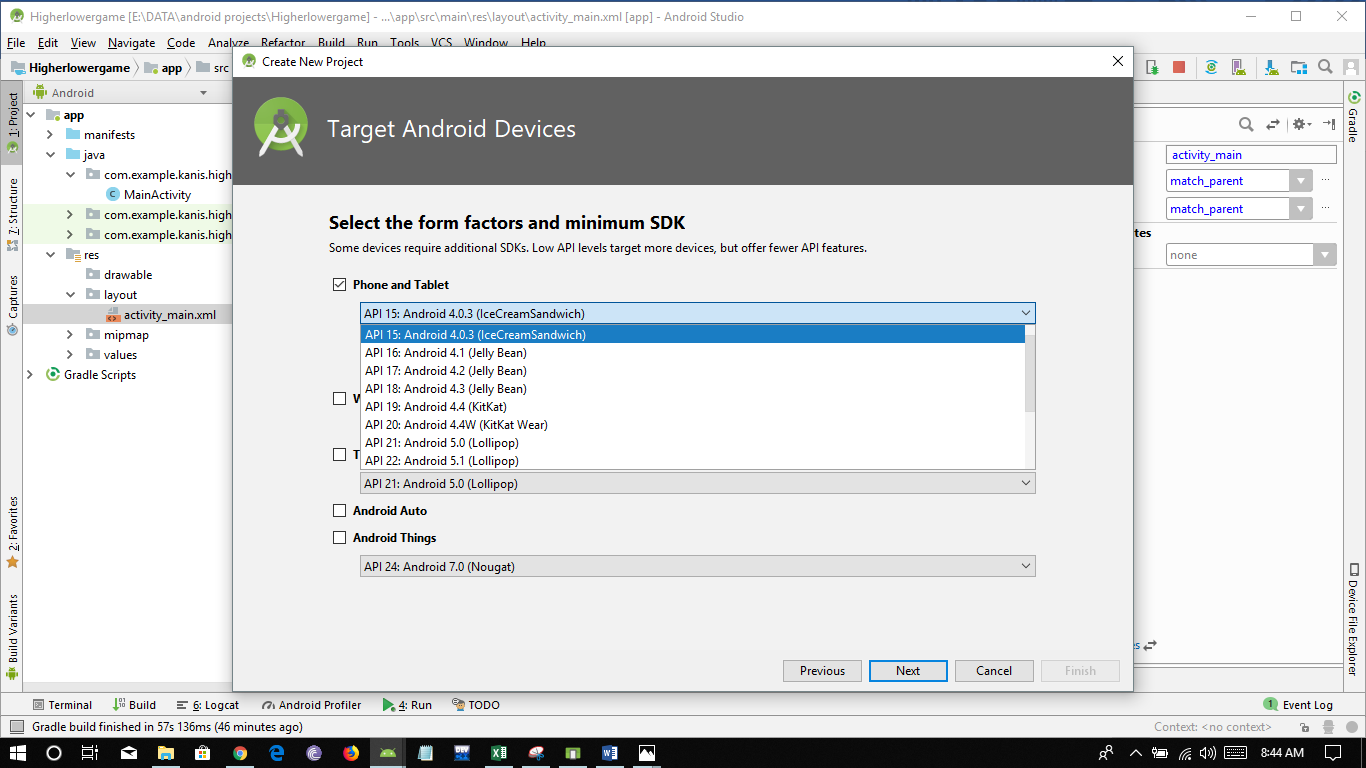
after that give your project a name.

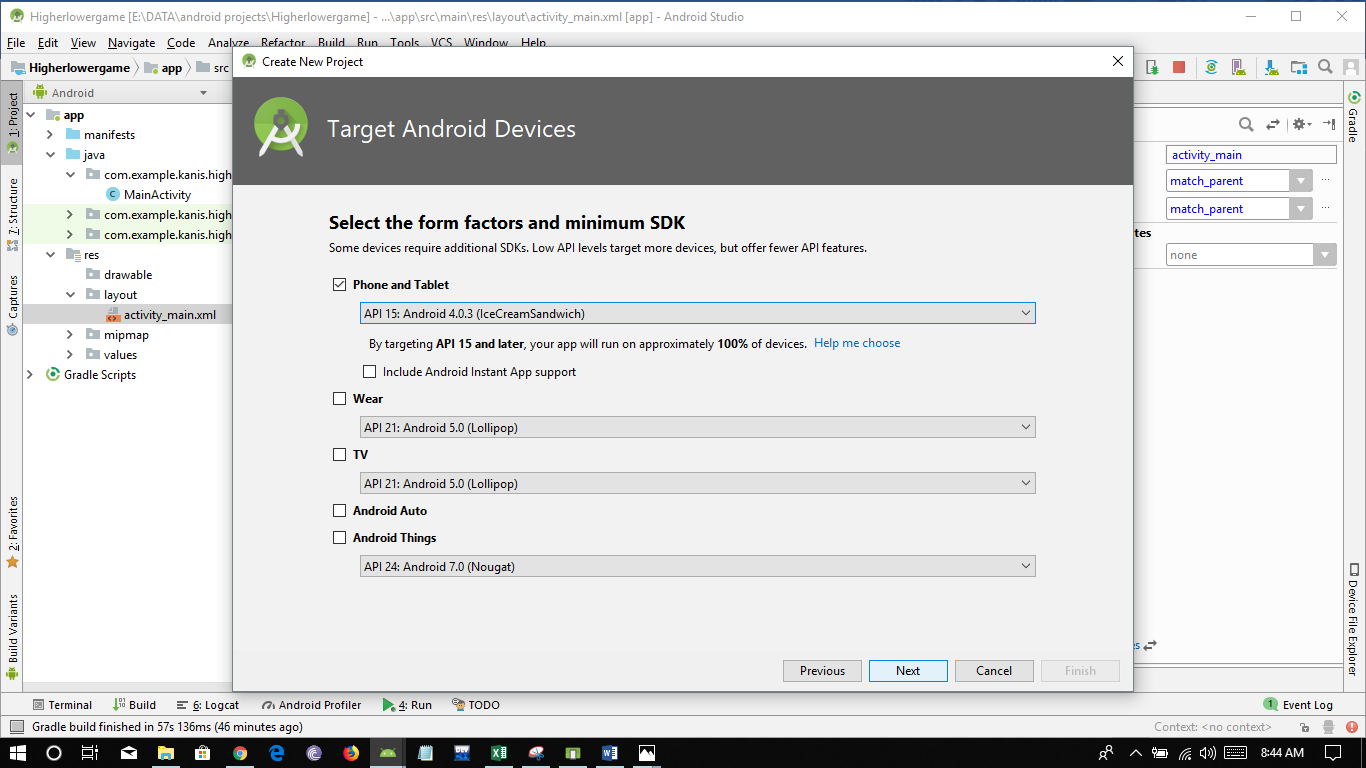
Step 1: Create a new project 

Step 2: Give your project a name “Game” of your choice and click next:



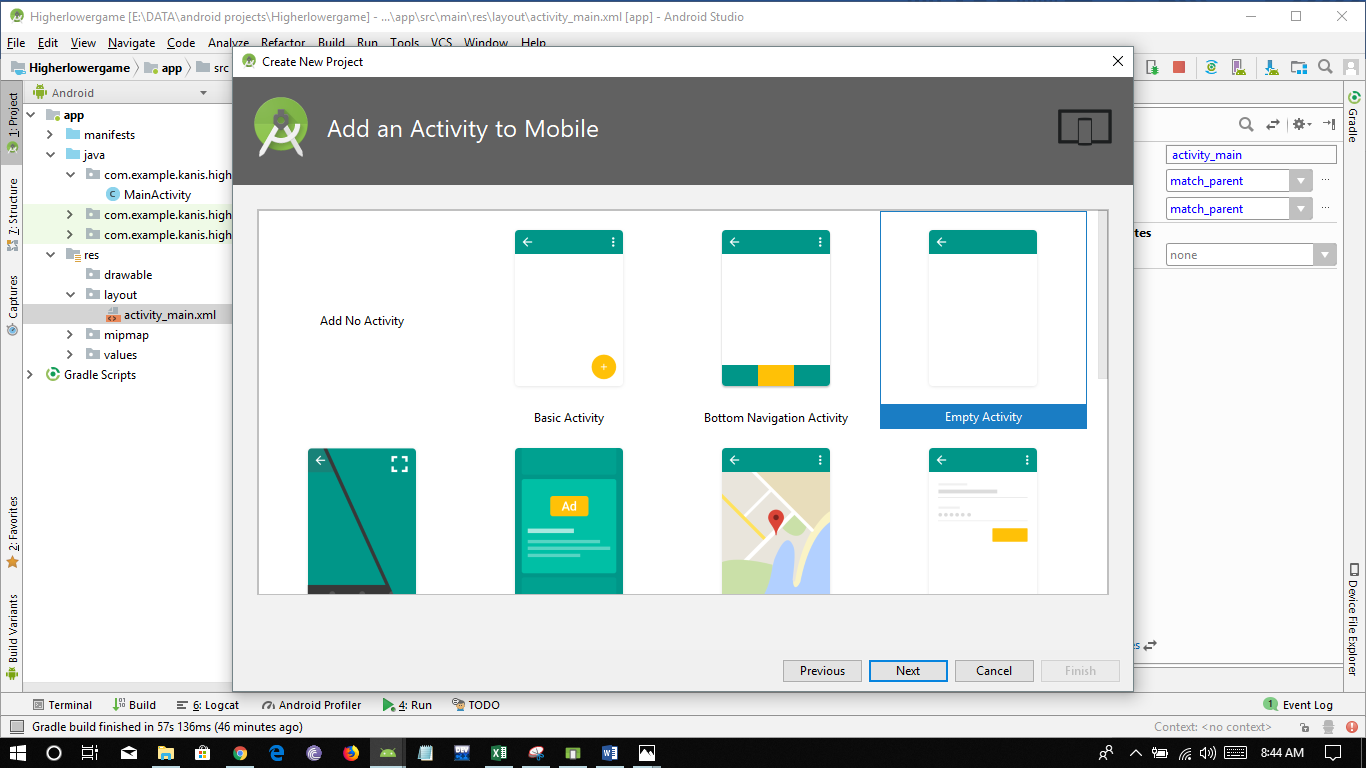
Step 3: Select the platform and desired Android Version to target by default its Android 4.0:



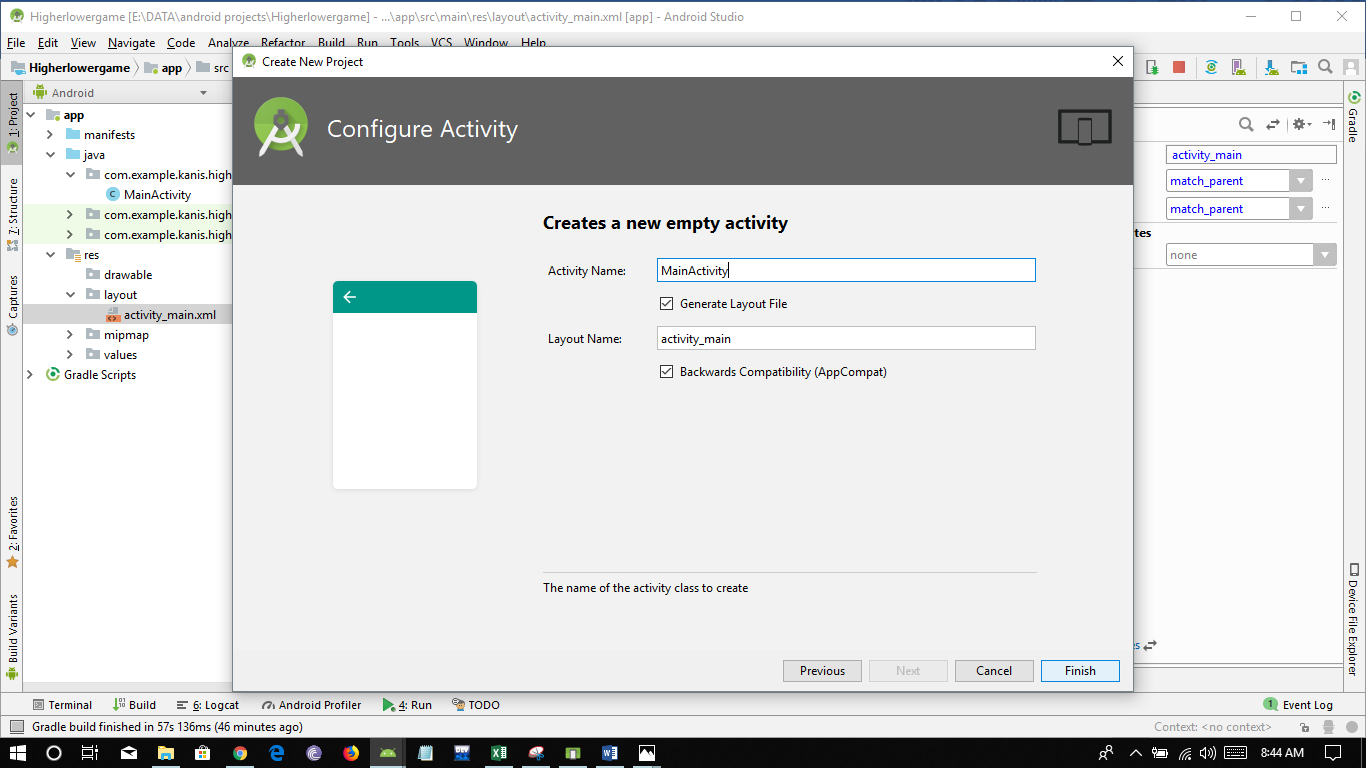


and then click next

Step 4: Select an Activity in this case select an Empty Activity then click on next:



Step 5: Finally give a name to your main activity or leave it as default then click on next button:

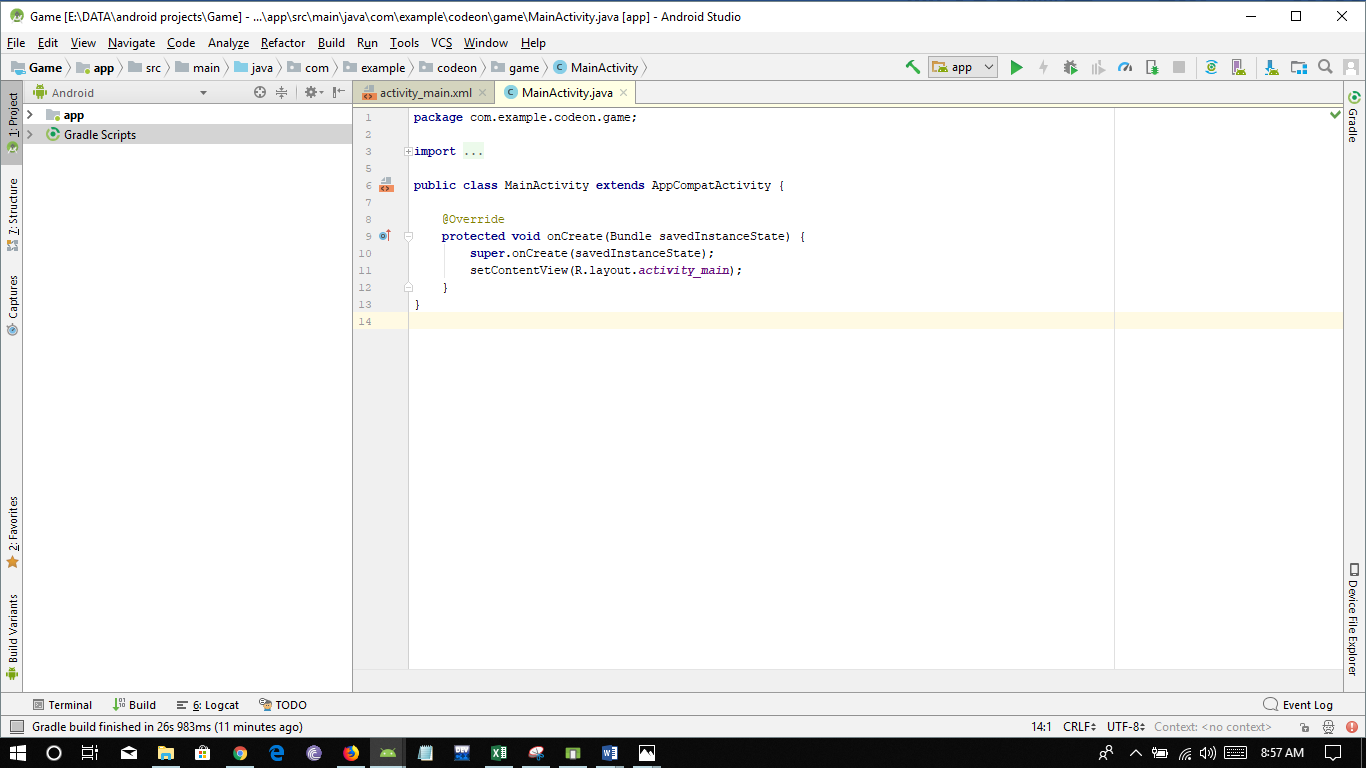


Wait for some time until the gradle and project is build.

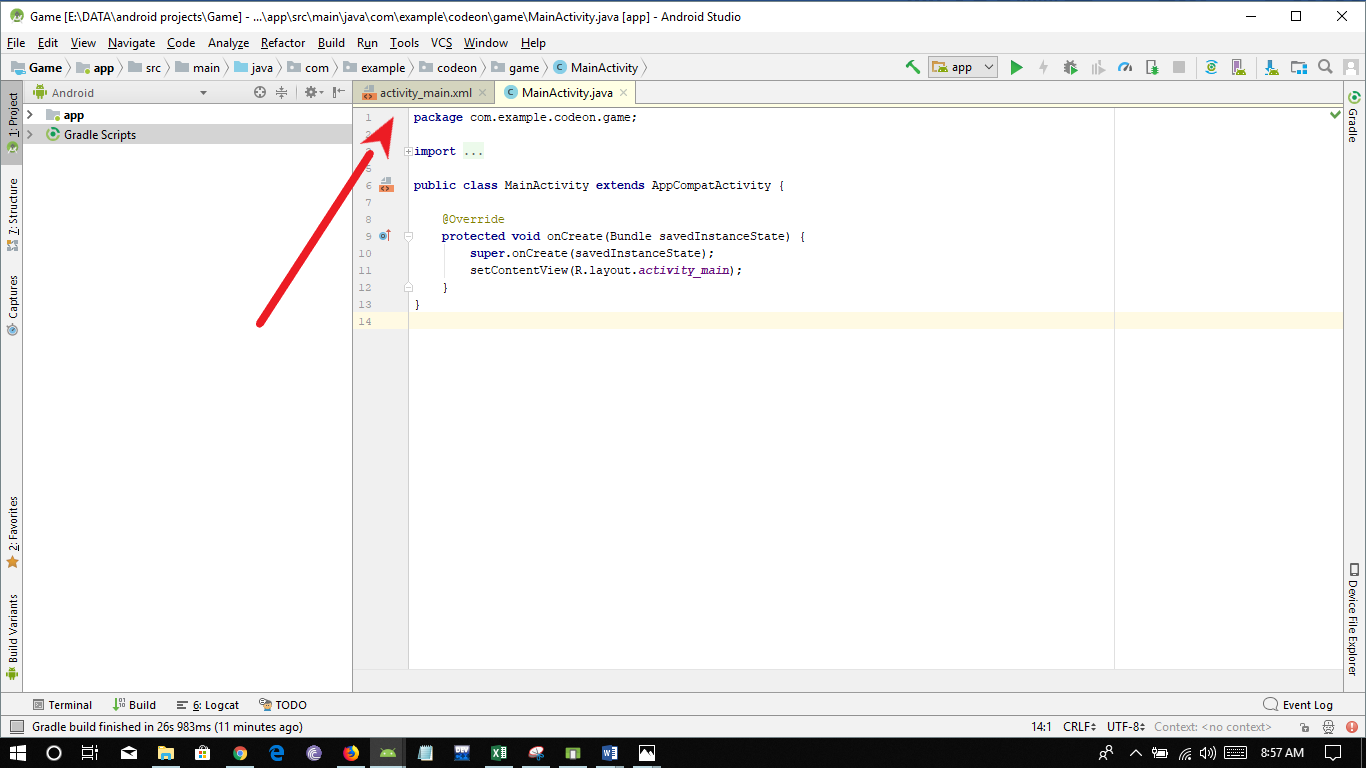
Congratulations you have successfully created a new project for building your app!!!

After your project is build your will be getting this window and MainActivity would be opened:

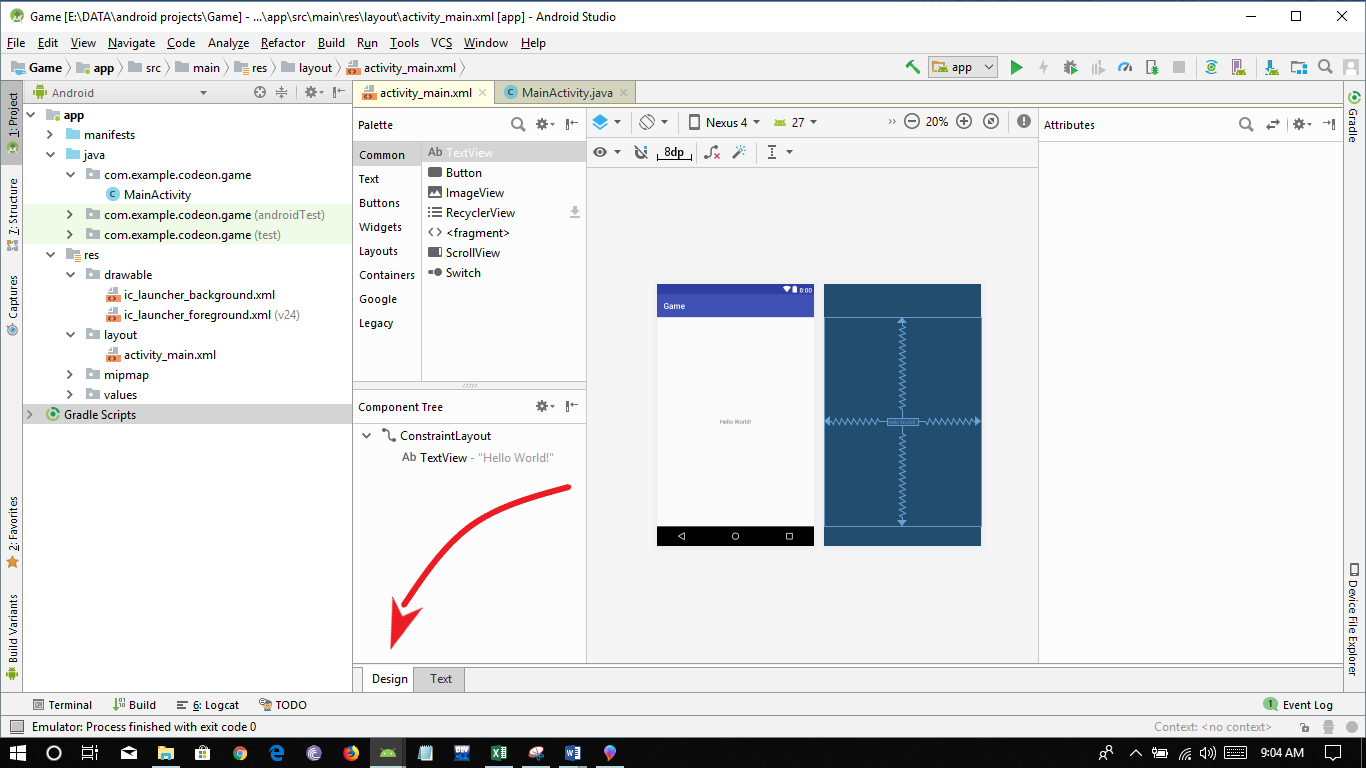
if main activity is not opened then find it under app>java> domain name> MainActivity



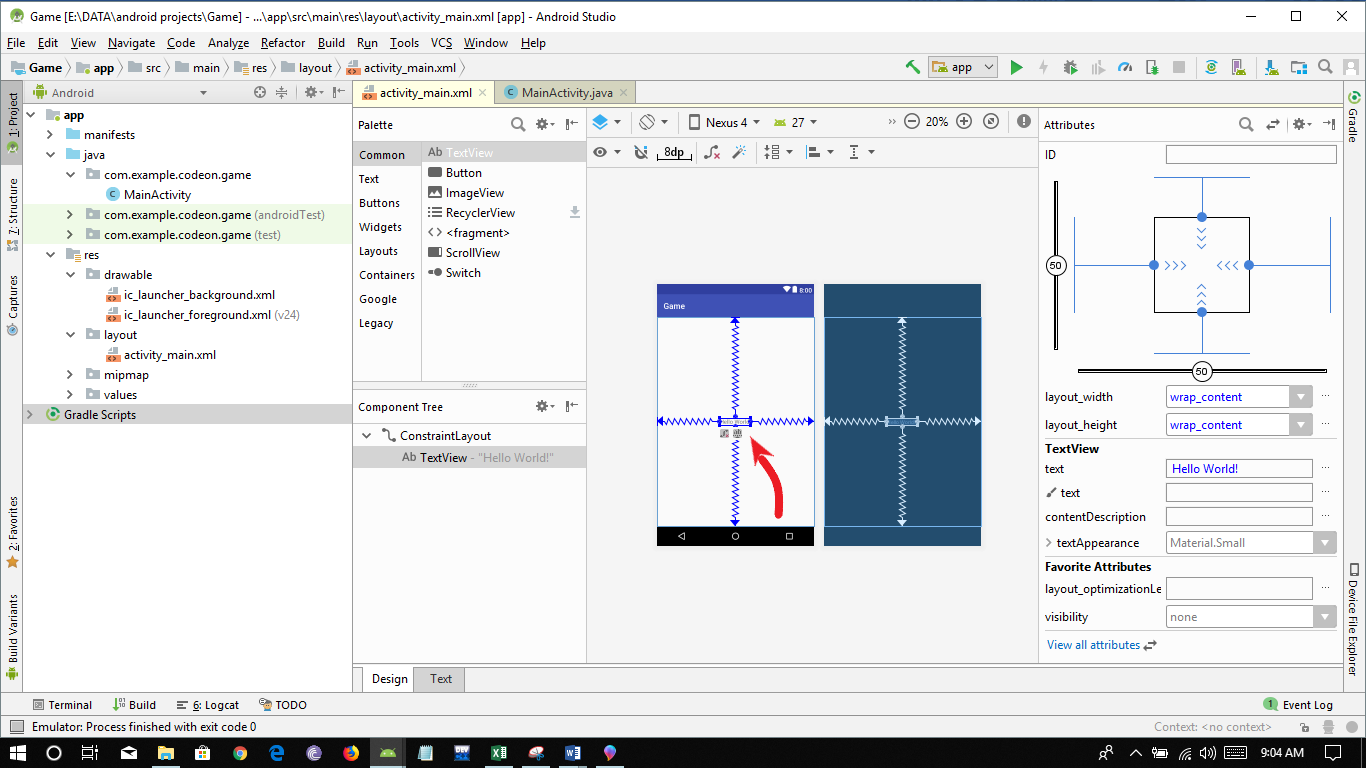
Go to activity\_main.xml tab:



After opening **activity\_main.xml** tab go to the **design** tab located at the bellow of screen:

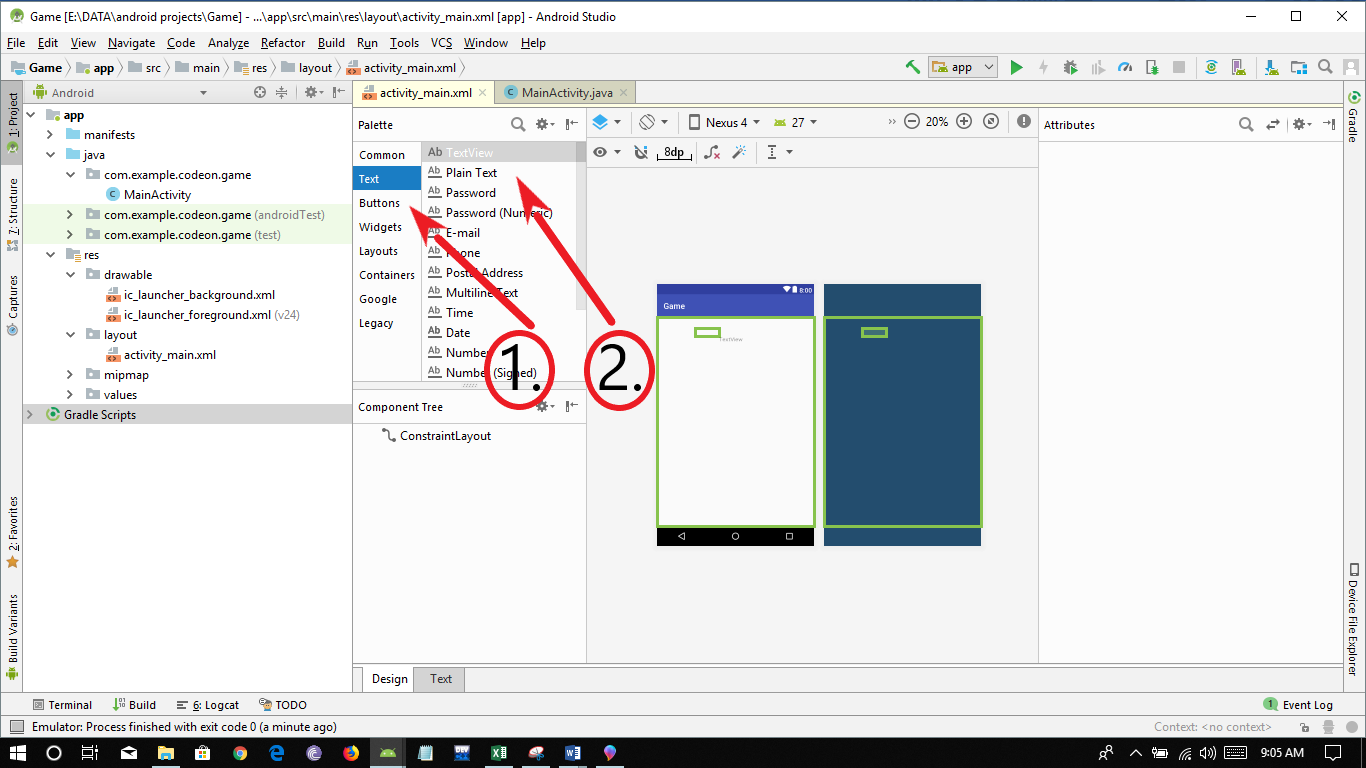


Select the Hello World Text view and then delete it by pressing the delete key.

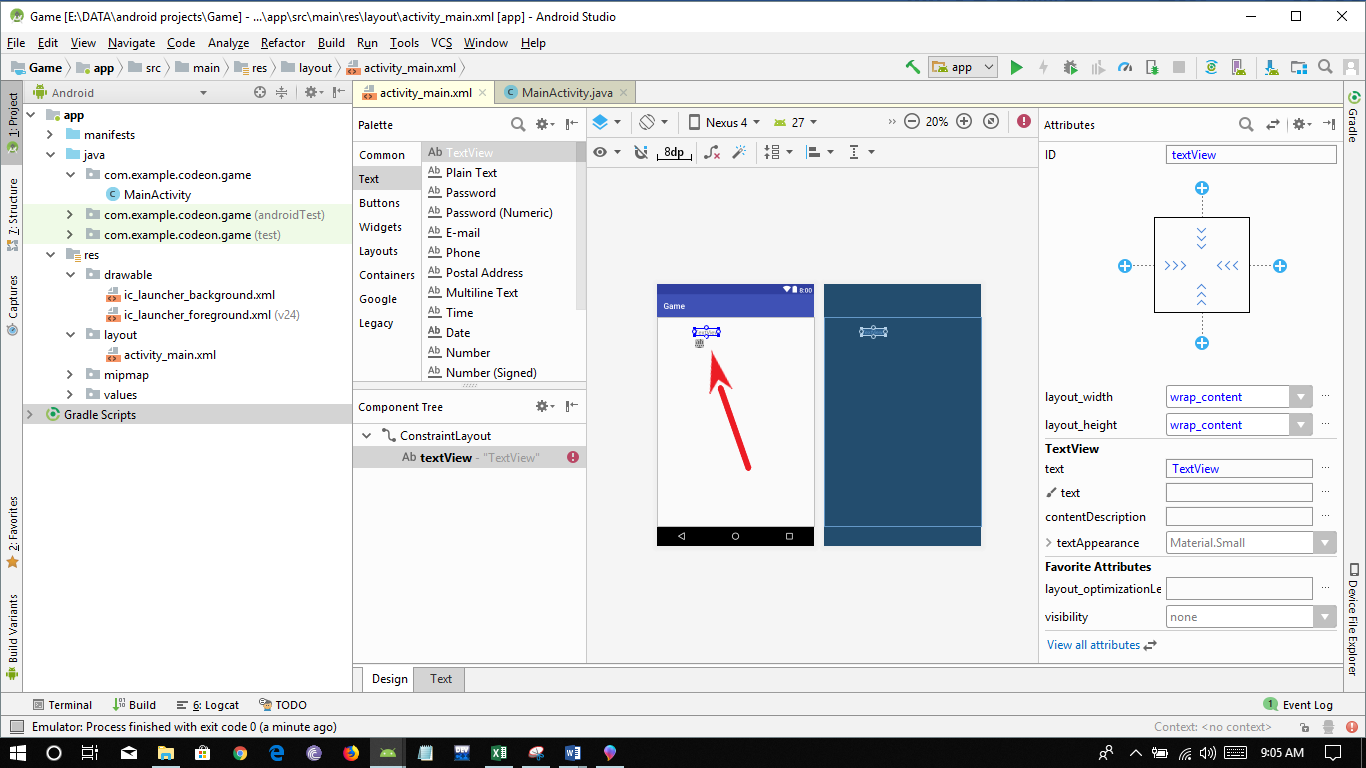


Now we need to add a text view to our app which will display about the app:

so select the TextView under Palette section as shown in the image bellow:

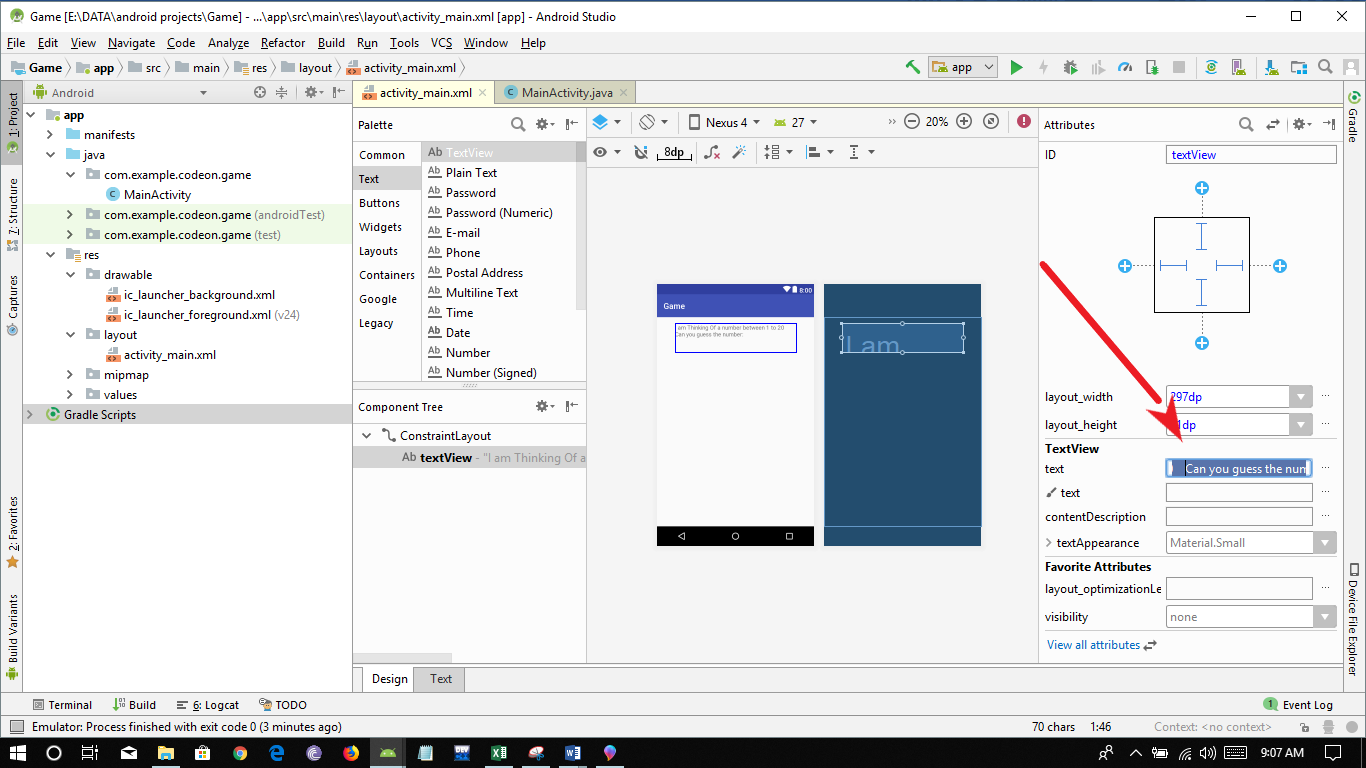


Than drag the TextView to application activity window:

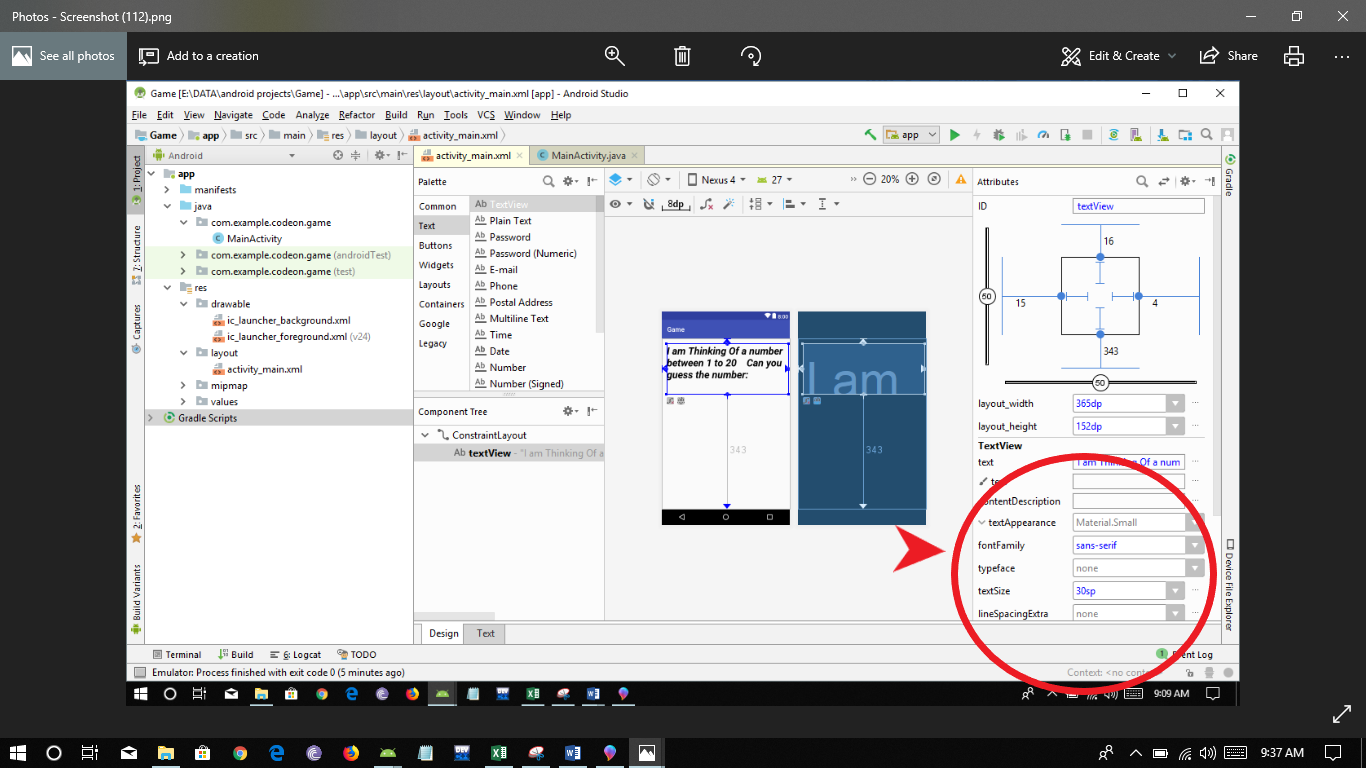


Increase the size of text view by adjusting the corners of the box.

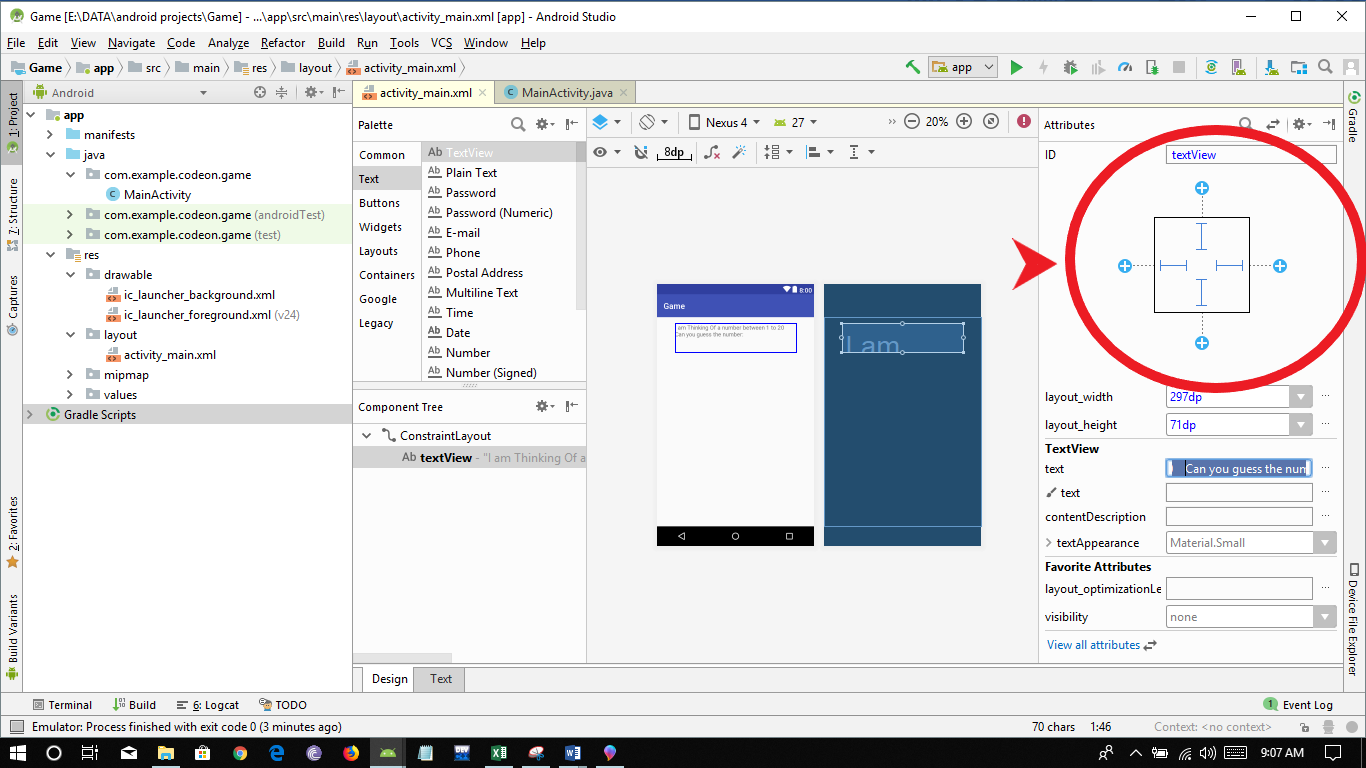
You need to display text on the window for that click on the text and type” I am Thinking of a number between 1 to 20 Can you guess the number”



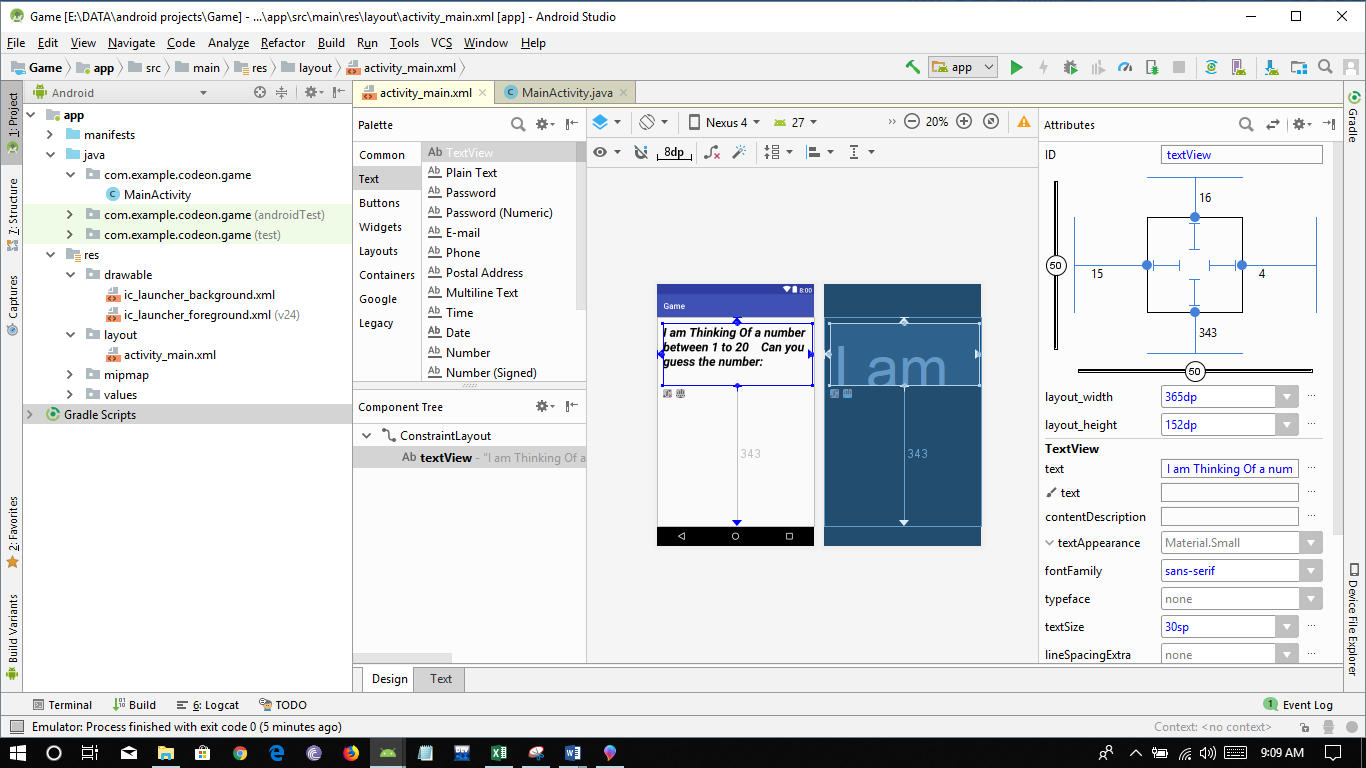
After that you need to edit the font of text for that double click on textAppearance option and then select desired font color and size of text:



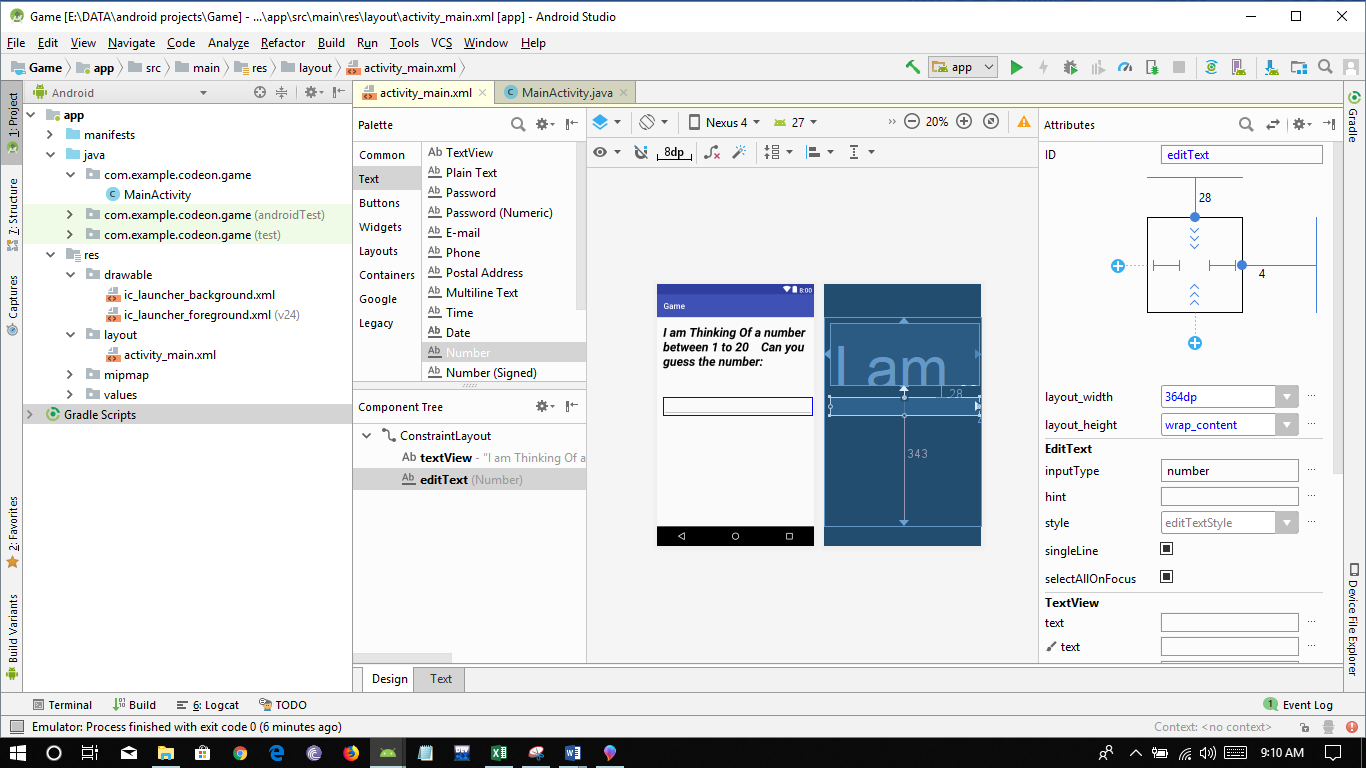
Finally adjust the position of text box by adjusting the four corners simply click on + button as shown below:



Now your App looks like this:



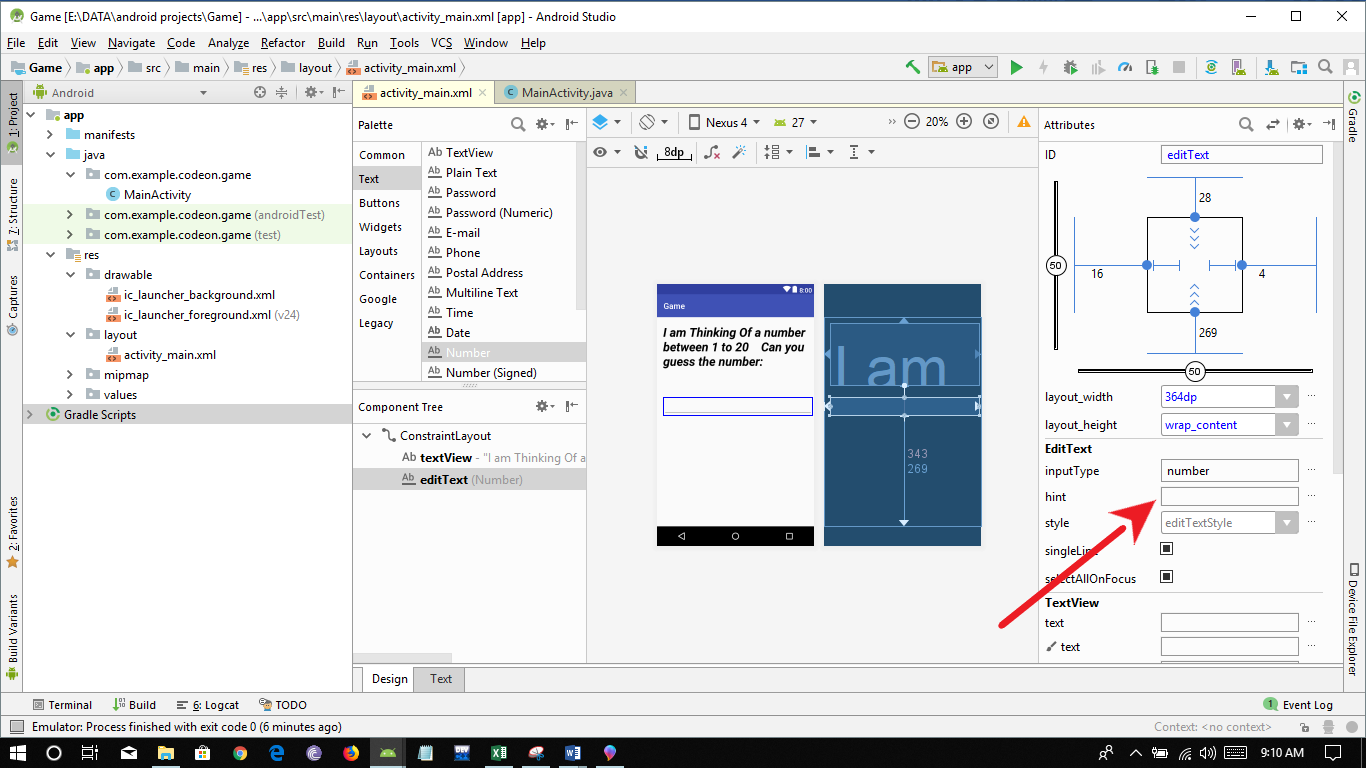
Now you need to add a text bar which will take number from user for the game:

You will find number text bar under text>Number palette same way as you have done for textView.

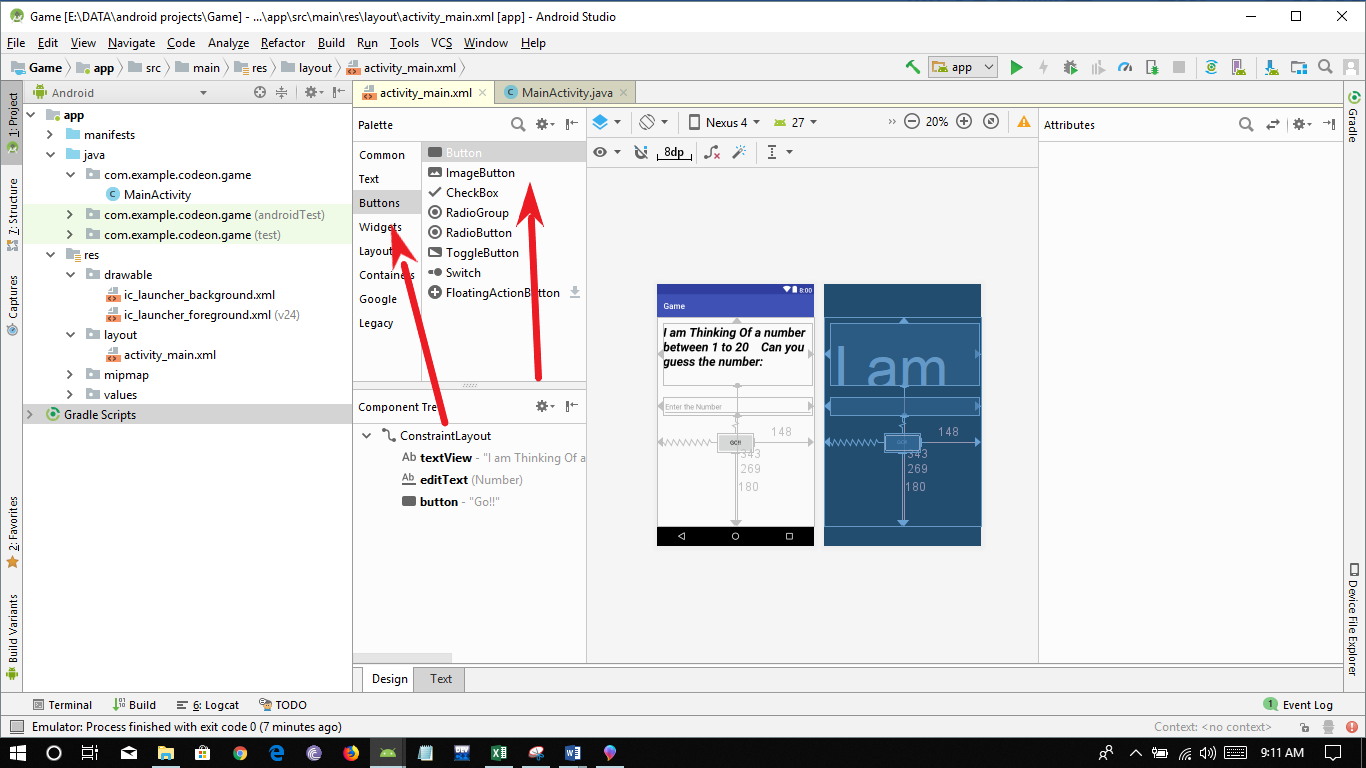
Place the number input bar below your text view and then adjust the position as you have done for textView by clicking + button for four corners.

Add a hint “Enter a number”

(Hint is basically which shows what to enter it disappears when text is input)

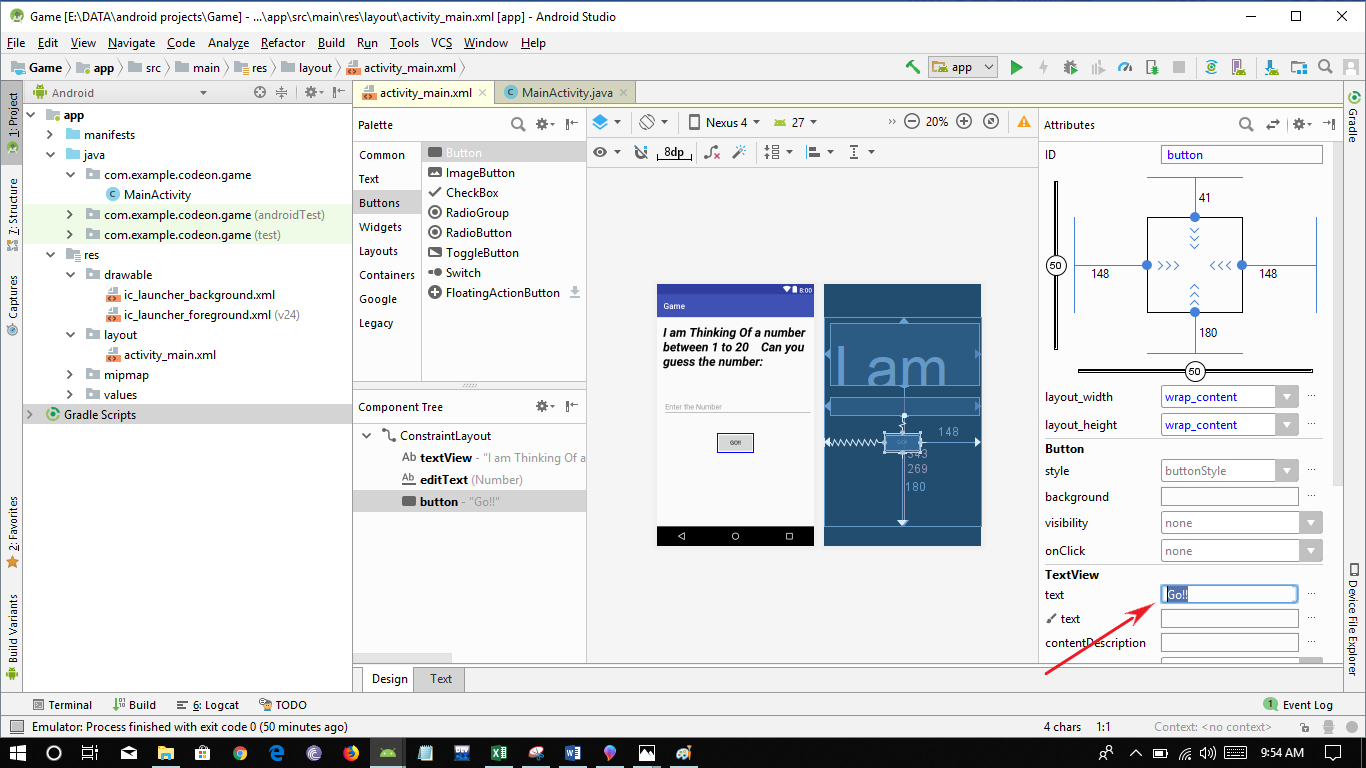


Then add a button to your app when button is pressed user will get the result.



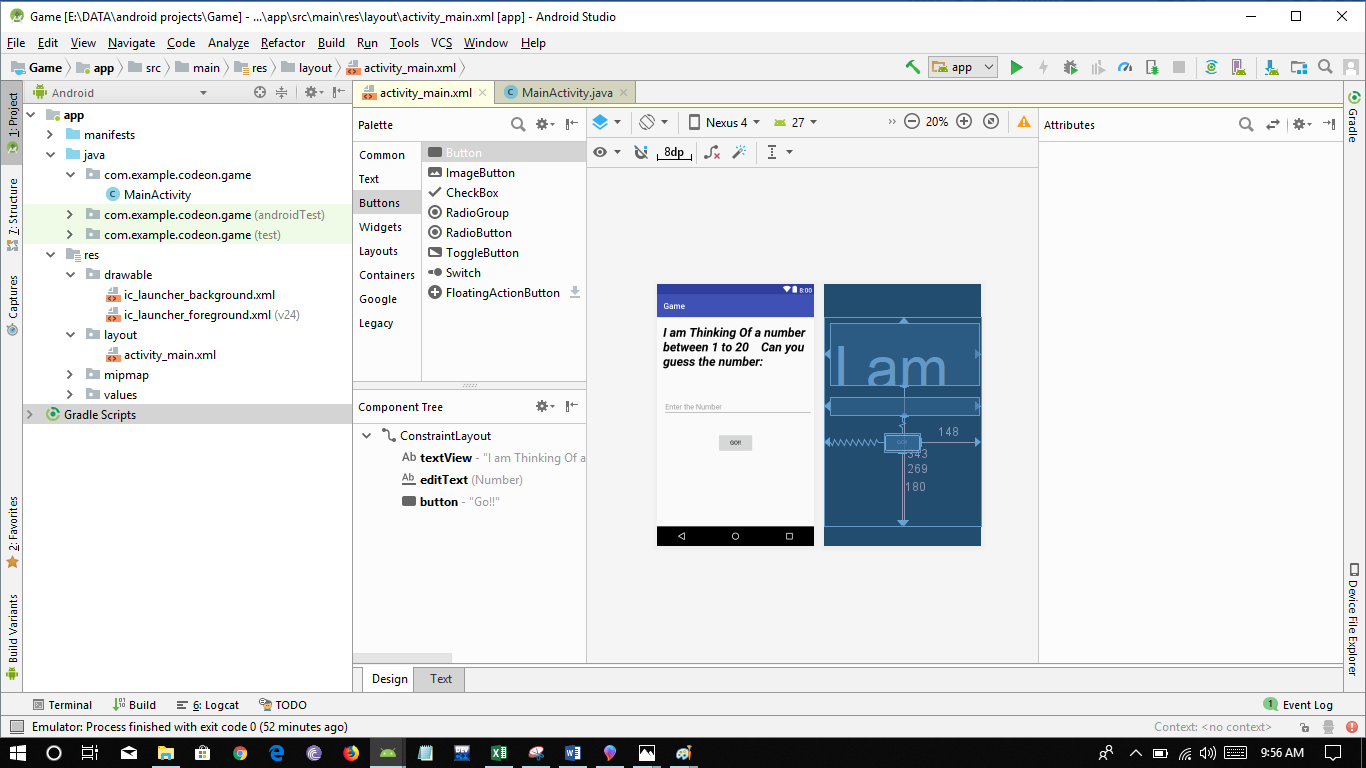
Drag the button bellow number bar and then adjust its position by clicking on + buttons as you did previously.

Give your button a name “Go!!”



Finally, you have created the interface of you app.

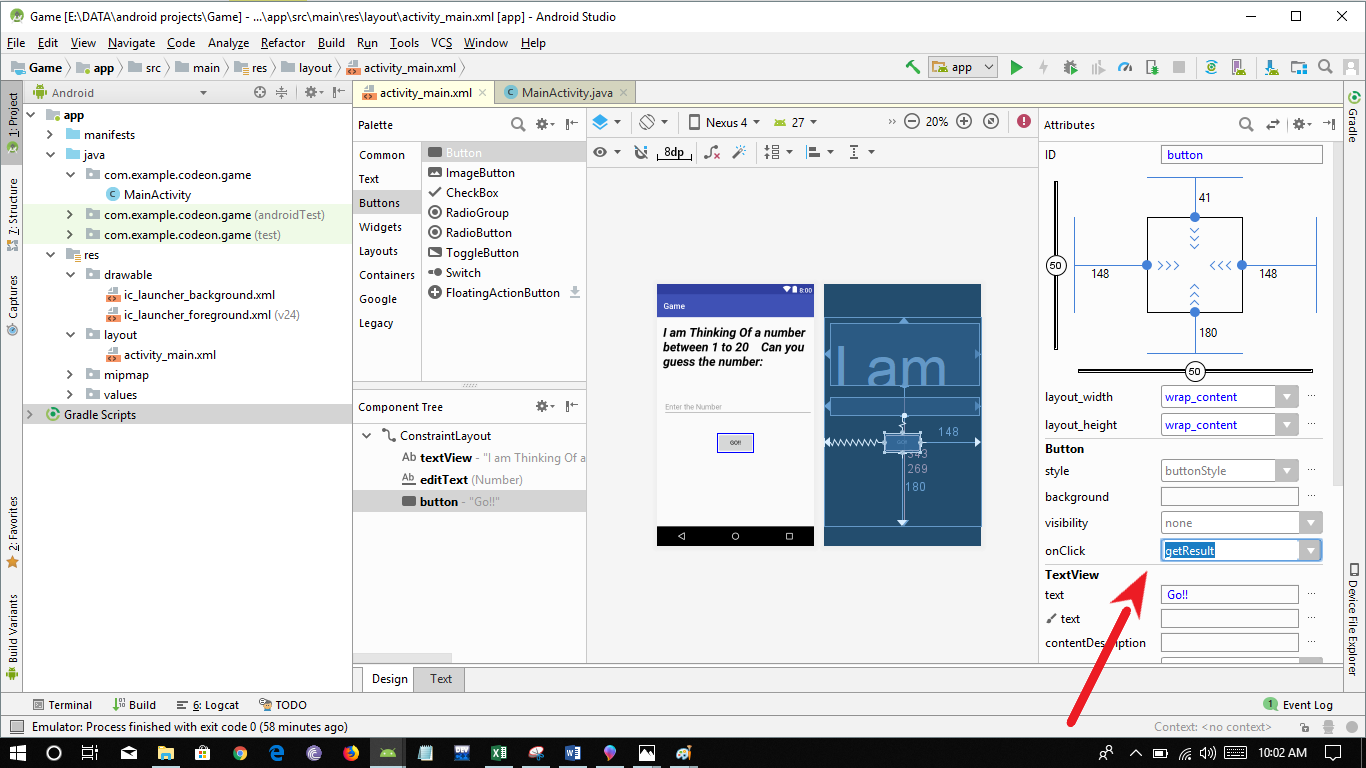
Your app will look like:



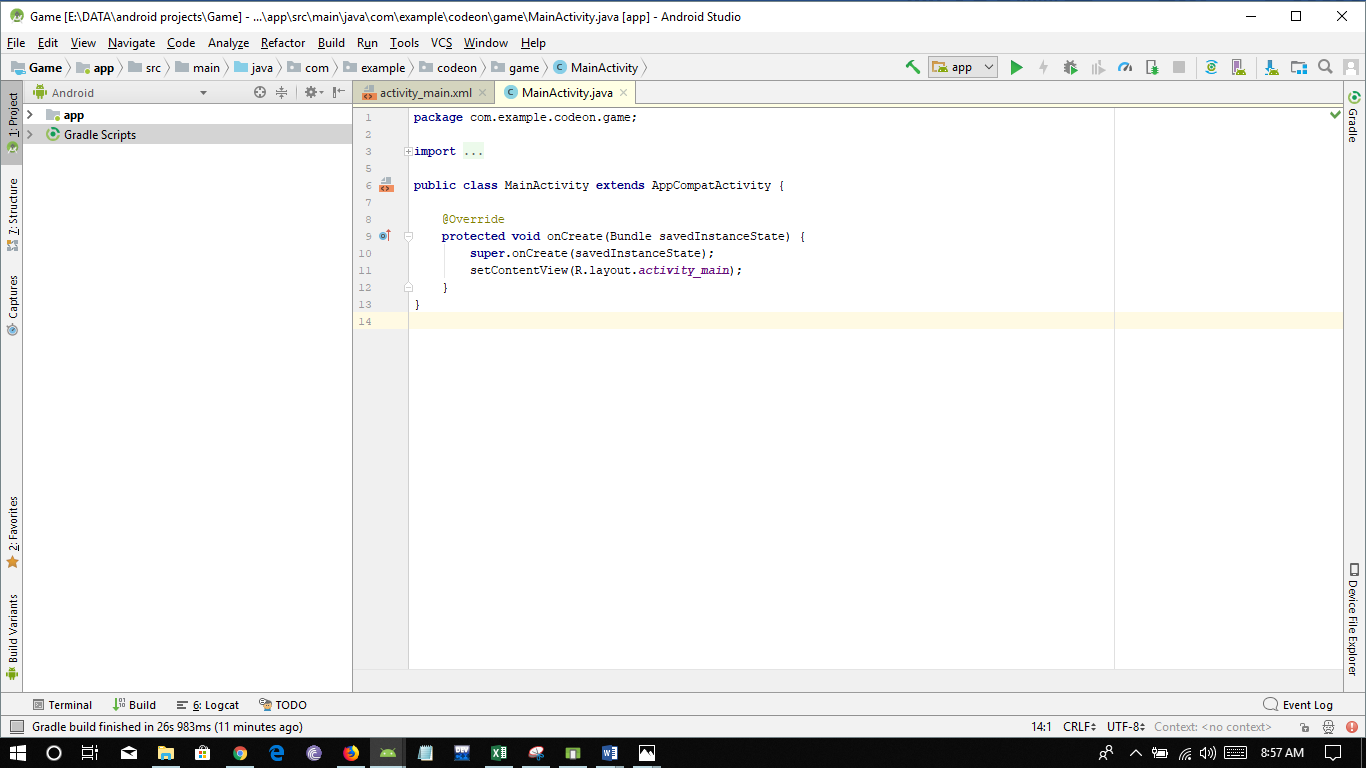
Now let’s focus on the coding part of this app:

Let the coder inside you come out and make this app……...

First you need to create a function that will work when button is clicked so that give a name to onclick option which you will find under attributes section on clicking on go button I have given “getResult”.



Now go to main activity tab



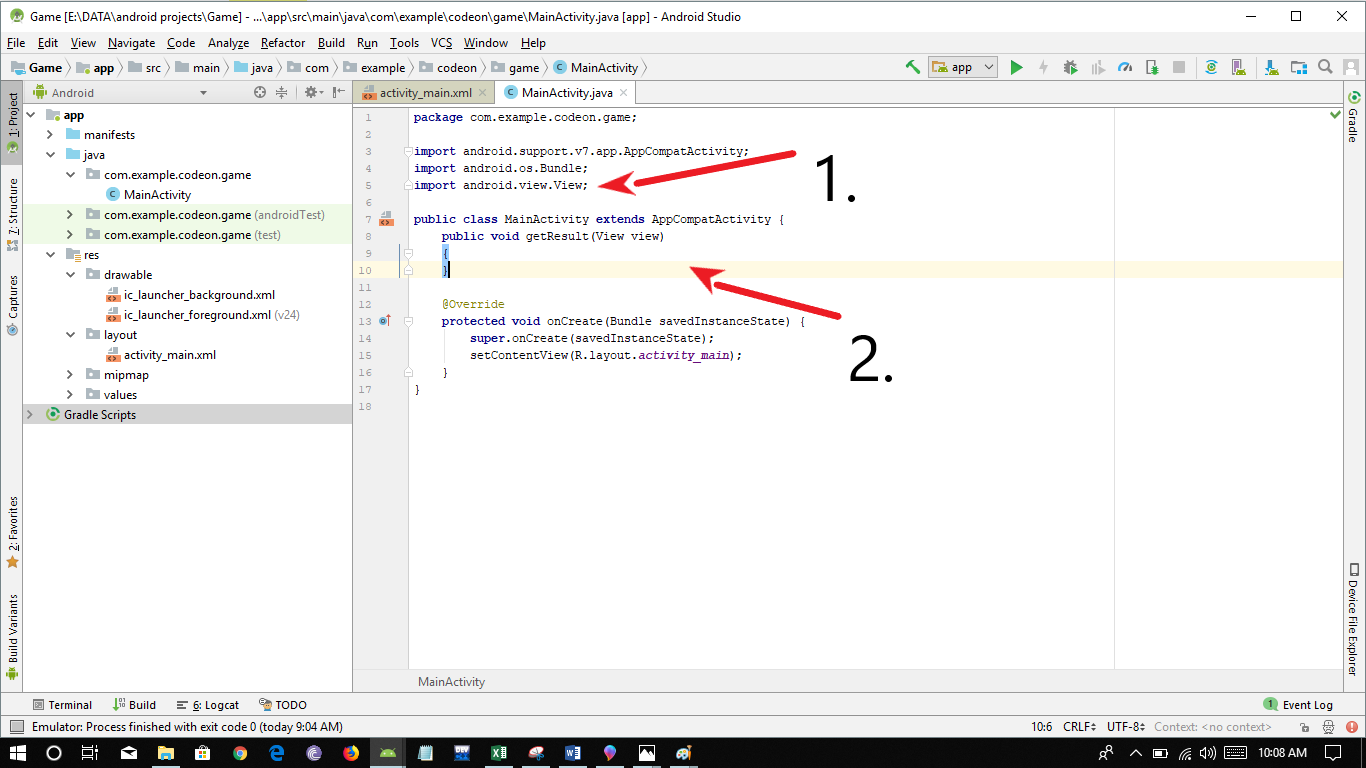
Now create a function named “getResult” (in my case as I have given this onClick option)

First include the view package since we are using view type than create the function getResult ()

**public void** getResult (View view)

{

}



Then we need to take input from the user for this we need to create a variable of integer type and store the number.

For this we first need to store the input data by searching it by its id in required data type of the box.

[default id is editText you can change it as desired form by changing the id under attributes box]

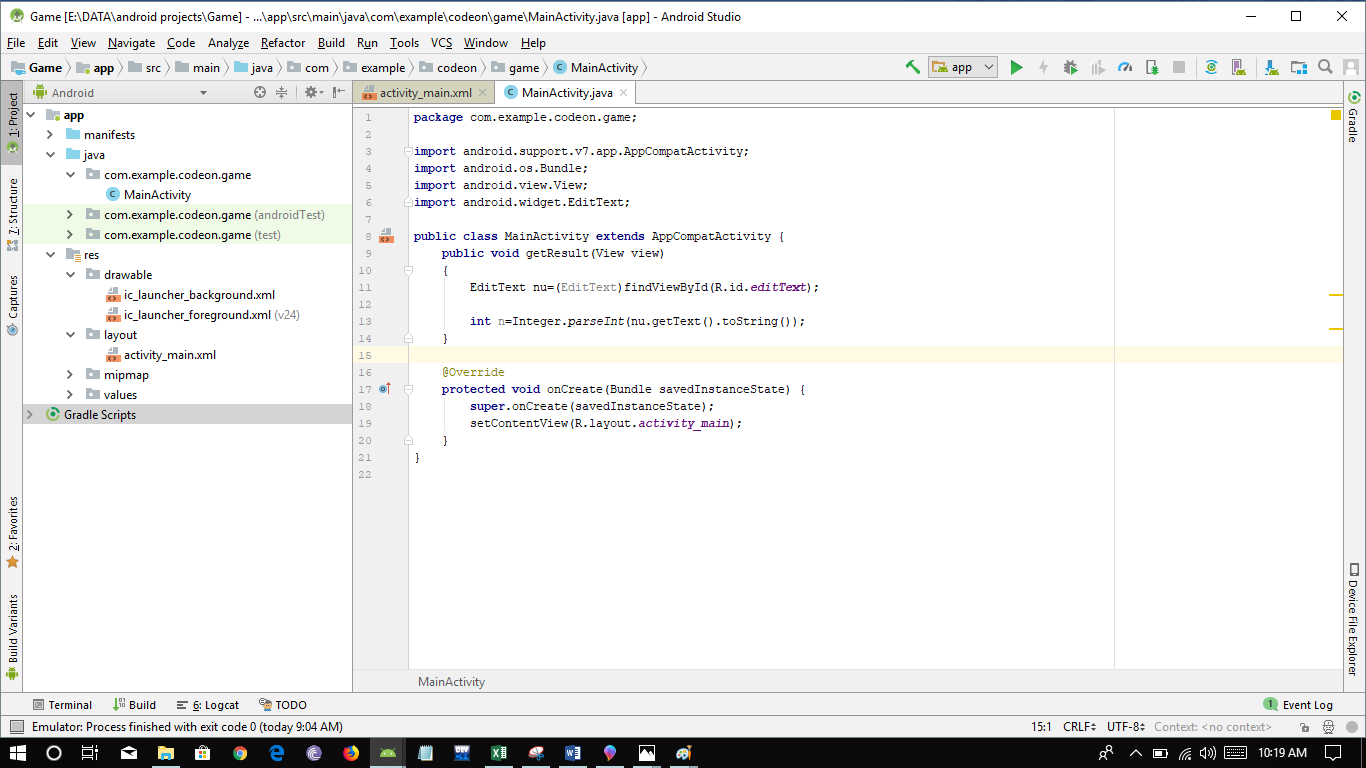
we need to create a EditText type variable and store data by using findViewById function.

Than store the data into integer form.

write the following code under getResult() function:

EditText nu=(EditText)findViewById(R.id.***editText***);  
  
**int** n=Integer.*parseInt*(nu.getText().toString());

Since data is stored in EditText format so we first need to convert it to string and then to integer.



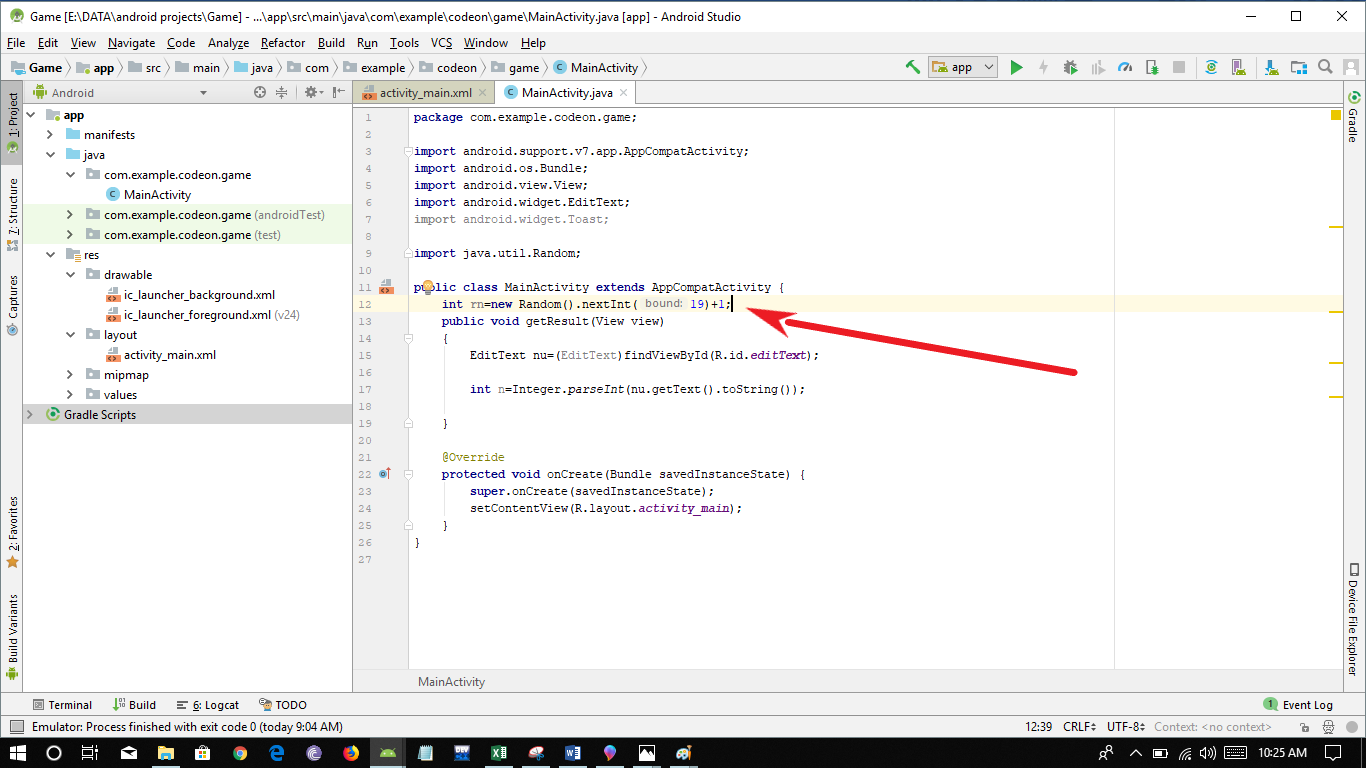
Finally, we have taken input from user now we need to add three cases which would display result to user whether number is equal or lower or higher by help of toast.

Toast:

Toast is a pop-up message which pops-up below the screen with required message or instruction of the user.

Before that create a random number between range of 1 to 20 for this declare and initialize variable just before getResult function:

**int rn**=**new** Random().nextInt(19)+1;



Now the final step adding three cases:

**if**(n==**rn**)  
{  
 Toast.*makeText*(MainActivity.**this**, **"good found the no"**, Toast.***LENGTH\_SHORT***).show();  
}  
**else if**(n>**rn**)  
{  
 Toast.*makeText*(MainActivity.**this**, **"go lower"**, Toast.***LENGTH\_SHORT***).show();  
}  
**else if**(n<**rn**)  
{  
 Toast.*makeText*(MainActivity.**this**, **"go higher"**, Toast.***LENGTH\_SHORT***).show();  
}  
**else**{  
 Toast.*makeText*(MainActivity.**this**, **"wrong input"**, Toast.***LENGTH\_SHORT***).show();  
}

1. If number entered is equal to generated than it will display as toast

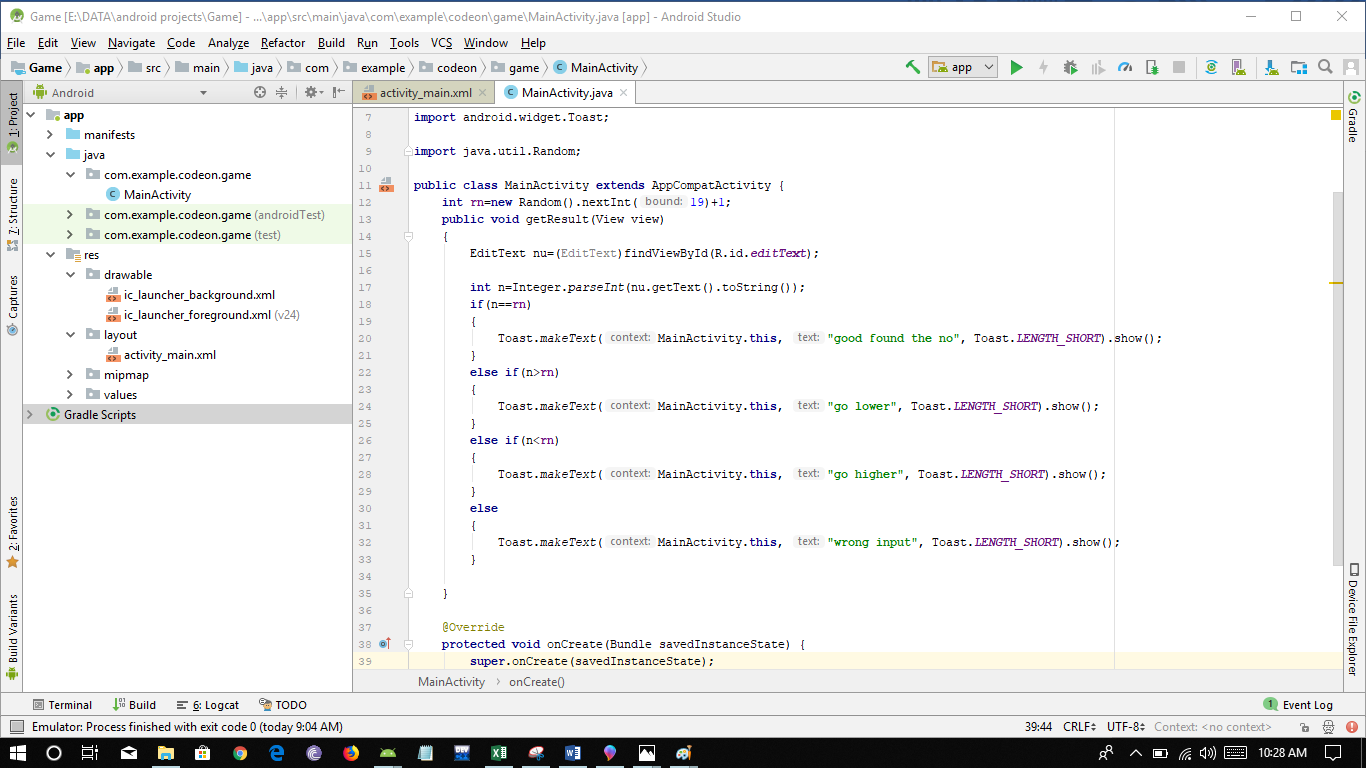
“Good found the no.”

1. If entered number is greater then generated number then it will display as toast

“go lower”

1. If entered number is lower than the generated number than it will display as toast

“go higher”

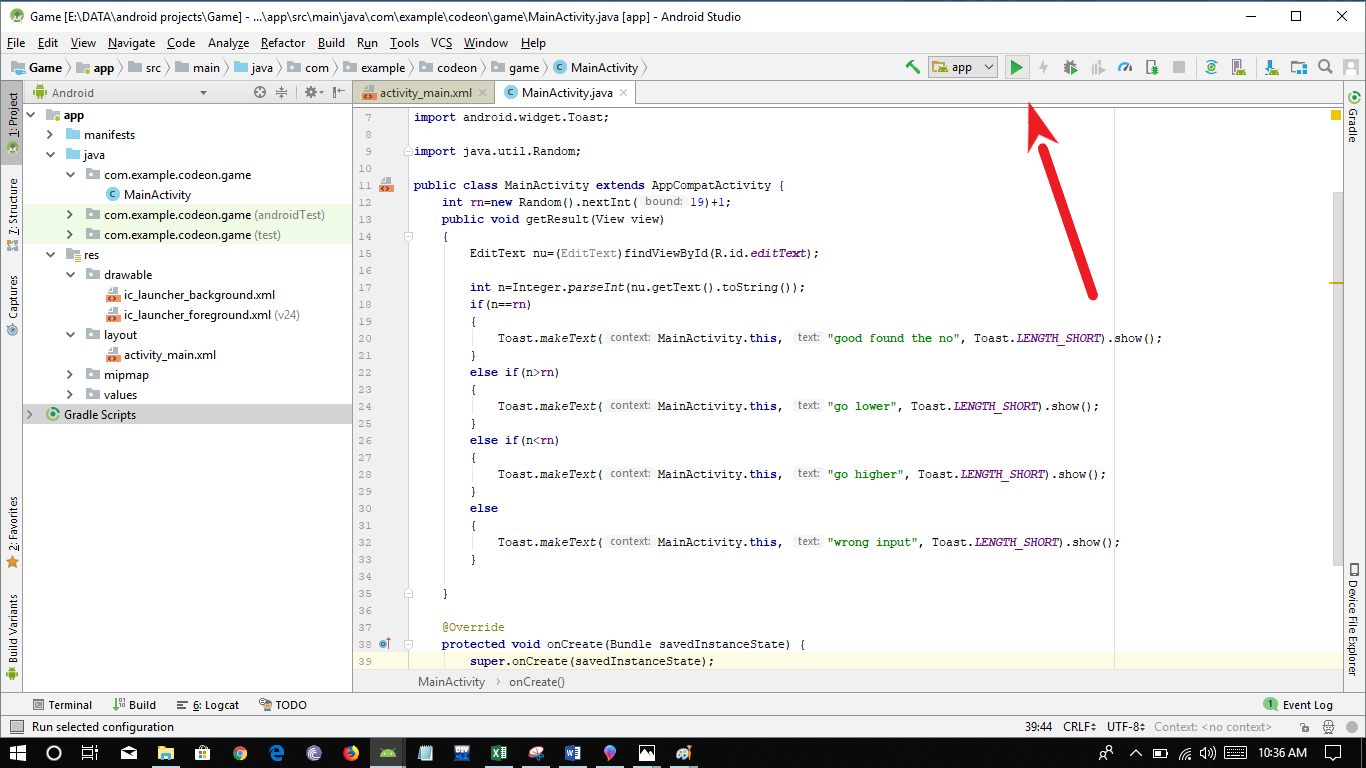


Holla!!!!

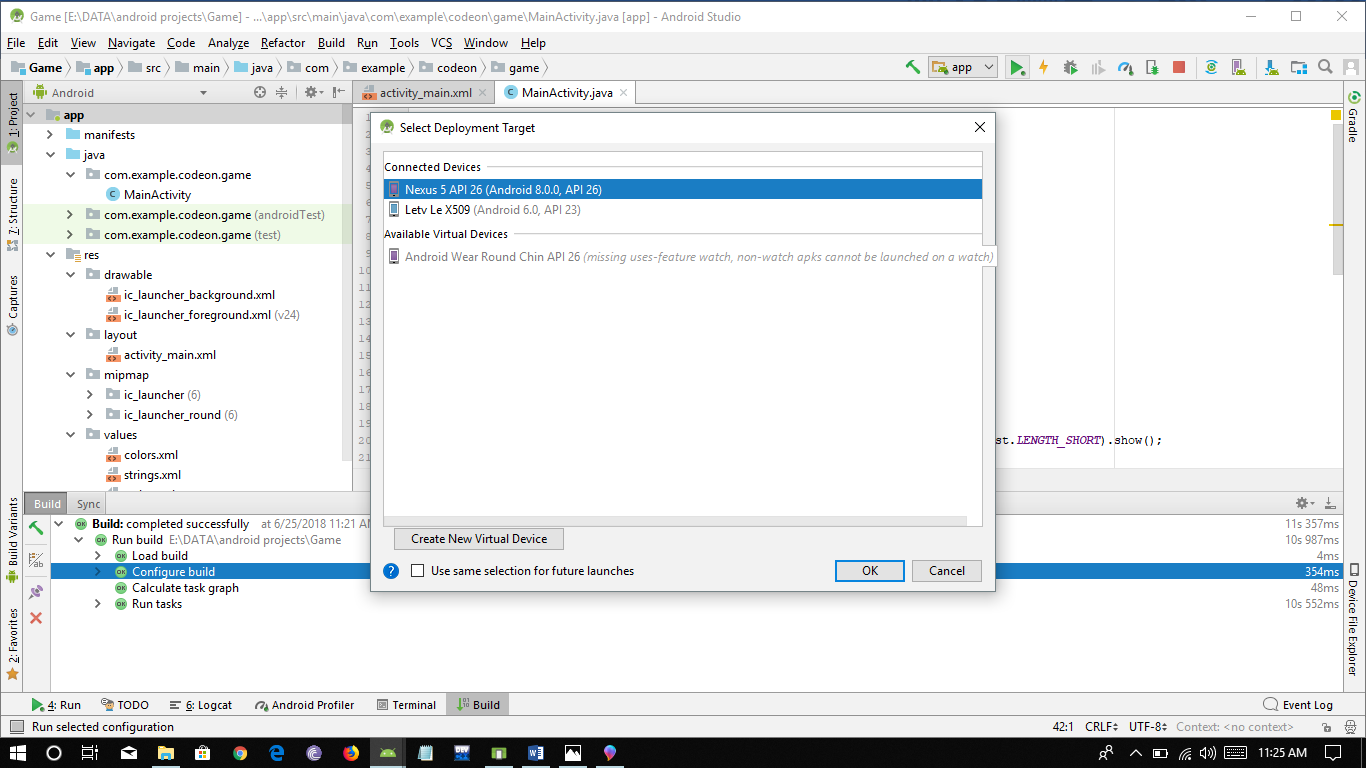
You have successfully build your game app which will interact with user.

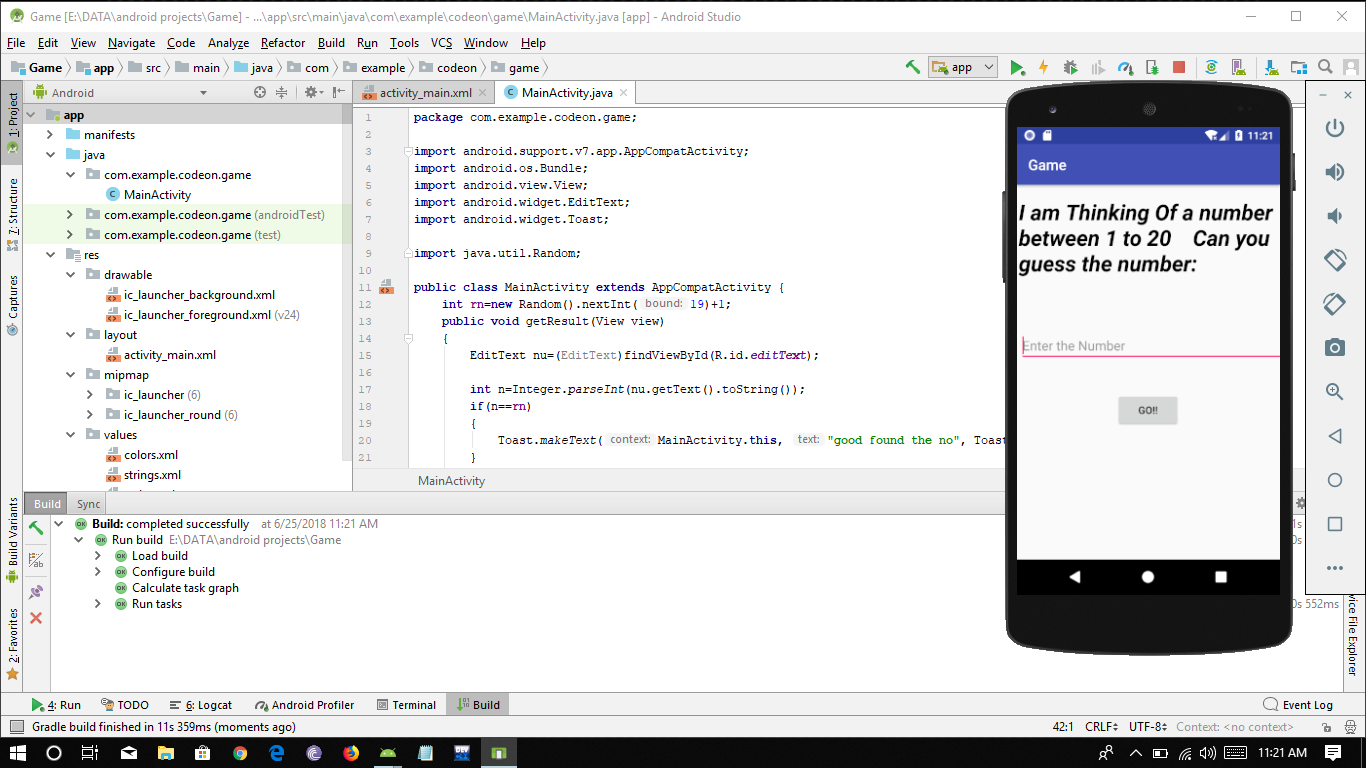
Final step is to run and see whether app is working fine on the device

For this run the app on your AVD or plug-in your android device on USB-Debugging mode.



And then click on desired device and click ok:





Congratulations!!!!

You have successfully build and run you first interactive app.