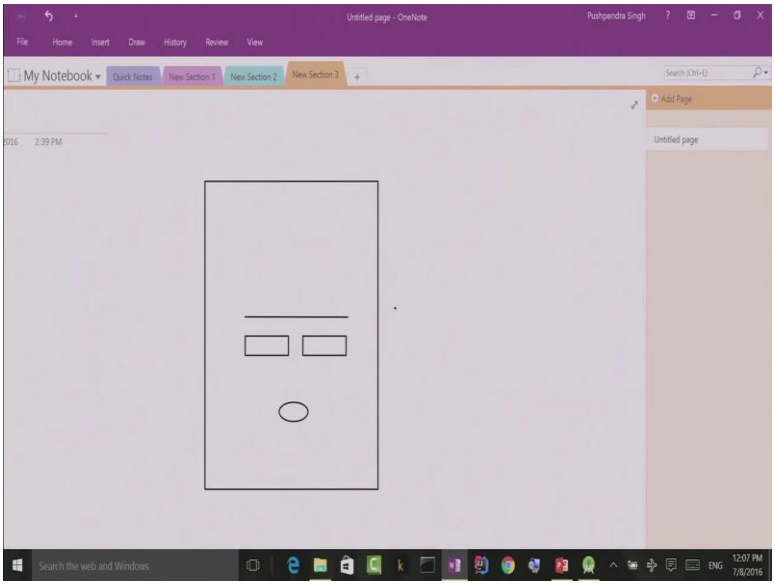
Mobile Computing
Professor Pushpedra Singh
Indraprasth Institute of Information Technology Delhi
Andriod Development
Lecture 09

Hello, today we will create another application called a math quiz. This will be a very simple application which will ask you a math question. Ask you to press whether it is true or false and then depending on your output it will display you whether you are correct or incorrect. You may have seen similar applications where you have to press some button and some pop up appearance. This application is one of those types.

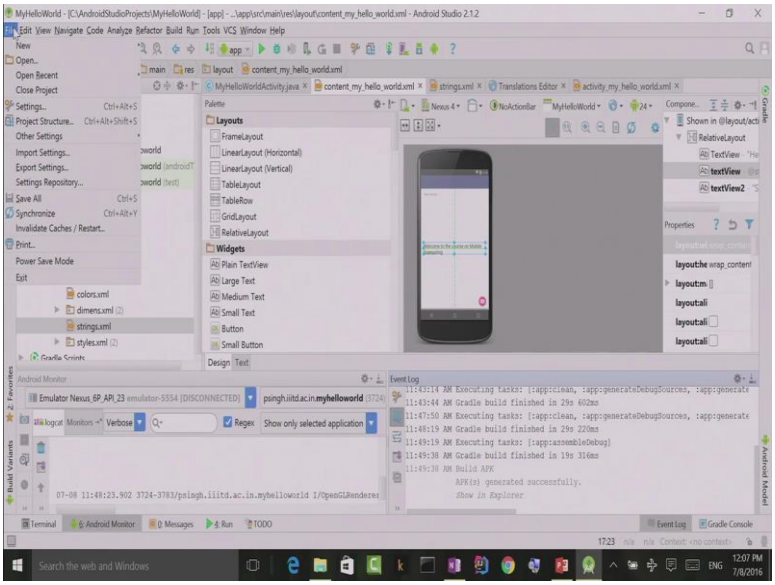
Today I will also explain you while developing the application what is activity? What is the lay out? And how do they work together? We will also navigate thorough different classes that are available by android development environment to us using android studio to show you how your development environment is set up. And how the overall program is working?

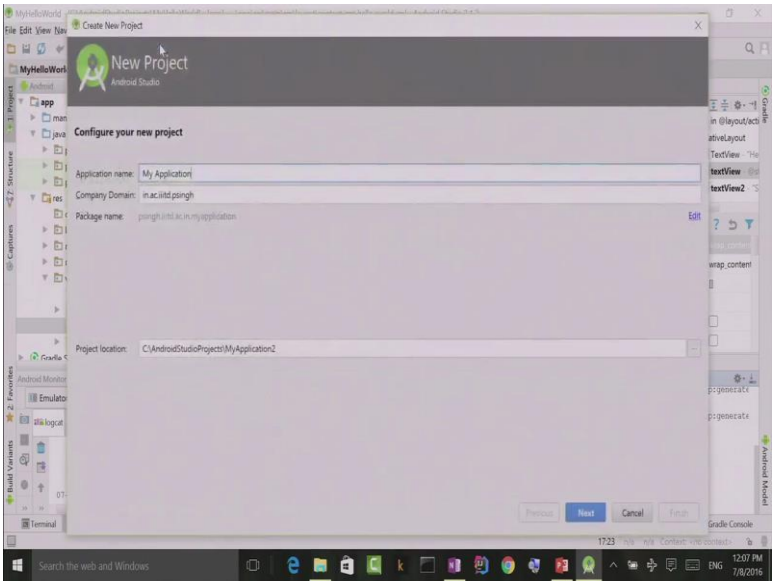
(Refer Slide Time: 1:06)



Let's start with our android studio again.

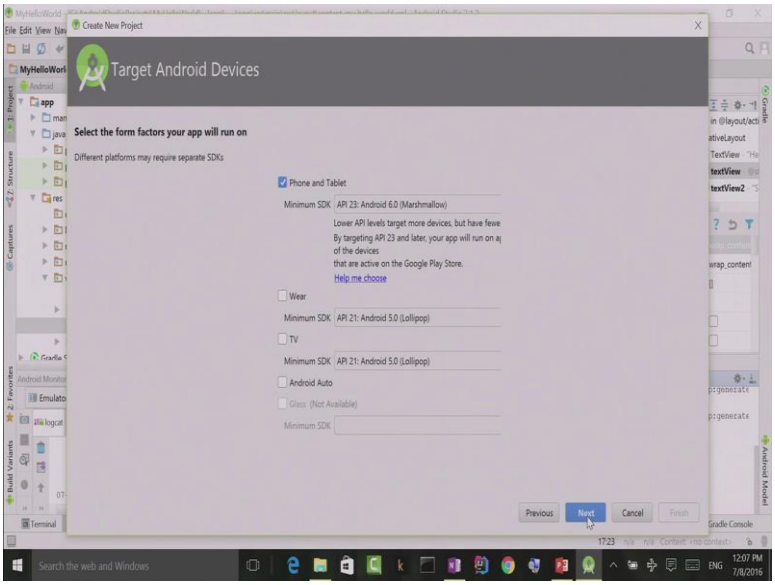
(Refer Slide Time: 1:12)





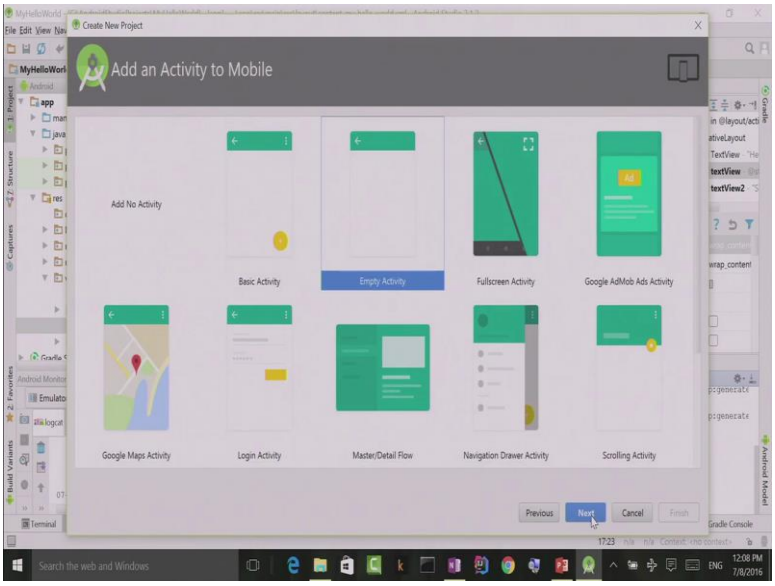
So I am going to create a new application. I will give it the name math quiz. I will not change my package, I will not change my location.

(Refer Slide Time: 1:35)



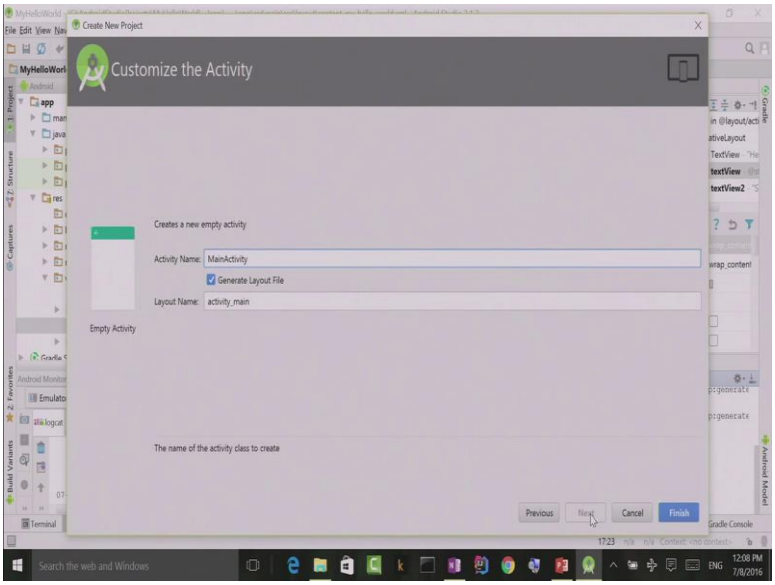
Again I am being asked to choose a SDK I will continue with the SDK that is available on my phone. You may plan to choose to an SDK which is available on your phone. As you can see in the drop down you can read what version satisfies what API.

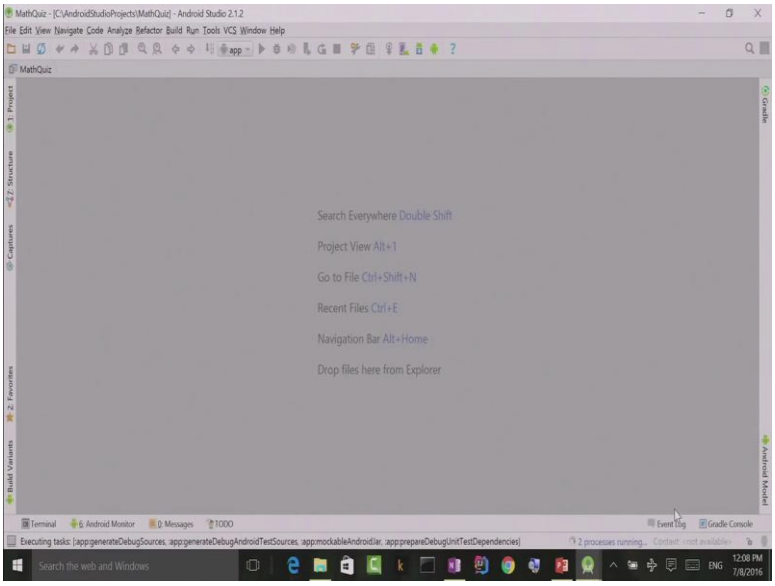
(Refer Slide Time: 2:03)



We have been asked to choose a template for an activity. Last time we have chosen a basic activity this time let us choose an empty activity.

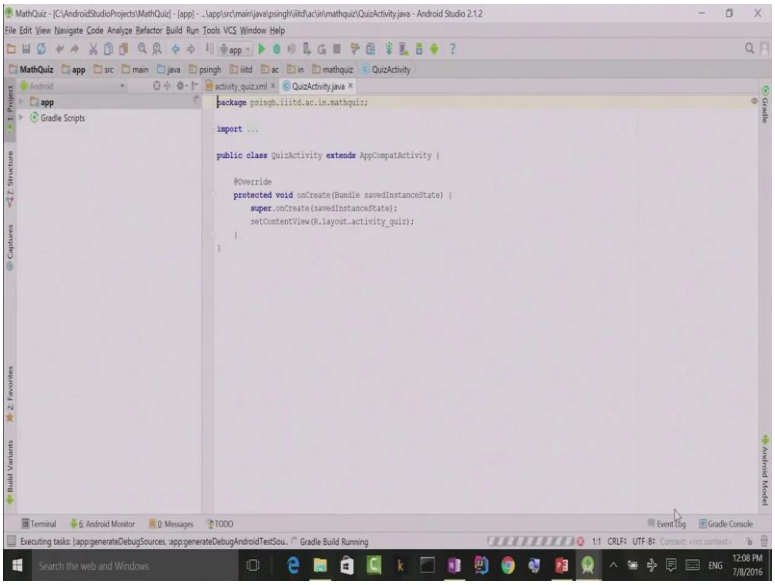
(Refer Slide Time: 2:21)

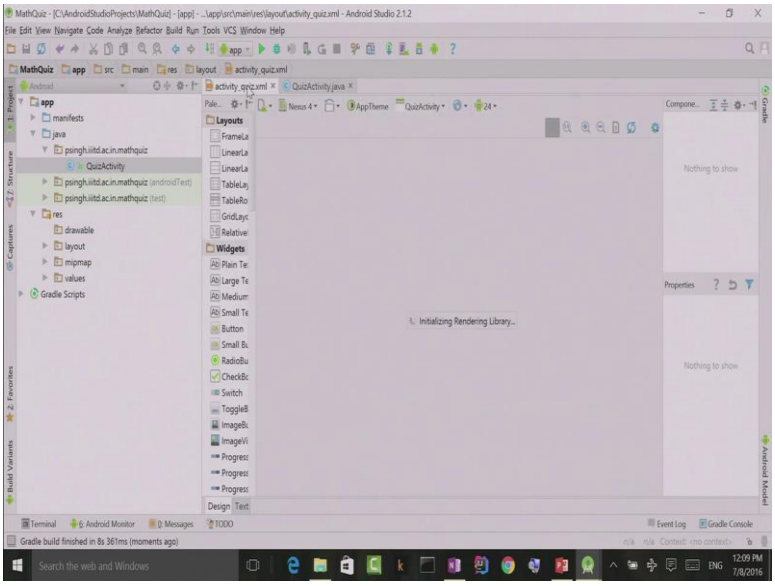


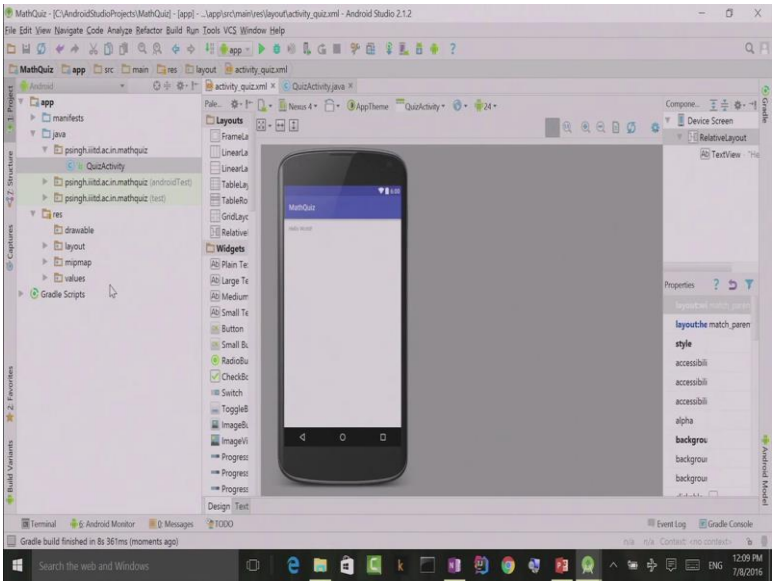


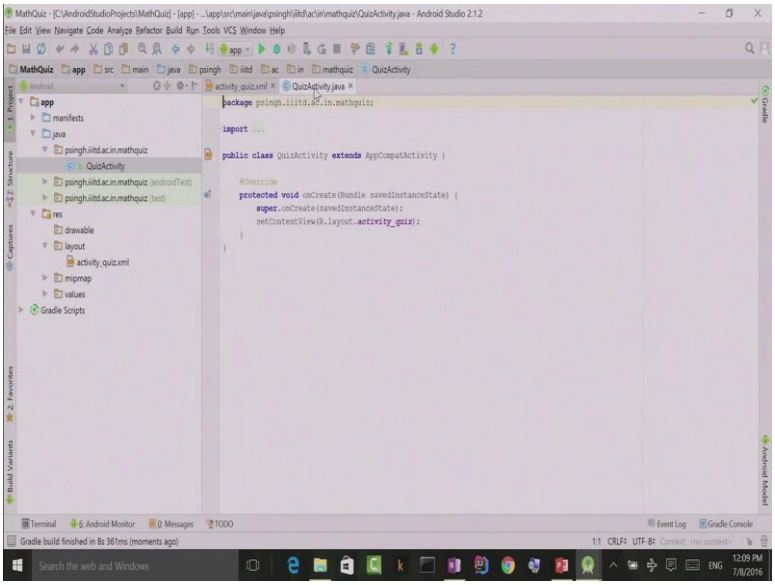
It is again asking us for the name of the activity I will call it quiz activity. I selected to generate a layout file.

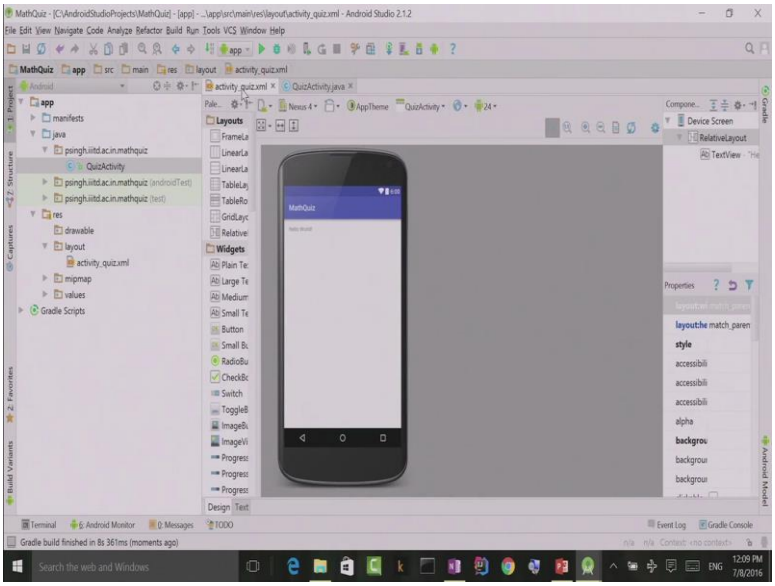
(Refer Slide Time: 2:47)





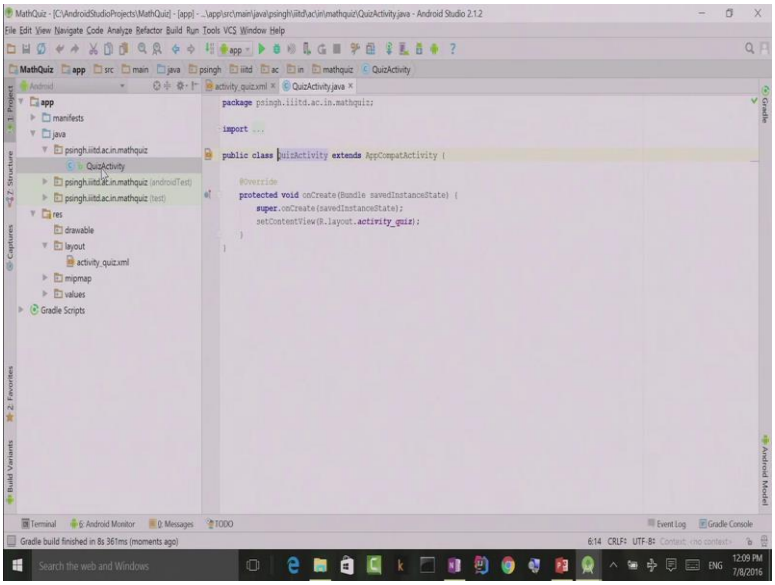


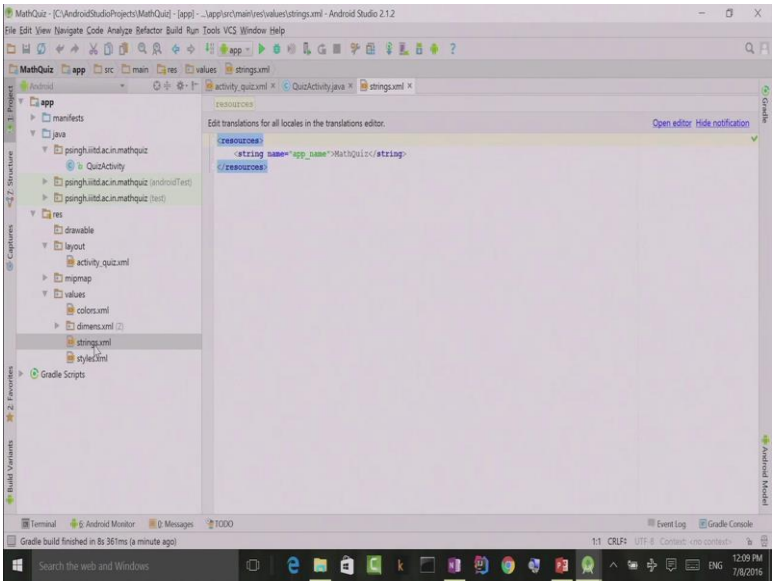


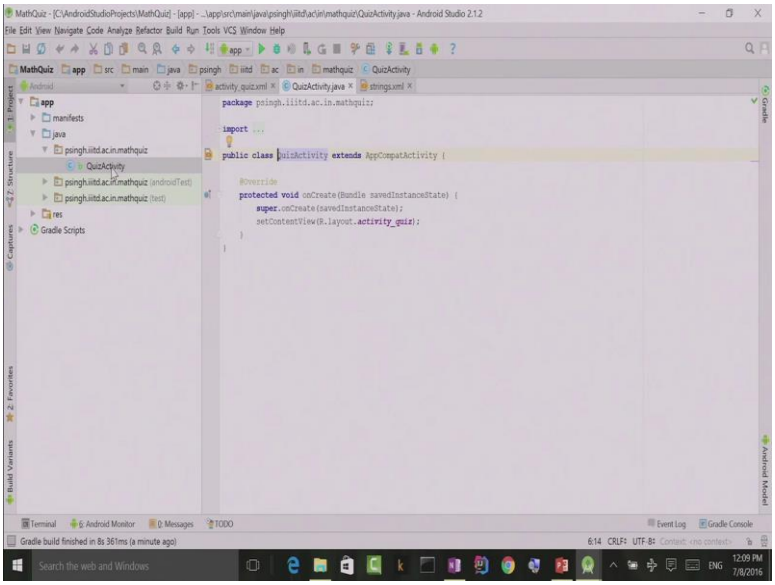


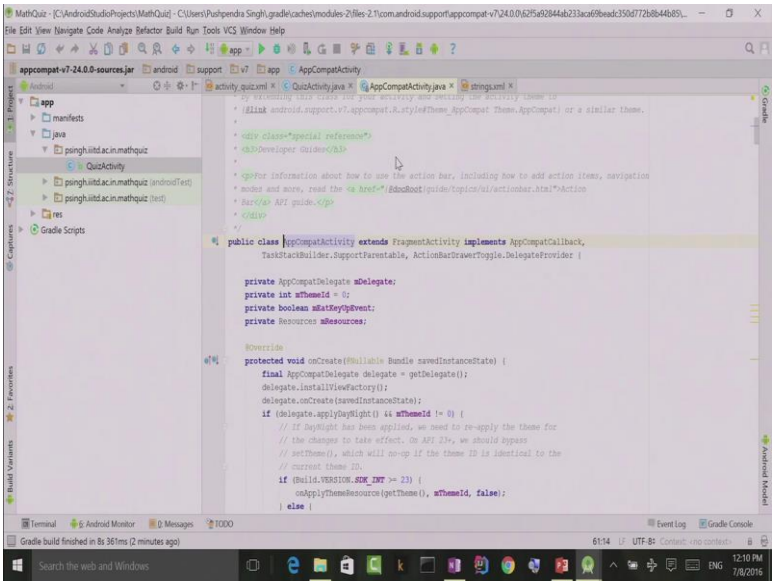
Now my android studio is creating our new project. You can see the build is running. So yes have a quiz activity XML has been created.

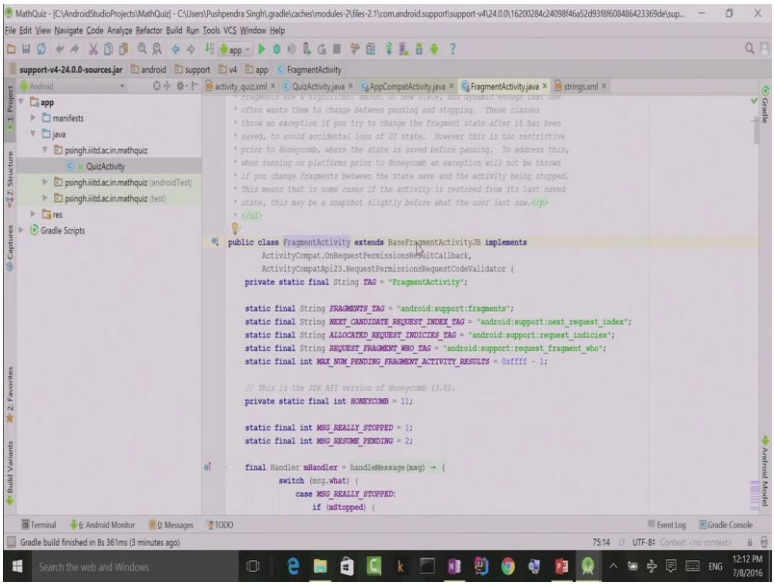
(Refer Slide Time: 3:32)











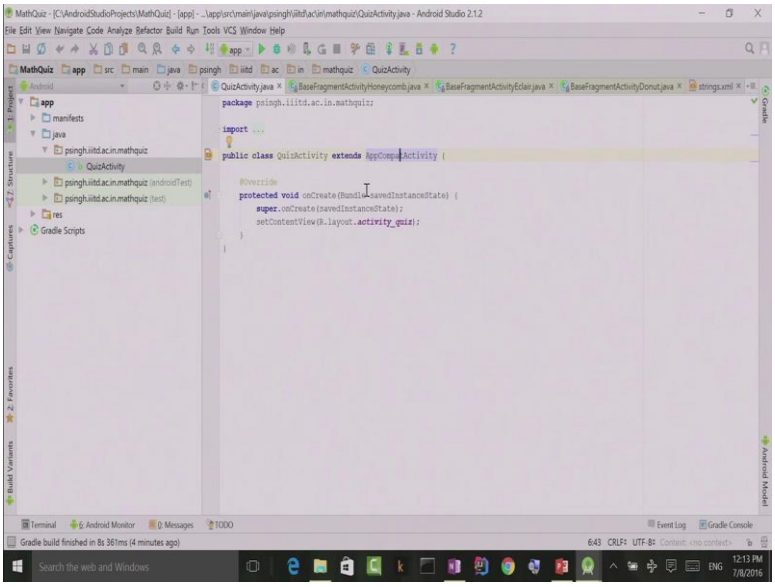
I can go and see the basic structure of the file. Just like the previous project we have a res folder which holds all our resources. Same string which currently only has the app name and our java file. Quiz activity, as you see that this java file is extending another class called app compact activity as we learn in java that extends is used when we drive a class from another class. So you may guess that quiz activity is a sub class off app compact activity class.

Now let us see what is app compact activity has. For seeing this I will press the control button on my keyboard and when I bring my mouse here and If I click I jump into the code of app compact activity. Now I can see my app compact activity extends fragment activity and implements app compact call back tasktackbuilder.supportparentable action bar drawer toggle.deligate provider.

As you may have already guessed these are different interfaces that is implementing. Let's go back up in hierarchy and click on fragment activity. We find that fragment activity extends base fragment activity JB class and implements two more interfaces. Let's go further up we find that even base fragment activity extends another class. We keep going further up one more, one more.

You may have also noticed that every time I go up I am actually going to another version of android now I have come till donut and from that I reach to a class called activity which is no longer extending another class. So yes, this was our path lets come back and see what we wanted to check by going up in hierarchy.

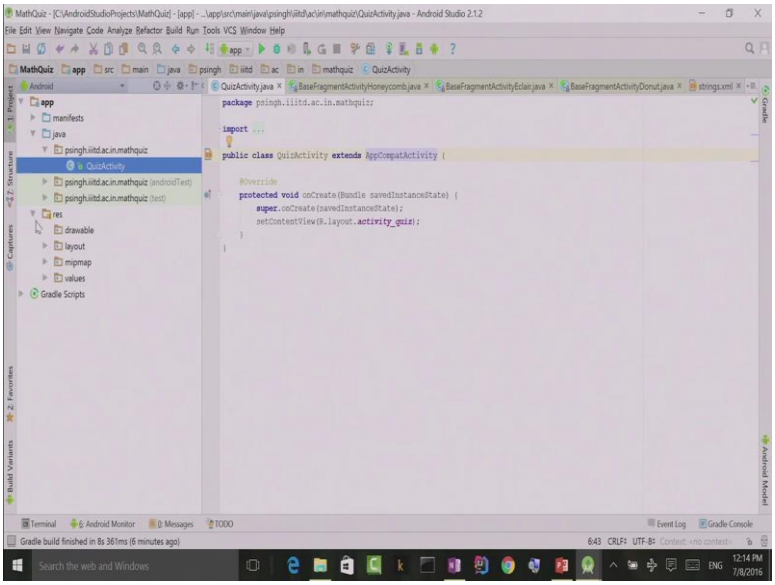
(Refer Slide Time: 6:35)

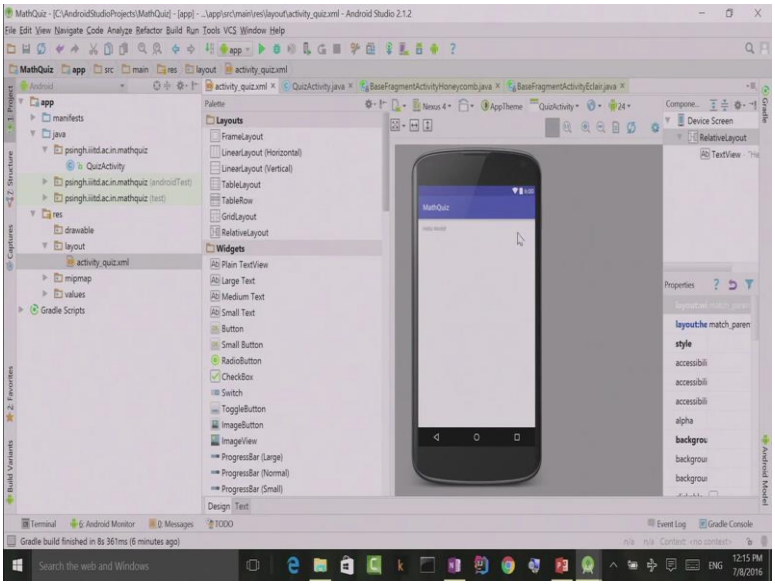


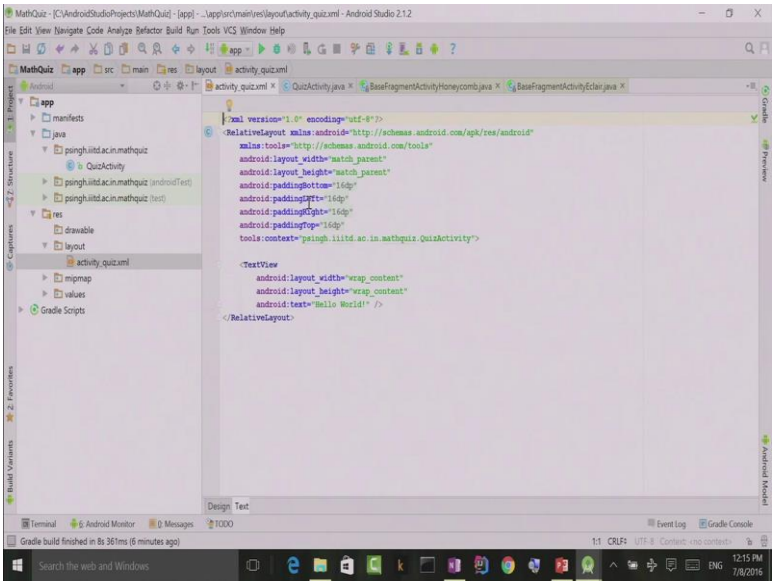
So I come back to our quiz activity class. You saw that quiz activity is nothing but a extension of multiple subclasses which all end up at a class called activity. An activity in java is an instance of activity which is a class in android SDK. We use an activity to manage user interaction with a screen of information. Just like in this example we will use the activity to manage interaction when you press true or false button to a mathematics question.

The another concept that we will use in this program is layouts. We also used layouts in our previous example. A layout defines a set of user interface objects and how they are positioned on the screen. Layout is made up of definition written in XML as we saw in case of strings and today we will say C for layouts and where each of these definitions will describe one of the objects on the screen.

(Refer Slide Time: 7:49)

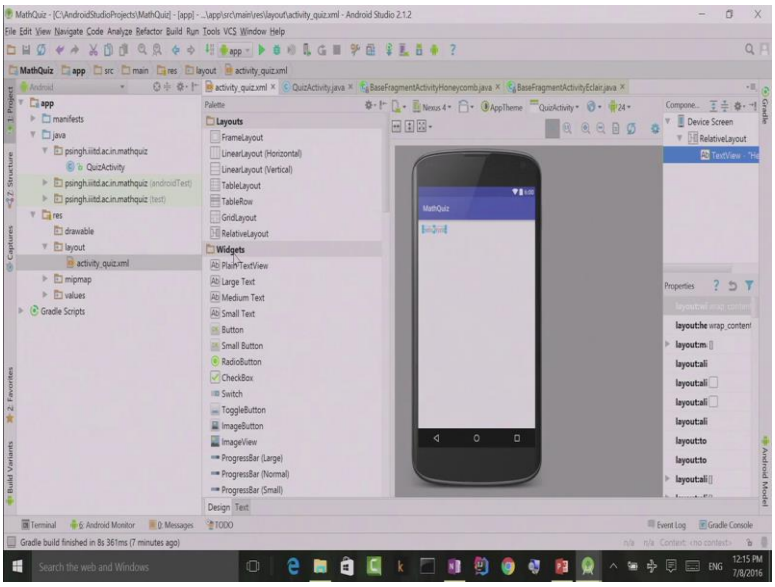






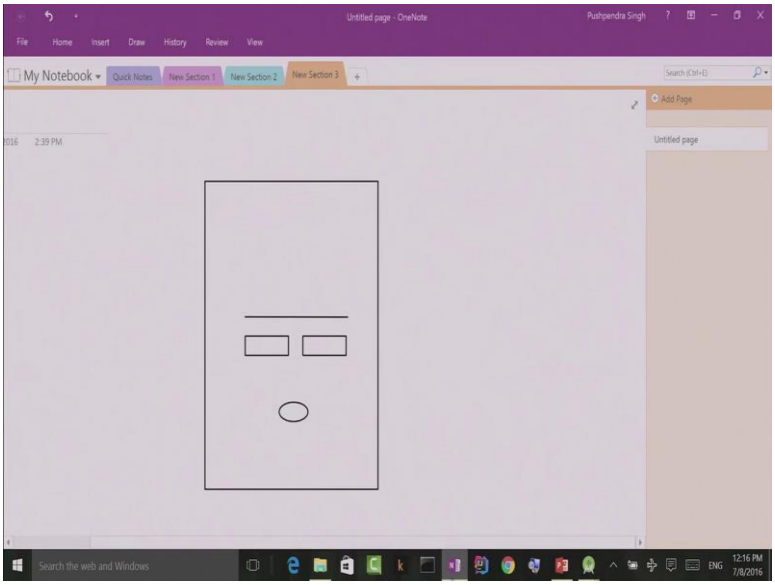
So let's go back to our quiz activity and let's get back to our resource activityquiz.xml yes. Now we see that this is showing us a relative layout. I will go to the test and now I can see the XML written for it. Today we will try to change this layout not graphically but by converting the XML.

(Refer Slide Time: 8:38)



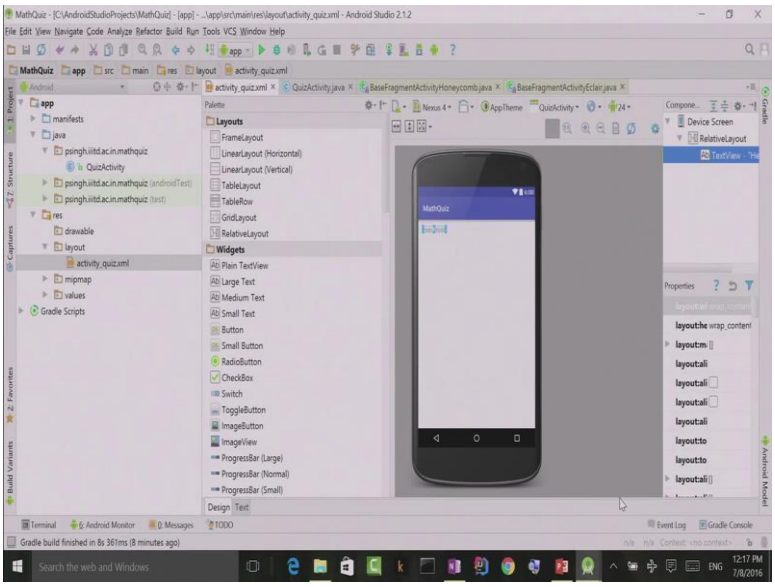
As explained in the last example you may also see different widgets in the last example we had chosen a last text to display a welcome message which was kind of a widget. Widgets are building blocks in android which are used to compose a user interface. You can see different types of widgets that are available to us. Buttons, small buttons, check boxes, radio buttons, progress bar, toggle button, etc. Now let's come to our application that we have in mind.

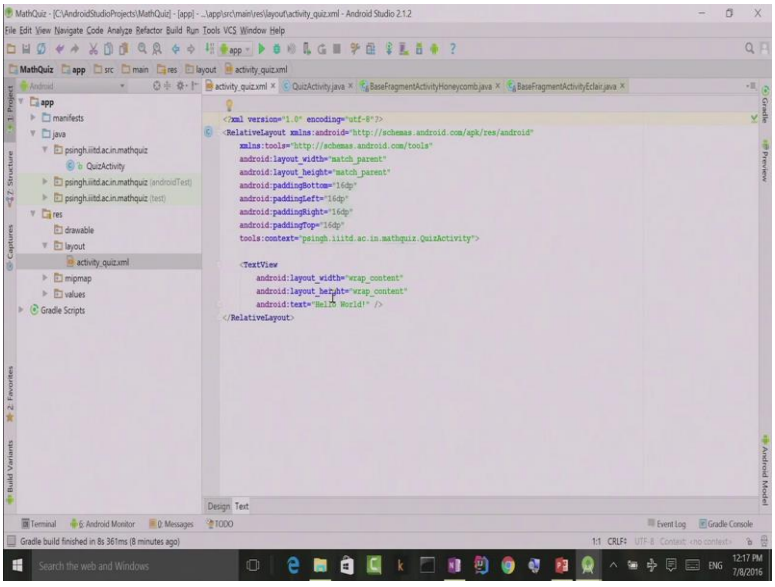
(Refer Slide Time: 9:19)



So we are thinking of an application which is roughly looks like this. This is our android screen we will display some text here that will be our question these will be our two buttons and once one of these buttons is pressed we will have a pop up which will show us whether it is correct or incorrect. In android language I see it as a layout which is vertical which then contains a text view that covers our question then there is a horizontal layout which contains two buttons and then there is a (pop) pop up which in android language we called as toast.

(Refer Slide Time: 10:08)

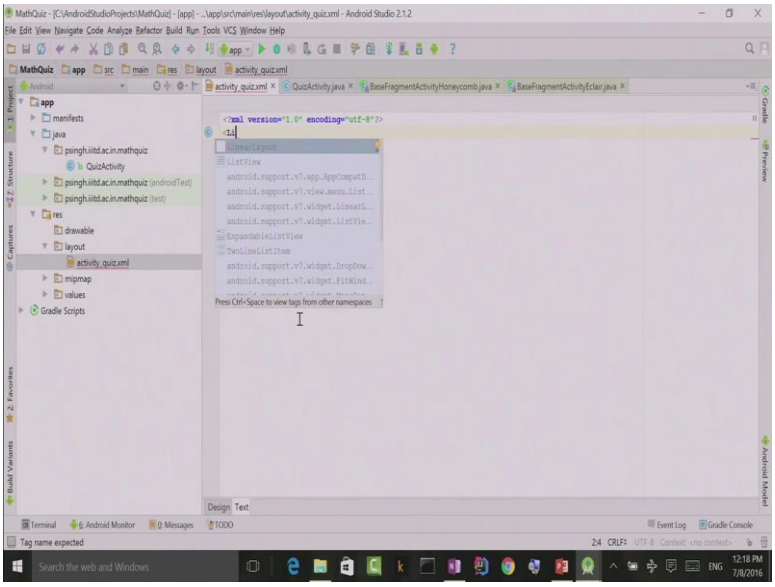




Let's go back to our android studio. And try to see what we can do. So now I would like you to open your android studio the text format of the activity quiz.xml and start modifying it using the xml not using the GUI. As an android developer you need to have expertise both in

manipulating the xml and also in manipulating the GUI component. First we are going to delete everything which is available to us.

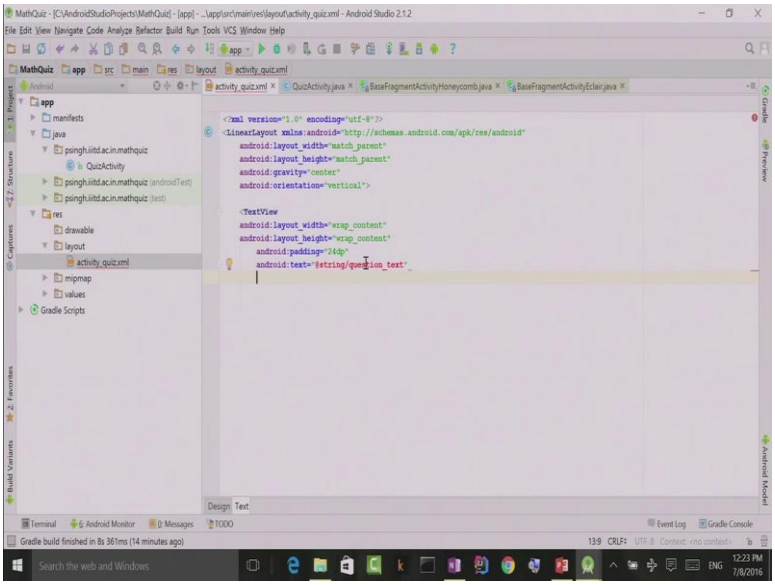
(Refer Slide Time: 11:30)



Now let's type the layout that we want for our application we will start with the linear layout as you type android will give you enough hints. Please see that you type carefully but if you make a mistake your program will not compile. I want to set out my layout width I get multiple options. I choose the right option of width uhh sorry uhh type android layout I come to the width and I set the width to something called match parent. I will explain what it means later.

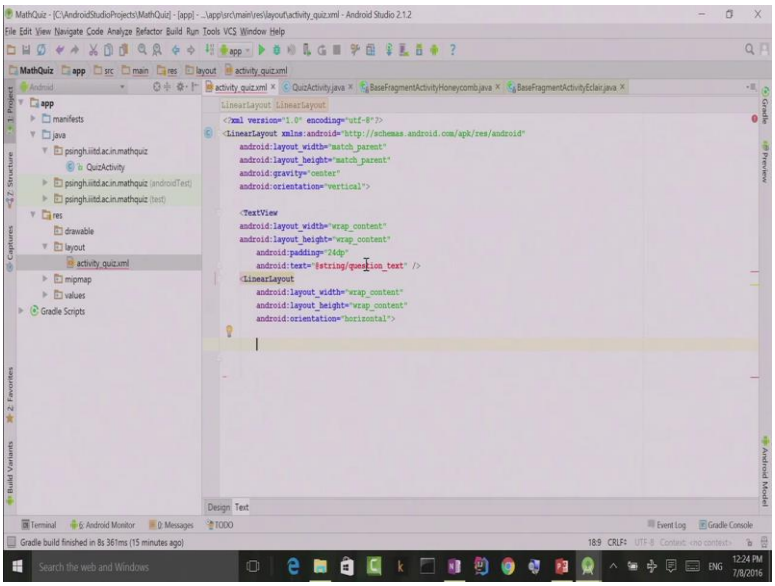
Similarly I want to set out the height. Height will also set to the same. So I want a vertical orientation so this is my initial outer screen now I want a text view displays my question. So in last example we added a text view using the GUI and this we will add everything by modifying the xml.uhh I am missing. So if your android studio stop giving you hint you should look up in the xml may be you are missing a closing bracket.

(Refer Slide Time: 16:07)



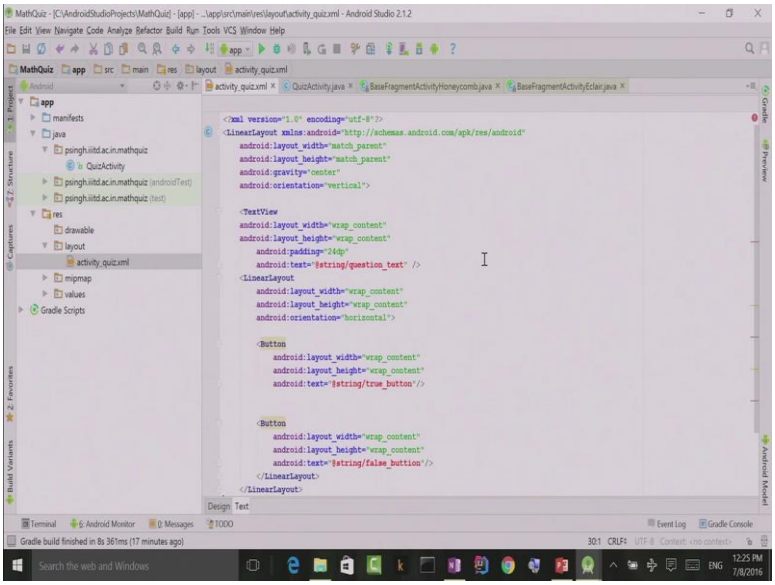
You can see that android has started displaying red which basically means that are currently do not have the symbol in my program and that's why android cannot resolve it. Don't worry we will soon create it. That's a start with another linear layout.

(Refer Slide Time: 17:16)



This layout I will put two buttons.

(Refer Slide Time: 18:30)

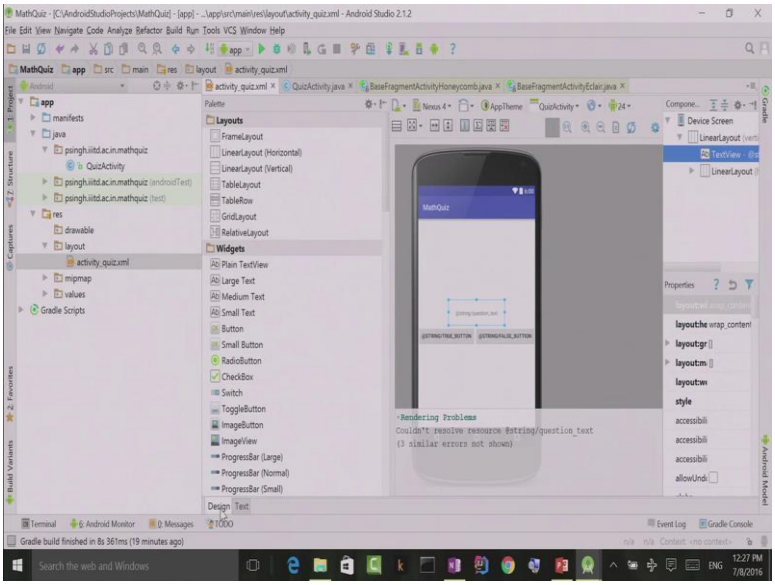


Yes, so now we are done with our xml file lets go through it again. Instead of the relative layout that was given to us we converted it into a linear layout. We will be learning more about layout later. I gave the width height parameters with a value match parent which essentially means that it should match with the screen size. Then I gave a text view with wrap content which means that it will take as much as space as it wants.

I provided some padding and I then gave text to refer to a resource called question underscore text as you can see that android is currently displaying it as red and giving me a warning that it cannot resolve symbol. If I compile my program at this stage I will get an error.

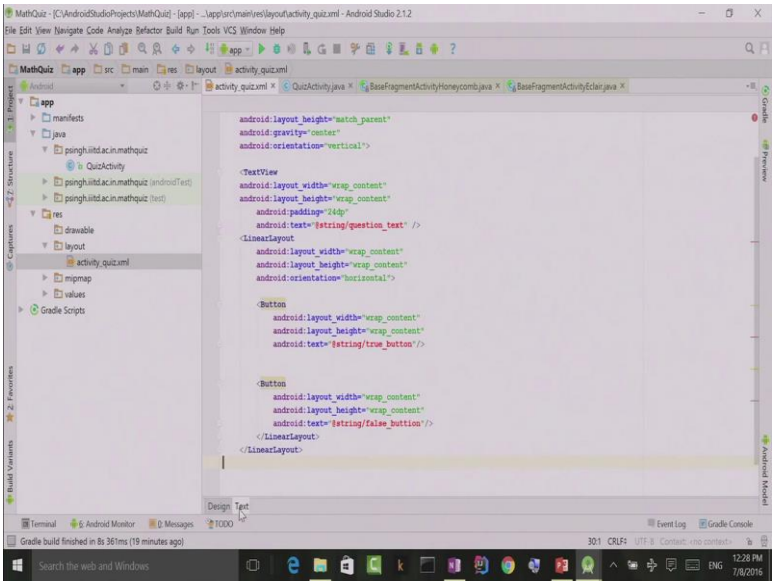
I defined another linear layout in that I describe a horizontal layout and within that I created two buttons I correspond them to resources true button and false button and I get similar warning that android cannot resolve these resources. Can you think where these resources should be? Okay we will come to that part later.

(Refer Slide Time: 20:30)



Let's go back save it and try to see what are we getting in the design. Okay so as you will see that design has changed now we are seeing our question text, we are seeing the place holder for our buttons. Though we are getting problem which are saying could not resolve the symbols. This is fine with us because we are so far not defined them.

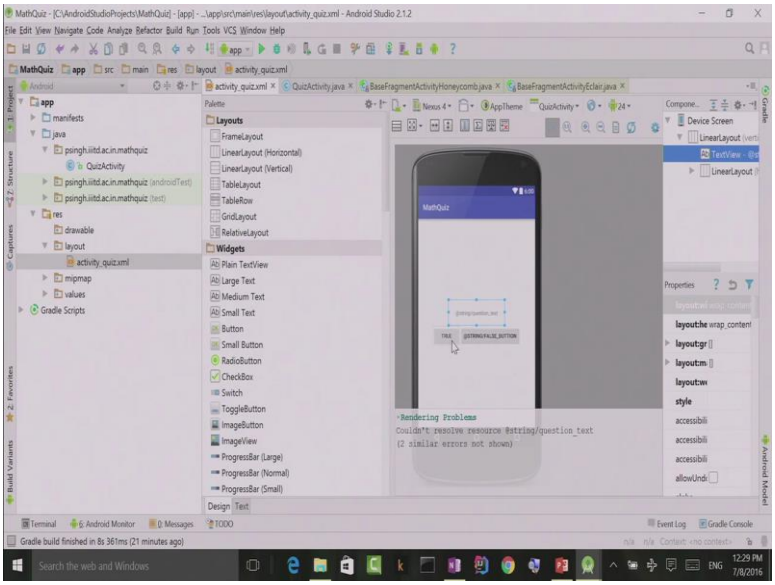
(Refer Slide Time: 21:08)

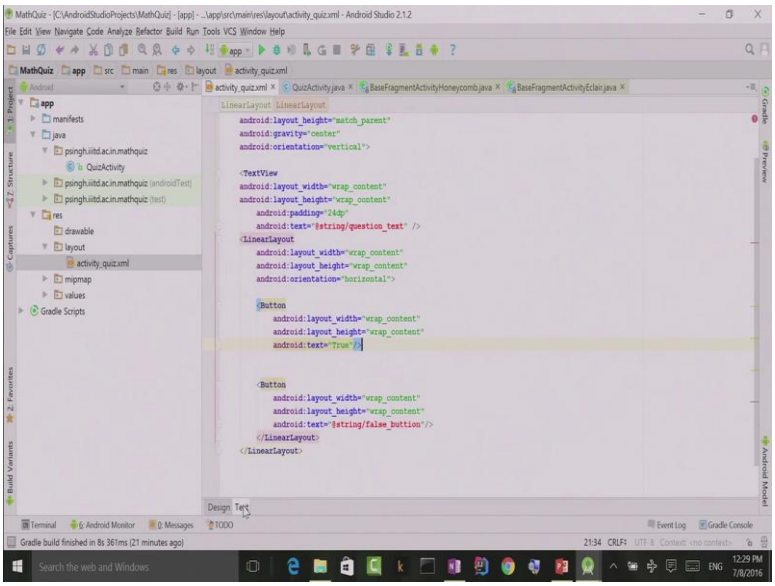


Let's go back to our text and try to see that what are the attributes values that we have given. The one value match parent means that you will be as big as its parent. The wrap content which means you will be as big as it content requires. You may also see other values such as fill parent and you may want to change some of these values to see how much do they effect. You may want to change some of these values to see the effect on your screen.

We give padding as you may know padding shows that somehow there is a space available between the boundary and the text. Then we give the orientation of vertical at one place and horizontal at other place as our need. And then we define the text that will be displayed now instead of heart coding this text I am referring it to a resource or let me do one thing let me first heart code it to a text and see how does it impact. Suppose I set it to true.

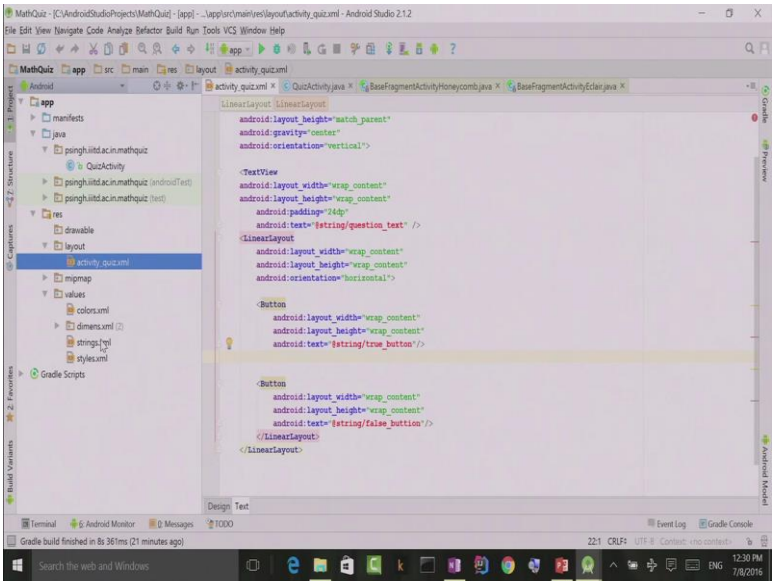
(Refer Slide Time: 22:35)

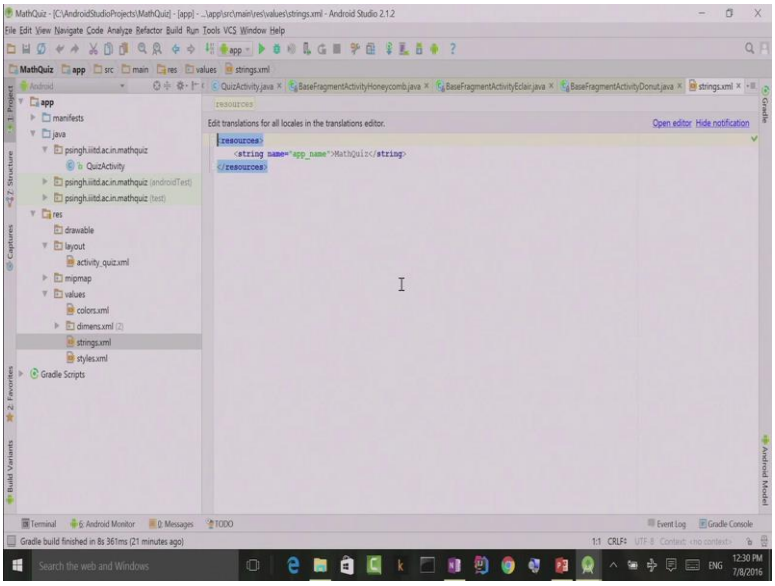


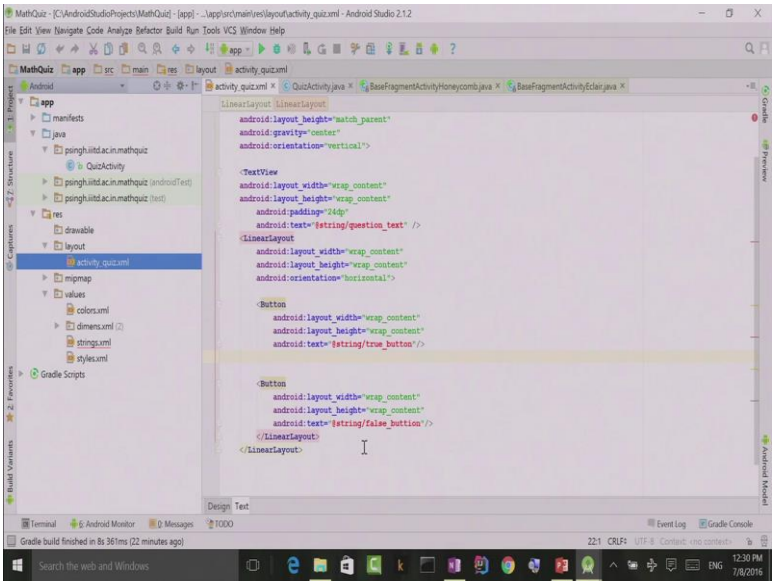


Then I go now I can see that I can see my button as true sometimes you may be tempted to do it because it saves some of your work. But this is not a current approach. You should always refer to a resource rather than hard coding a string. So let's convert it back. Now our first task is to create these resources.

(Refer Slide Time: 23:15)

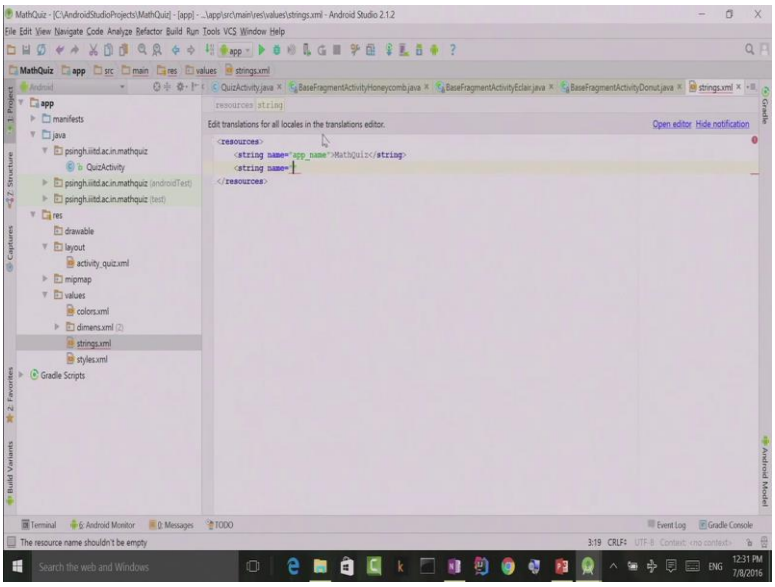






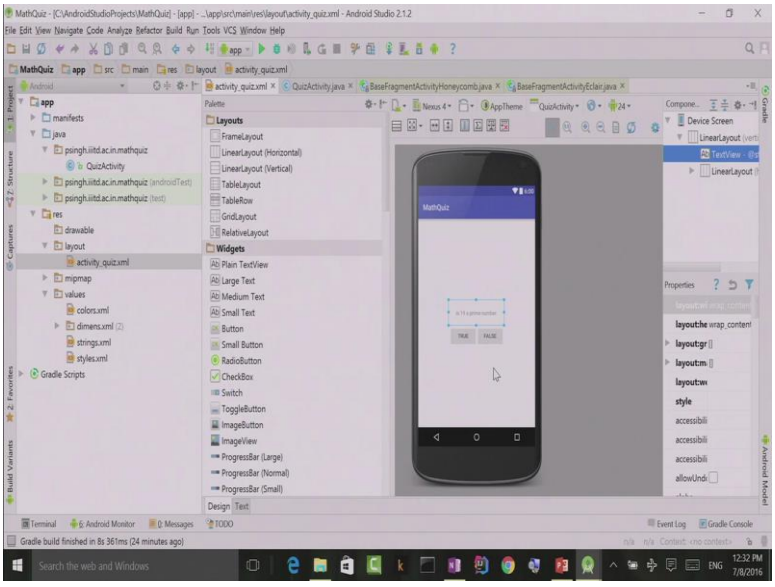
Let’s go into the values into the string. And start creating our resources. What’s the name that you should be giving here? Should obviously match to what you gave earlier question underscore text, true underscore button, false underscore button. Oohh we have a typo here now it is fine.

(Refer Slide Time: 24:02)



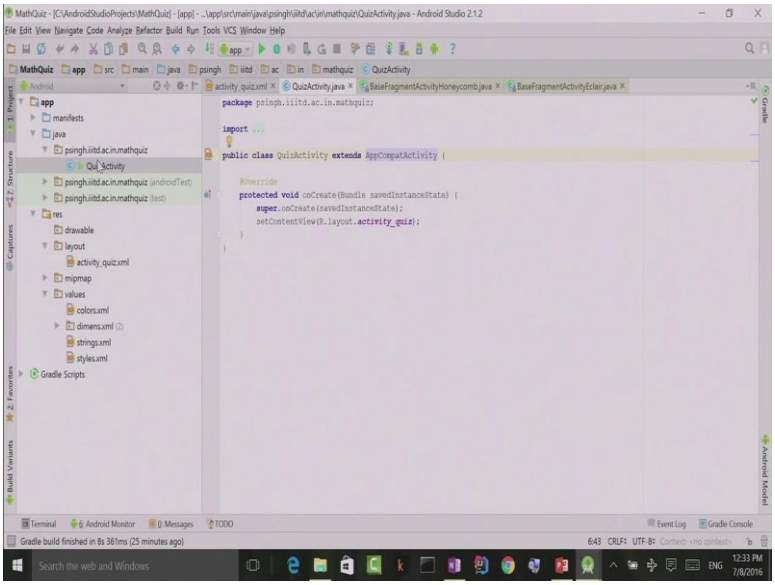
Now let’s go back to our strings. Create our first resource. Give it the value that you want suppose our question is 19 a prime number? Let’s create our second resource. Give it the value that we wanted to display and then finally create our third resource.

(Refer Slide Time: 25:41)



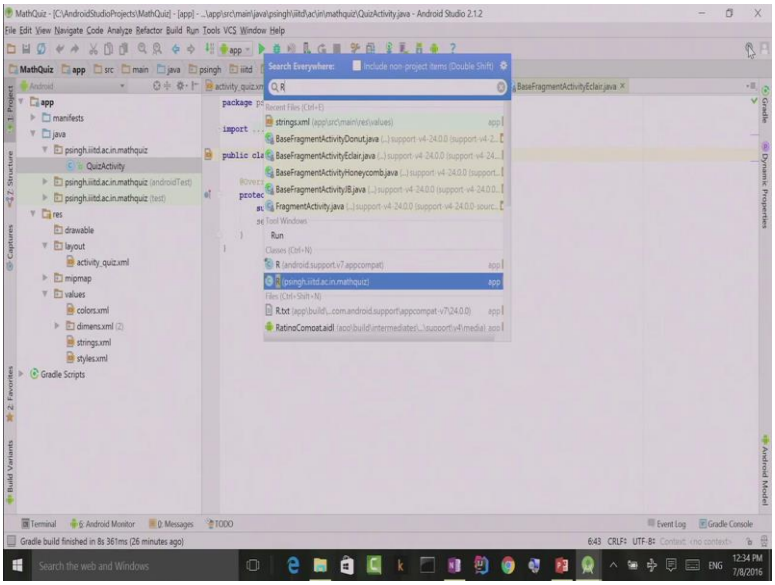
Right. Now you see that initial warring that we were getting from android studio is now gone. We can see the strings that we wanted to display that's our question and also the correct labels on the button that we have created. You may be wondering that how could your layout xml got created into view object. This is done for you by the android studio and this is done for many programs by multiple of files that are automatically created by the studio.

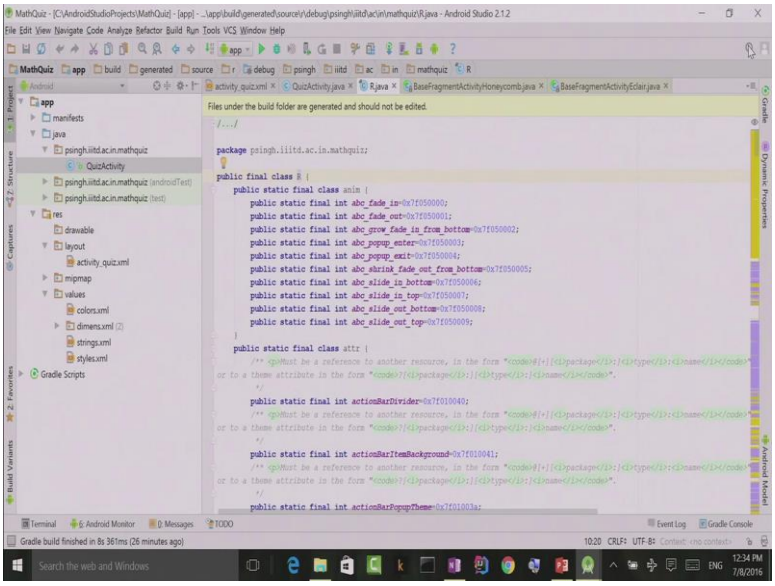
(Refer Slide Time: 26:30)



So far you have only been concerned with your main file quiz activity now let's see what other files are available that are making this program run. This file as you see has only two functions one is on create now at this on create function is calling the on create function of the super class or the parent class and then the second function is set content view. In set content view you see R.layout.activityquiz what this R is? Let's try to find the answer to this.

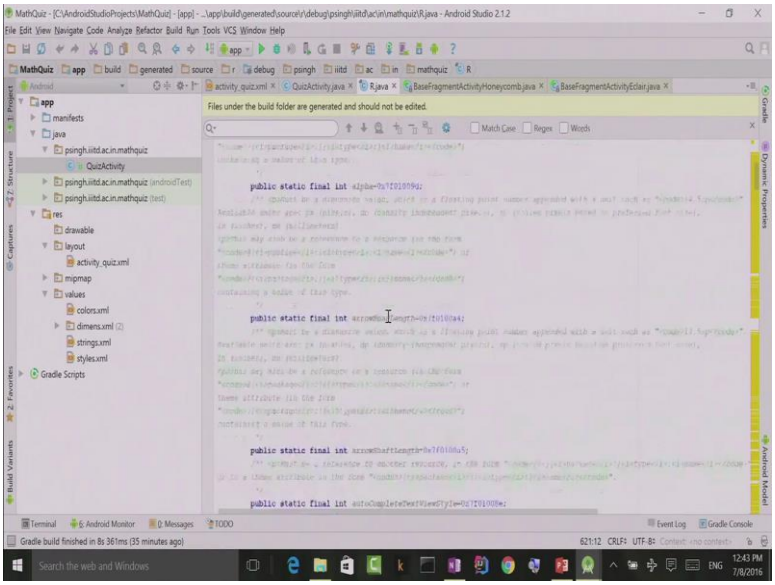
(Refer Slide Time: 27:30)

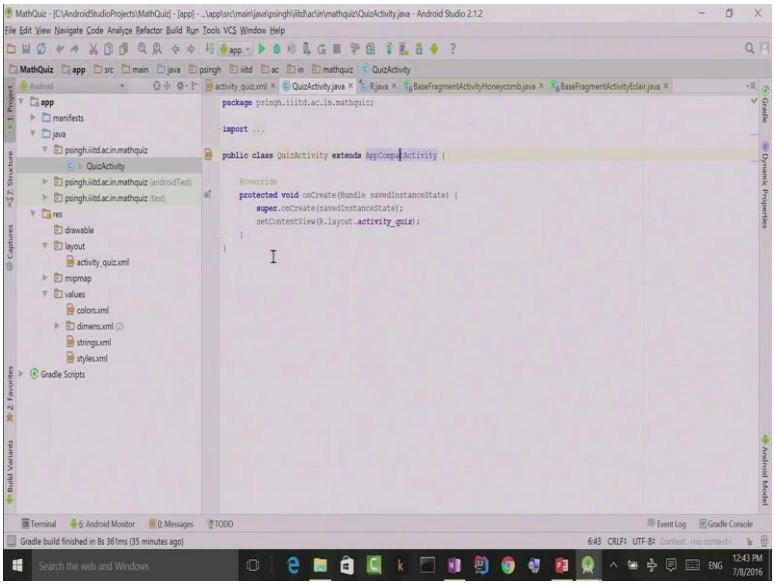




I type R and here we go there is file called R.java which has been created as part of this project. However if you go on the left side you will not see it. The reason is that these files are auto generated and they should not be edited. However we would like to view the contents of R to understand that how the how an android program works.

(Refer Slide Time: 28:20)

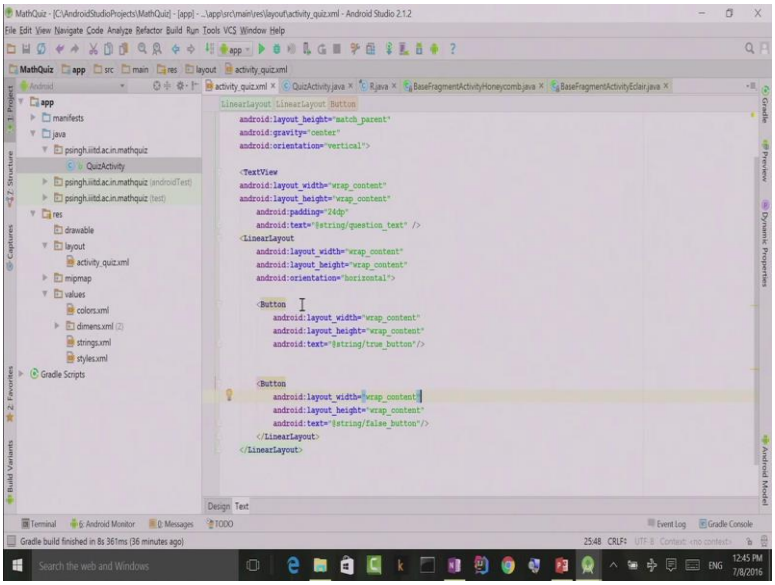


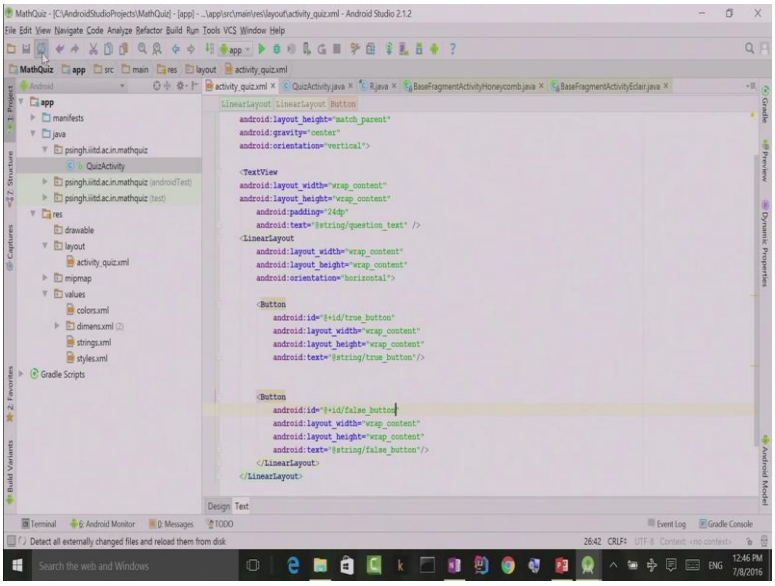


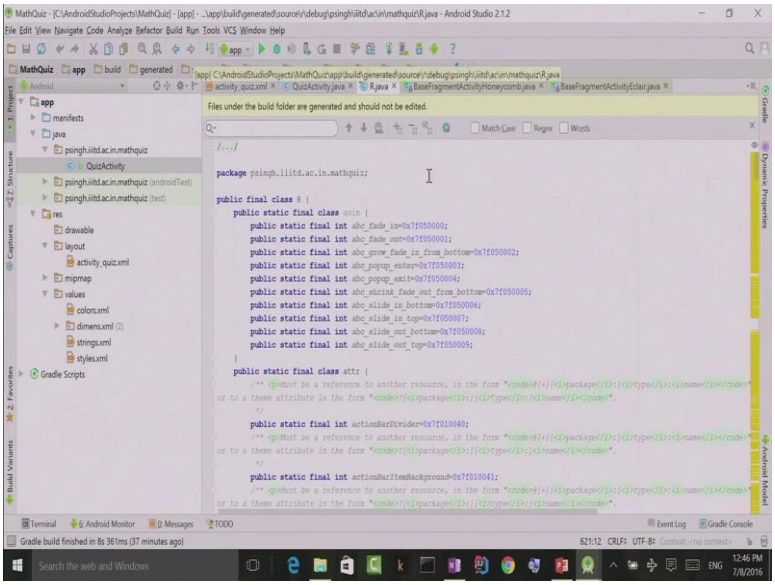
Hello each android program has an R file let us look at the R file that we have created for our program. As you will see that in the quizactivity.java file we have a function called on create which is calling another function called set content view. Set content view is taking a parameter R.layout.activity quiz now what is this R.layout.activity quiz?

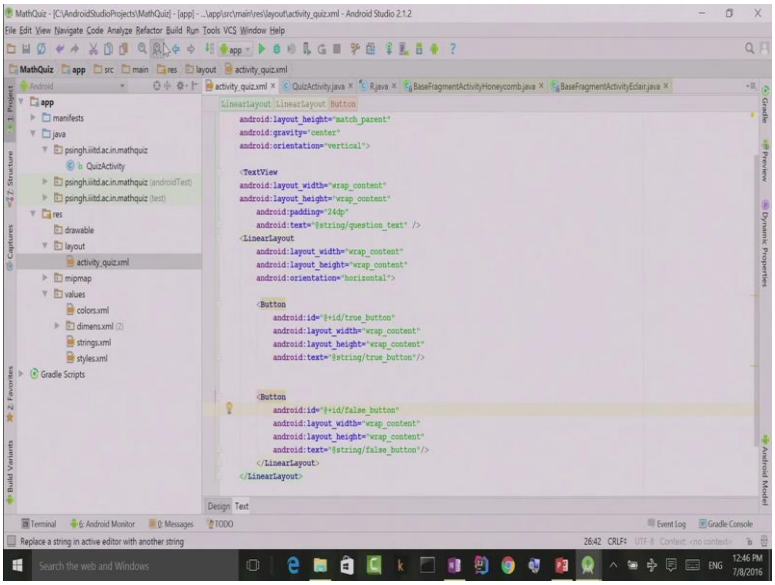
As I told you earlier that your android program consist of codes and resources and a resource is anything which is not code. So if you are using an image file in your code that is other source, if you are using an audio file, if you are using an xml file all those are called resource. And each resource has a resource ID infact you have the layout which you are using is also a resource. And this is the resource ID for layout. And we will create ID for the buttons that we have created in our example earlier.

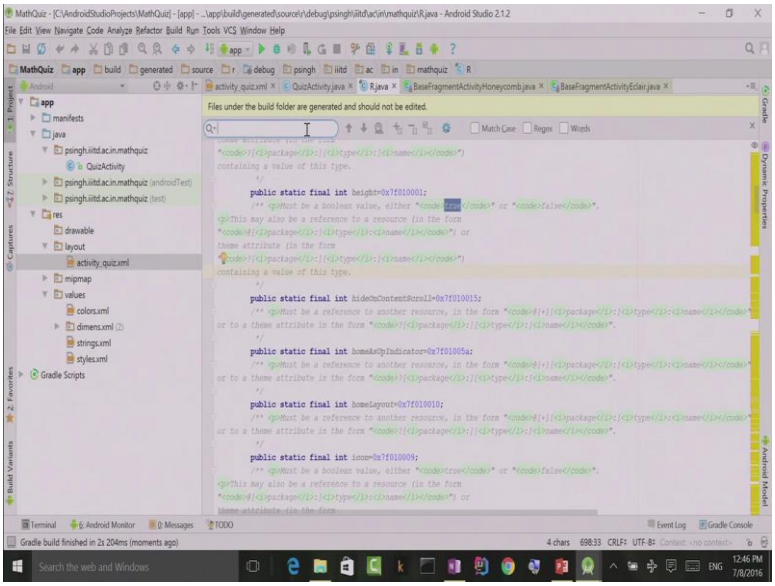
(Refer Slide Time: 29:29)

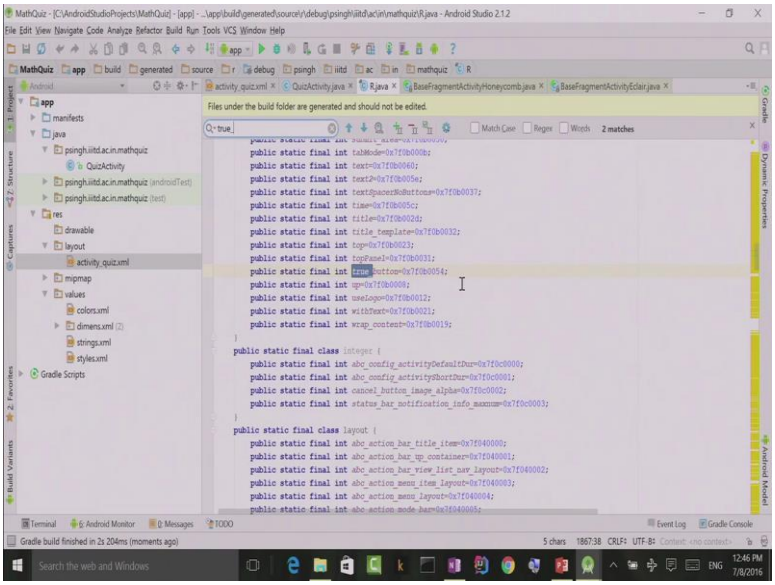


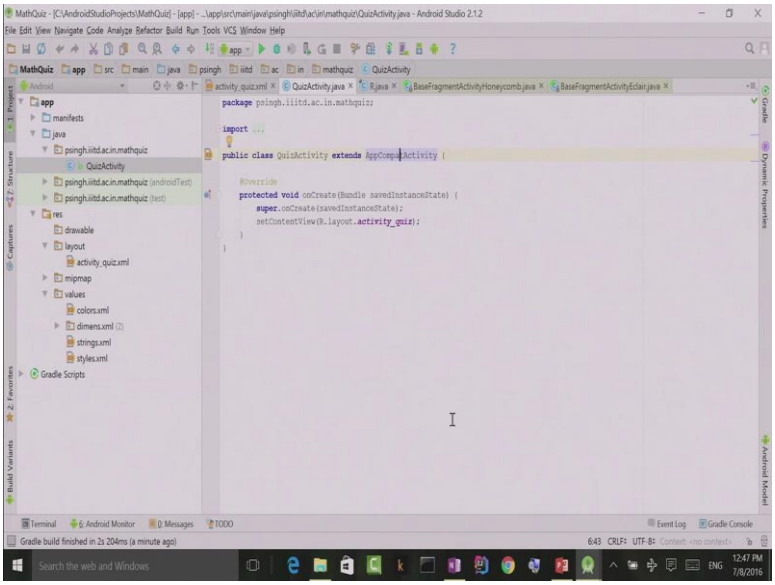


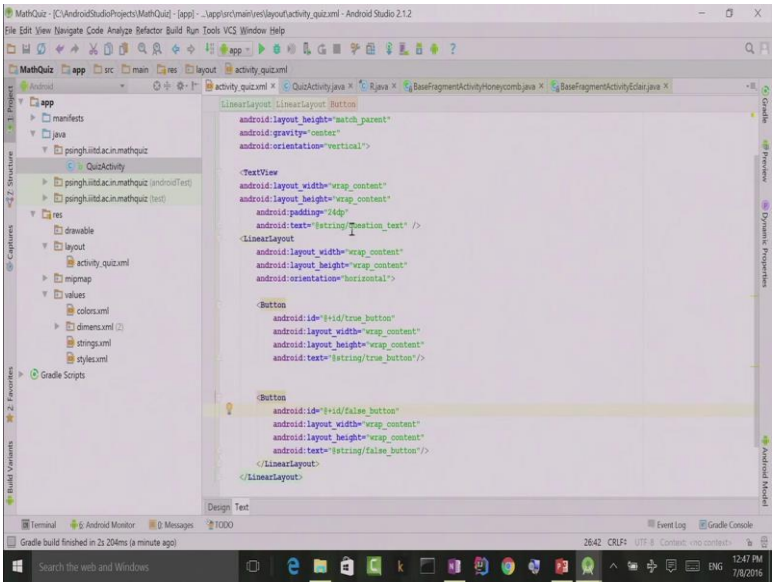



MathQuiz - [C:\AndroidStudioProjects\MathQuiz] - [app] - ...\app\src\main\res\layout\activity_quiz.xml - Android Studio 2.1.2
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
activity_quiz.xml
QuizActivity.java
R.java
BaseFragmentActivityHoneycomb.java
BaseFragmentActivityEclair.java
LinearLayout
Button
android:layout_height="match_parent"
android:gravity="center"
android:orientation="vertical">
<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:padding="24dp"
android:text="@string/question_text" />
<LinearLayout
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:orientation="horizontal">
<Button
android:id="@+id/true_button"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="@string/true_button"/>
<Button
android:id="@+id/false_button"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="@string/false_button"/>
</LinearLayout>
</LinearLayout>
Design
Text
Terminal
6: Android Monitor
0: Messages
TODO
Event Log
Gradle Console
Replace a string in active editor with another string
26:42
CRLF
Search the web and Windows
ENG
12:46 PM
7/8/2016


MathQuiz - [C:\AndroidStudioProjects\MathQuiz] - [app] - ...\app\build\generated\source\r\debug\psingh\iiitd\ac\in\mathquiz\R.java - Android Studio 2.1.2
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Files under the build folder are generated and should not be edited.
Match Case
Regex
Words
public static final int height=0x7f010001;
public static final int hideOnContentScroll=0x7f010015;
public static final int homeAsUpIndicator=0x7f01005a;
public static final int homeLayout=0x7f010010;
public static final int icon=0x7f010009;
4 chars
698:33
CRLF
UTF-8
Gradle build finished in 2s 204ms (moments ago)
Terminal
6: Android Monitor
0: Messages
TODO
Event Log
Gradle Console
Search the web and Windows
ENG
12:46 PM
7/8/2016

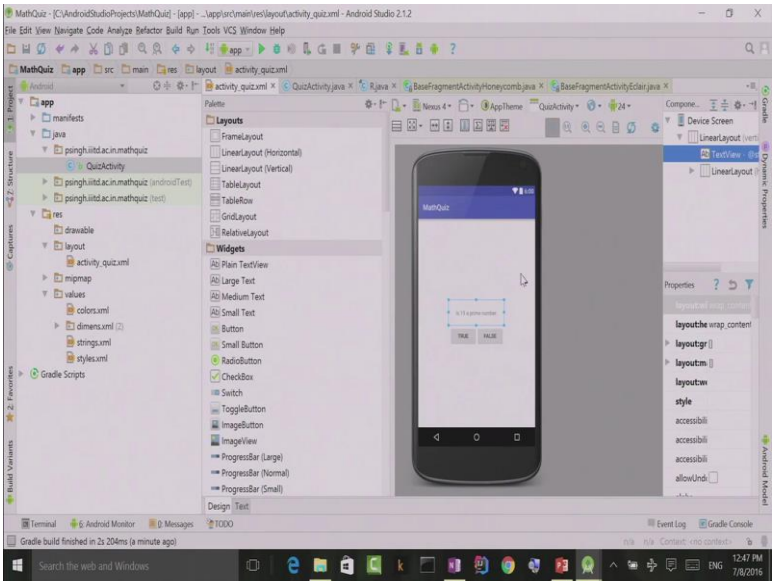

MathQuiz - [C:\AndroidStudioProjects\MathQuiz] - [app] - ...\app\build\generated\source\r\debug\psingh\iiitd\ac\in\mathquiz\R.java - Android Studio 2.1.2
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Files under the build folder are generated and should not be edited.
true
Match Case
Regex
Words
2 matches
public static final int tabMode=0x7f0b000b;
public static final int text=0x7f0b0060;
public static final int text2=0x7f0b005e;
public static final int textSpacerNoButtons=0x7f0b0037;
public static final int time=0x7f0b005c;
public static final int title=0x7f0b002d;
public static final int title_template=0x7f0b0032;
public static final int top=0x7f0b0023;
public static final int topPanel=0x7f0b0031;
public static final int true_button=0x7f0b0054;
public static final int up=0x7f0b0008;
public static final int useLogo=0x7f0b0012;
public static final int withText=0x7f0b0021;
public static final int wrap_content=0x7f0b0019;
public static final class integer {
public static final int abc_config_activityDefaultDur=0x7f0c0000;
public static final int abc_config_activityShortDur=0x7f0c0001;
public static final int cancel_button_image_alpha=0x7f0c0002;
public static final int status_bar_notification_info_maxnum=0x7f0c0003;
public static final class layout {
public static final int abc_action_bar_title_item=0x7f040000;
public static final int abc_action_bar_up_container=0x7f040001;
public static final int abc_action_bar_view_list_nav_layout=0x7f040002;
public static final int abc_action_menu_item_layout=0x7f040003;
public static final int abc_action_menu_layout=0x7f040004;
5 chars
1867:38
CRLF
UTF-8
Gradle build finished in 2s 204ms (moments ago)
Terminal
6: Android Monitor
0: Messages
TODO
Event Log
Gradle Console
Search the web and Windows
ENG
12:46 PM
7/8/2016





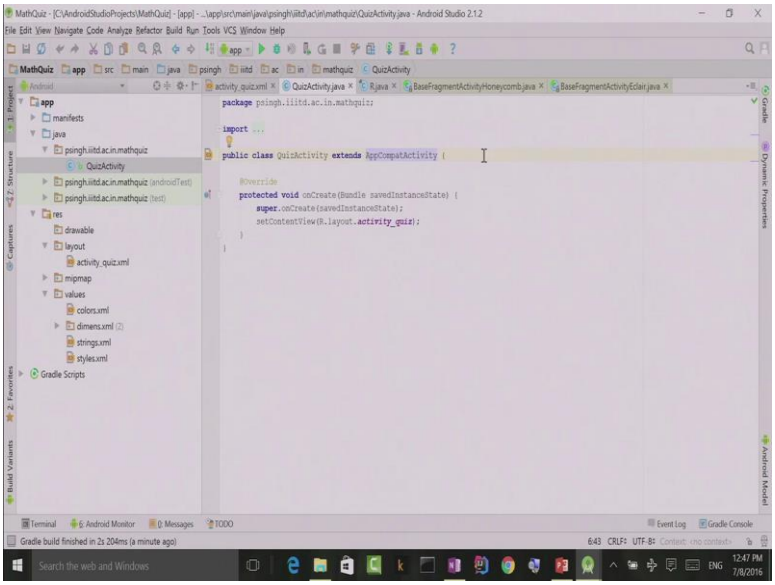
So let’s go back and let’s try to create some ID s for the buttons that we have. We save it (())(30:37) it. We go to R.java uhh lets build our project lets go to R.java and yes now you can see. That we have created new resource ID s for true button which was not earlier just a moment back and also for false button. Because we have created this ID s we can now use these resources in our code.

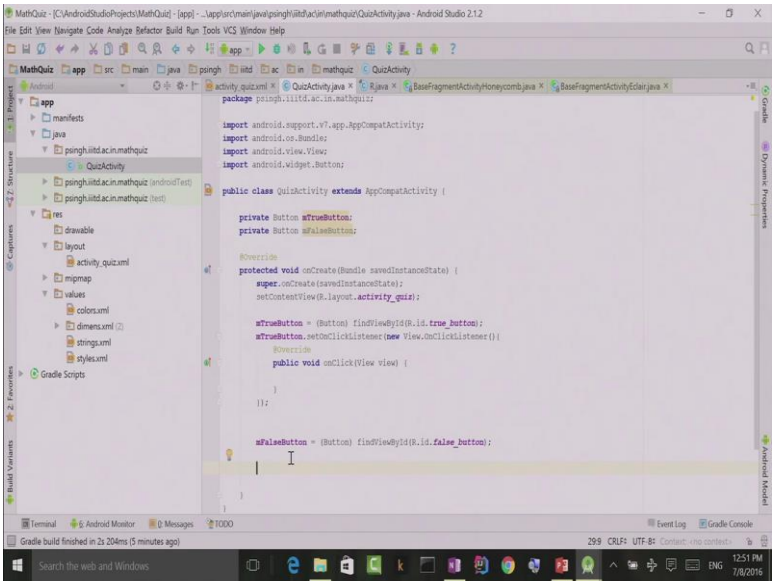
(Refer Slide Time: 31:43)



Let's go back to quiz activity. So so far what we have done is that we are displaying the question text and we have two buttons. But what we have not defined is what happened when we press these buttons. Actually if you run this program right now nothing will happen. So let's add some functionality to these buttons.

(Refer Slide Time: 32:00)





We will be adding some member variables to quiz activity which will define what happens when we press the buttons. So I am going to create two buttons. As you can see that android has started as you can see that android studio has started displaying some error. This is because I have not imported the right class files. It is say asking me to press alt and enter. Let me do that now if I go to import statements you can see that it has imported android.widget.button and now my errors are gone.

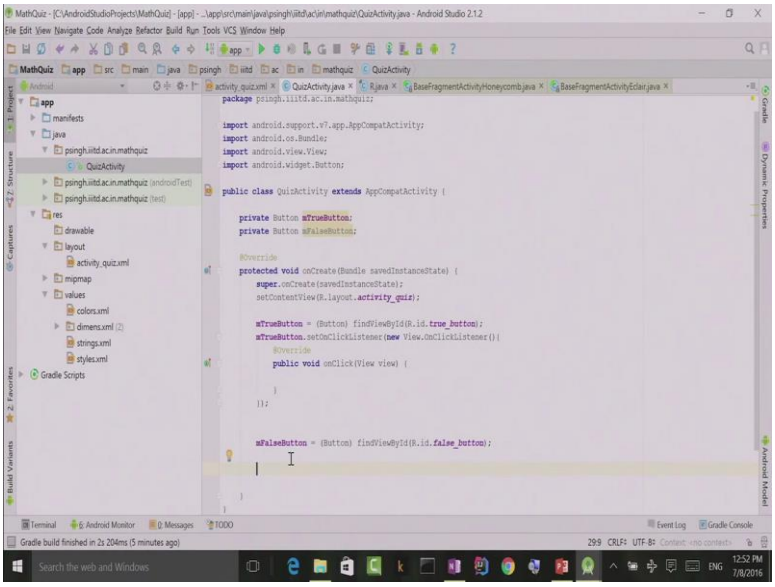
So I have created two buttons now I need to set actions to those buttons. Let me add some more code so that I can specify what happens when we press the button. I initialised. After that we have to set a listener. You may already be aware of the concept of listener and java. A listener is a piece of code that is run when an event happens. We want the event to happen when we click on the button. As you can see that different types of listeners are available I am looking for a listener that works when we click the button.

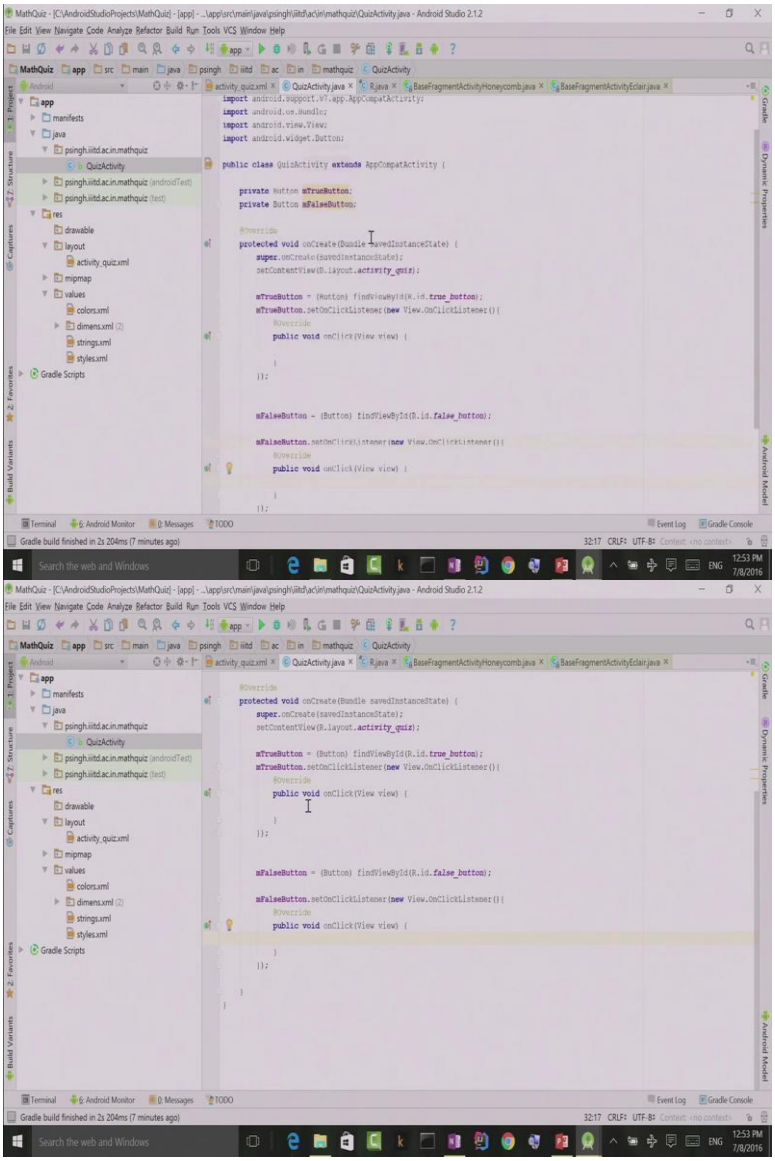
This is called set on click listener. Fine in our java revision we have talked about anonymous inner classes can you identify an anonymous inner class here? Well I have just described one right here. Anonymous inner classes saves us lot of code I could have done that by (de) by declaring another class then calling an instance of that class here but instead of that I chose to just create a code right here which satisfies my need.

If you are not familiar with anonymous inner classes please go back and read the java literature carefully there are multiple resources available on oracle website that explains what is an anonymous inner class and how to create and use anonymous inner classes. We will be

using anonymous inner classes very much while developing our android application because that saves us code and also make our code more readable.

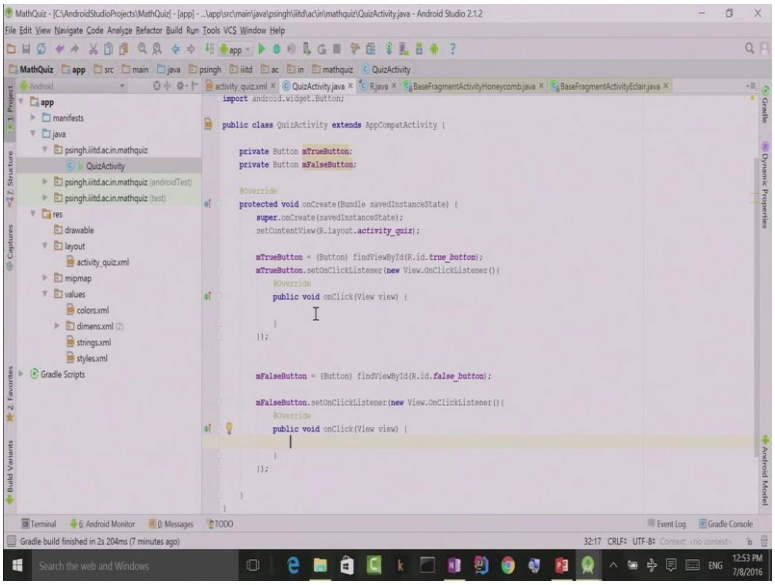
(Refer Slide Time: 36:40)

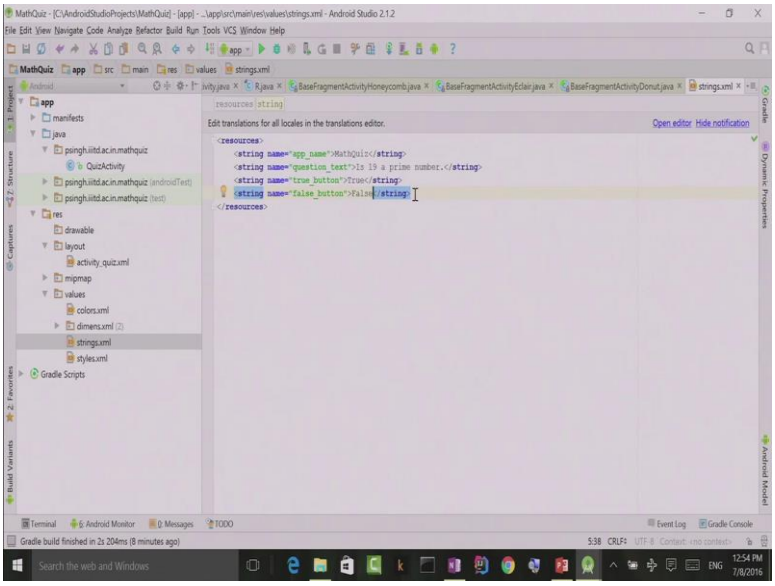




Let’s create the same listener for our second button. As you will see that currently we are calling the on click function but we are actually doing nothing. This is perfectly fine as far as the program is concern. However we will have to soon add some functionality which gets activated when a user actually clicks the button.

(Refer Slide Time: 38:08)

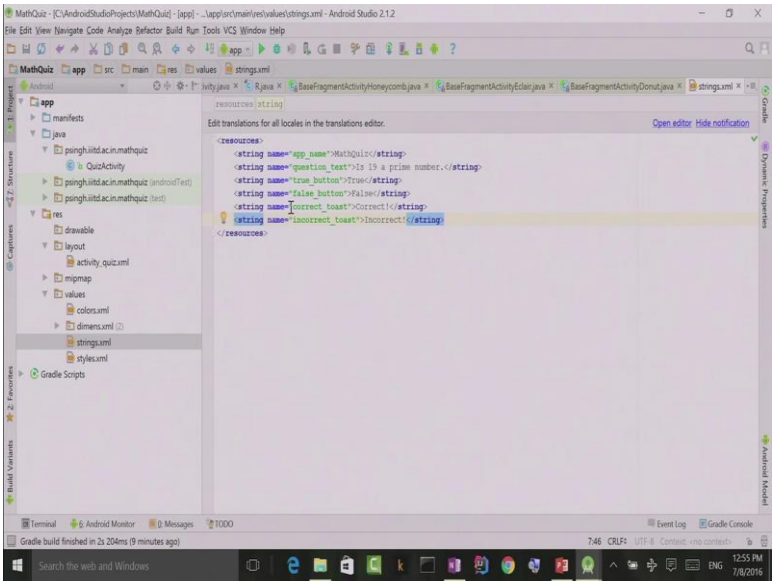


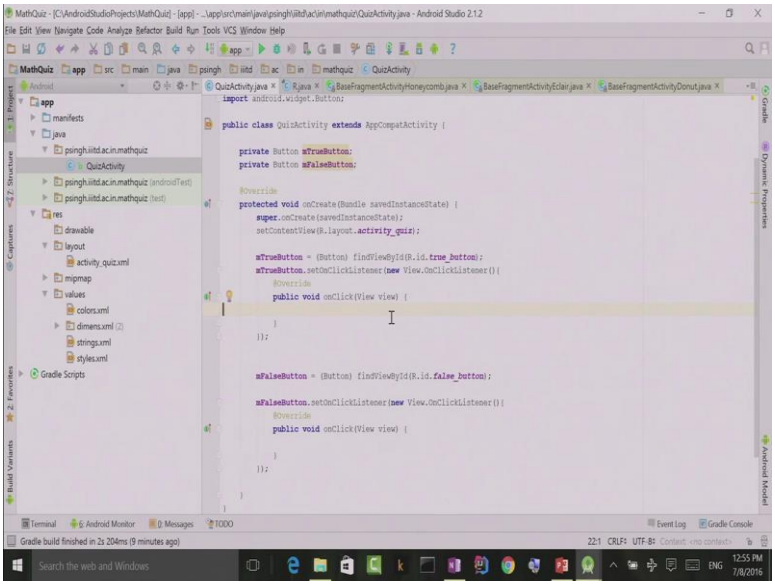


Okay so let's try to add some functionality here. What we will be doing is that we want actually want a pop up to come and tell us whether our answer was corrector incorrect. In android studio we call such pop up messages a toast. A toast is a short message that informs the user of something but does not require any input or action.

Let's create a toast for our program or may be two toasts. Number one because our toast will include a message we need to create more string resources that correspond to our toast. I am currently interested in two type of toast one which says correct another which says incorrect.

(Refer Slide Time: 39:58)

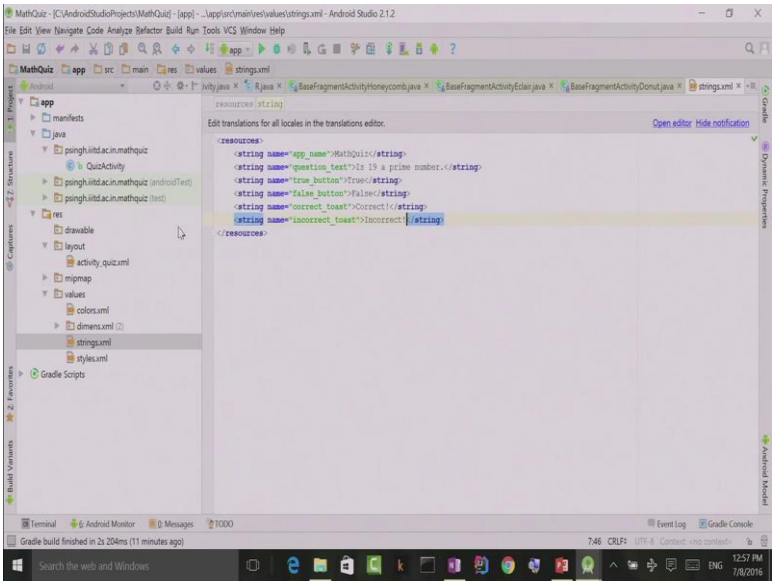


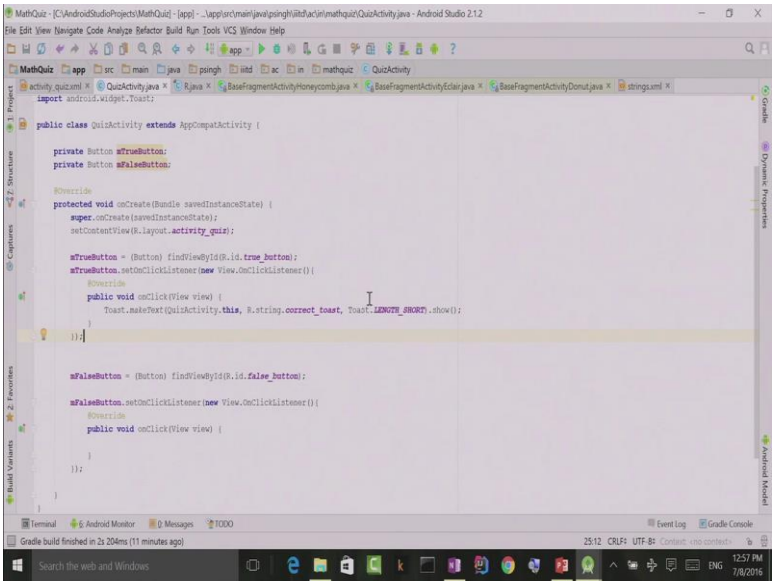


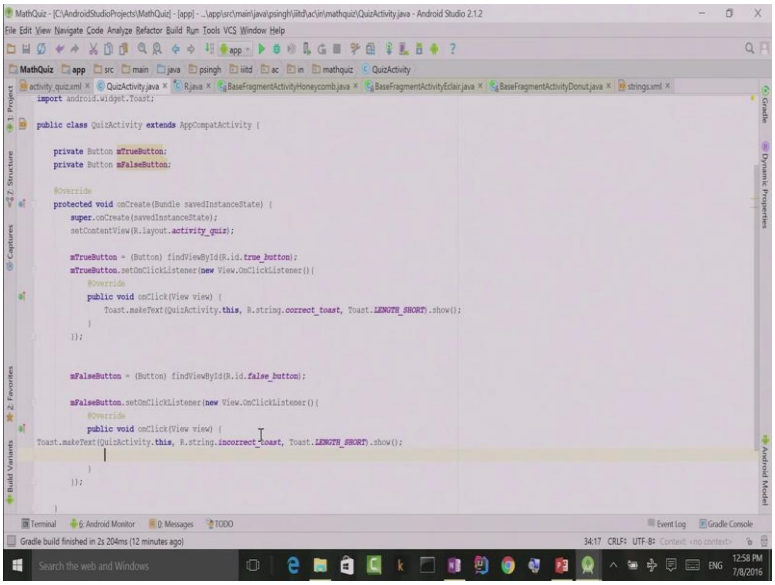
So now I have created strings resources that correspond to my toast message. I will go back and I will now create the toast. To create a toast you need to call a method called make text. Which is a static method and that's why it doesn't require an object. So as you can see I am directly calling it giving the reference of the class toast this method takes three parameters. First parameter is context second parameter is a resource ID and the third parameter is an integer variable of type duration.

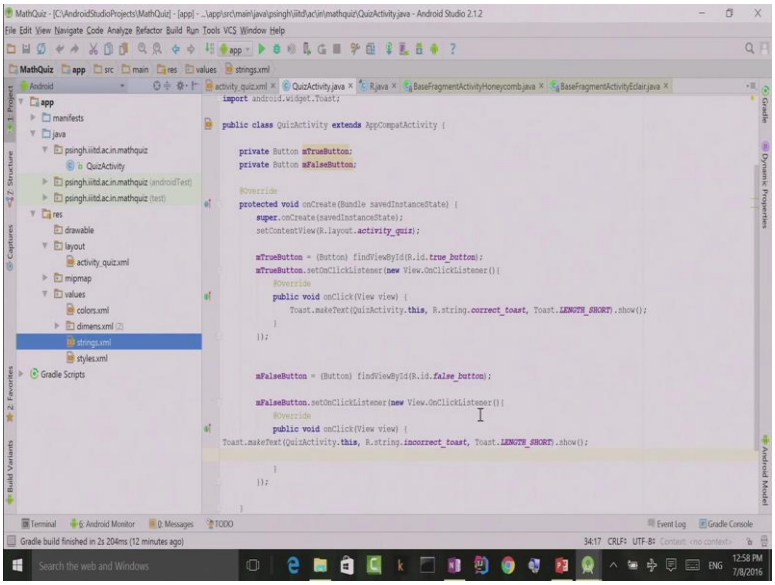
For the context I wanted in the context of the quiz activity I am using this for it. For the resource ID I wanted to show the resource ID corresponding to the correct underscore toast

(Refer Slide Time: 42:06)



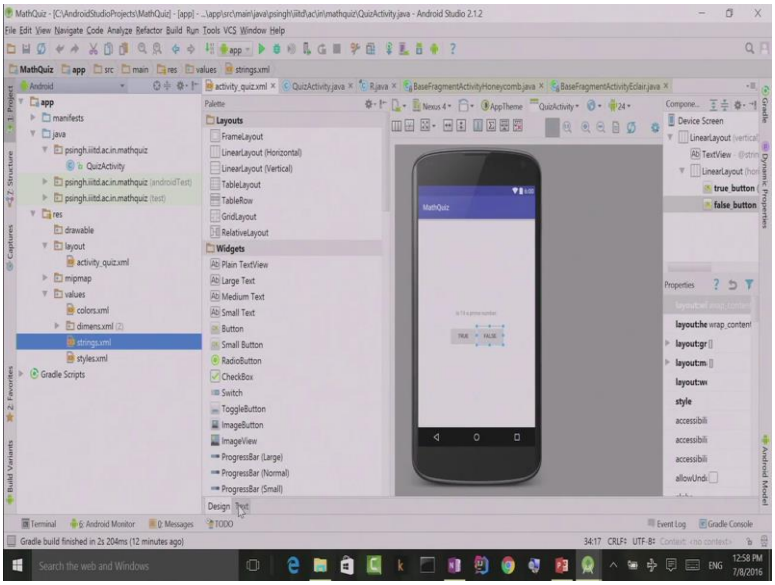


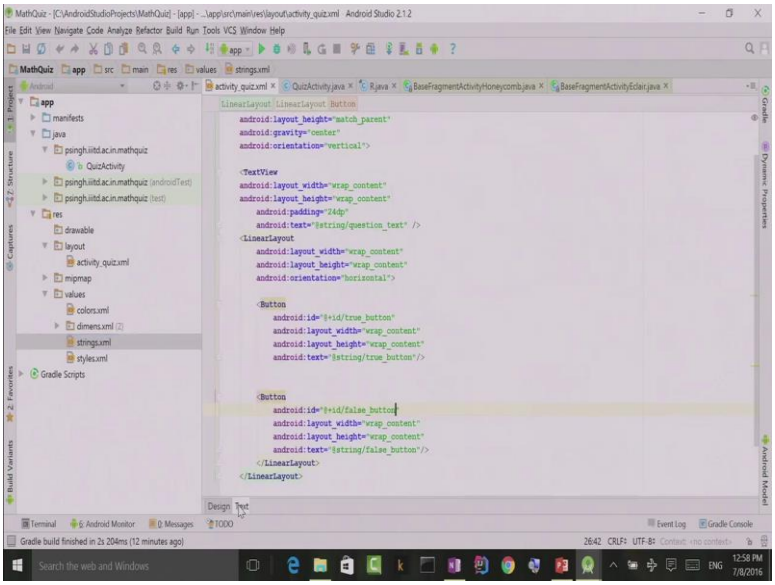


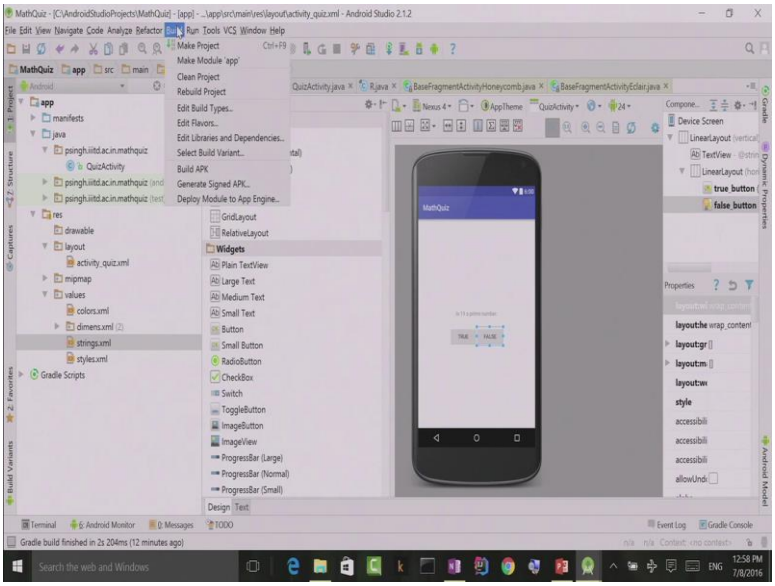


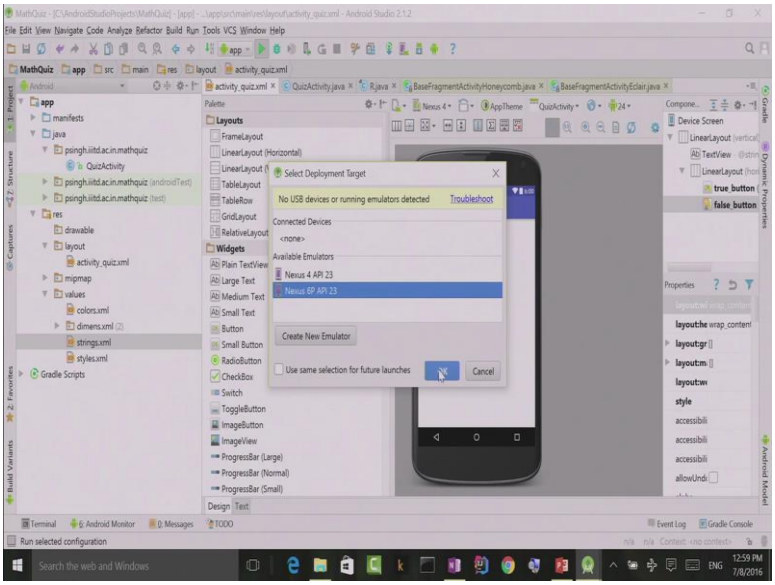
This one. And for the duration I want it to be a short duration let me create a similar toast for our another false message as well. Once I have created the toast I would like it to be displayed. Now our program is almost ready let's go back.

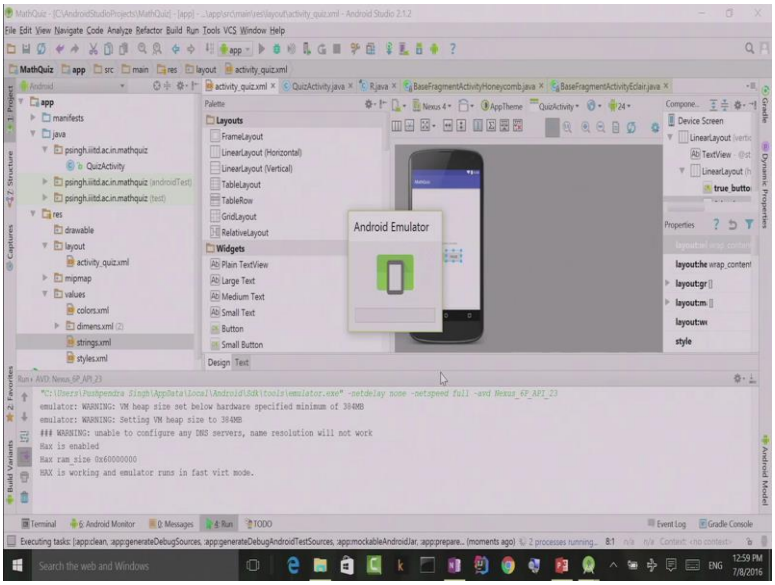
(Refer Slide Time: 43:13)




MathQuiz - [C:\AndroidStudioProjects\MathQuiz] - [app] - ...\app\src\main\res\layout\activity_quiz.xml - Android Studio 2.1.2
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
MathQuiz app src main res values strings.xml
activity_quiz.xml QuizActivity.java R.java BaseFragmentActivityHoneycomb.java BaseFragmentActivityEclair.java
LinearLayout LinearLayout Button
android:layout_height="match_parent"
android:gravity="center"
android:orientation="vertical">
<TextView
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:padding="24dp"
android:text="@string/question_text" />
<LinearLayout
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:orientation="horizontal">
<Button
android:id="@+id/true_button"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="@string/true_button"/>
<Button
android:id="@+id/false_button"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:text="@string/false_button"/>
</LinearLayout>
</LinearLayout>
Design Text
Terminal 6: Android Monitor 0: Messages TODO
Event Log Gradle Console
Gradle build finished in 2s 204ms (12 minutes ago)
26:42 CRLF: UTF-8:
Search the web and Windows
ENG 12:58 PM 7/8/2016

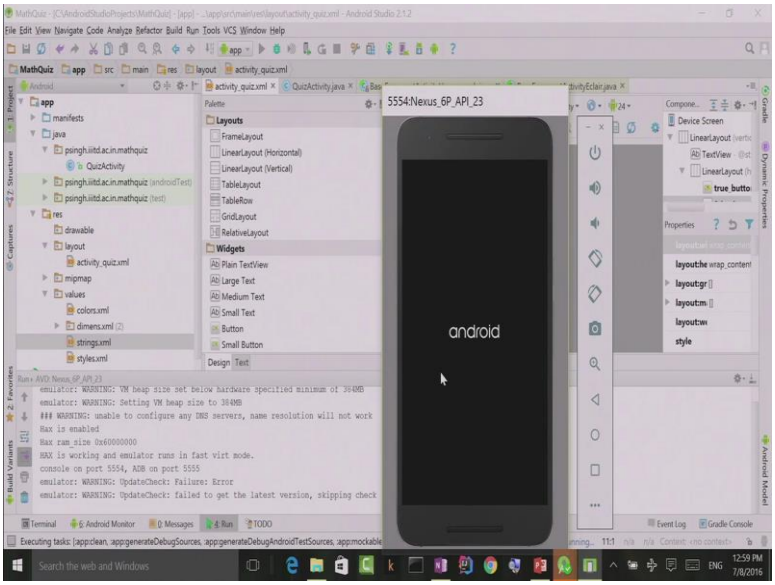

MathQuiz - [C:\AndroidStudioProjects\MathQuiz] - [app] - ...\app\src\main\res\layout\activity_quiz.xml - Android Studio 2.1.2
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Make Project Ctrl+F9
Make Module 'app'
Clean Project
Rebuild Project
Edit Build Types...
Edit Flavors...
Edit Libraries and Dependencies...
Select Build Variant...
Build APK
Generate Signed APK...
Deploy Module to App Engine...
GridLayout
RelativeLayout
Widgets
Plain TextView
Large Text
Medium Text
Small Text
Button
Small Button
RadioButton
CheckBox
Switch
ToggleButton
ImageButton
ImageView
ProgressBar (Large)
ProgressBar (Normal)
ProgressBar (Small)
Design Text
Terminal 6: Android Monitor 0: Messages TODO
Event Log Gradle Console
Gradle build finished in 2s 204ms (12 minutes ago)
Search the web and Windows
ENG 12:58 PM 7/8/2016


MathQuiz - [C:\AndroidStudioProjects\MathQuiz] - [app] - ...\app\src\main\res\layout\activity_quiz.xml - Android Studio 2.1.2
File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
Palette
Layouts
FrameLayout
LinearLayout (Horizontal)
TableLayout
TableRow
GridLayout
RelativeLayout
Widgets
Select Deployment Target
No USB devices or running emulators detected
Troubleshoot
Connected Devices
<none>
Available Emulators
Nexus 4 API 23
Nexus 6P API 23
Create New Emulator
Use same selection for future launches
OK
Cancel
Design Text
Terminal 6: Android Monitor 0: Messages TODO
Event Log Gradle Console
Run selected configuration
Search the web and Windows
ENG 12:59 PM 7/8/2016



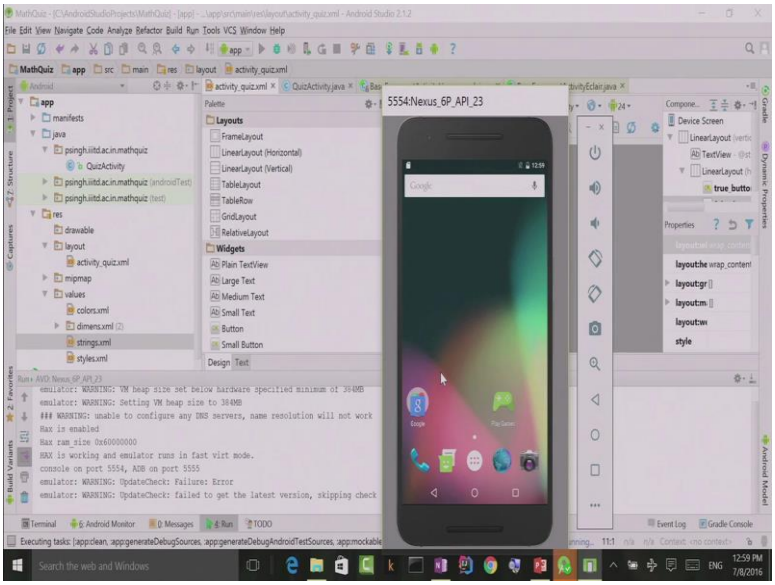
Everything looks fine. I can make project and we may want to now run it. Again we have a choice to running it on emulator or on a phone. I would run it on emulator so that you can see it.

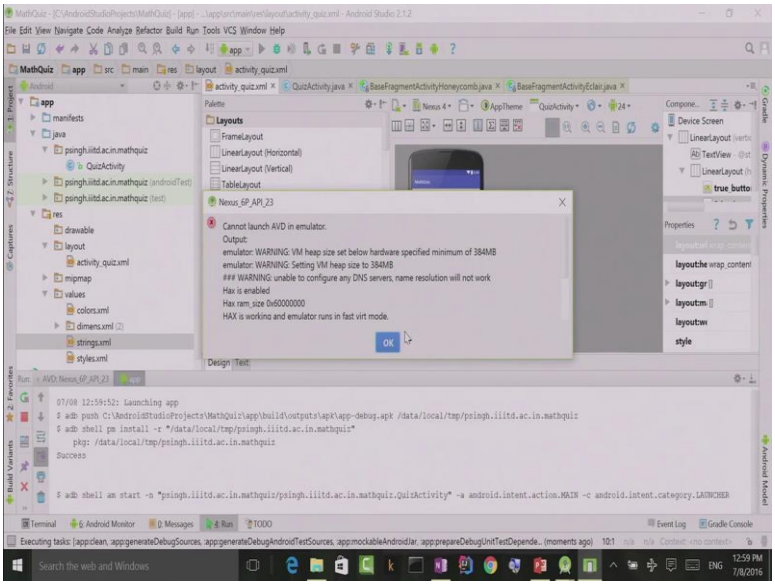
(Refer Slide Time: 43:53)

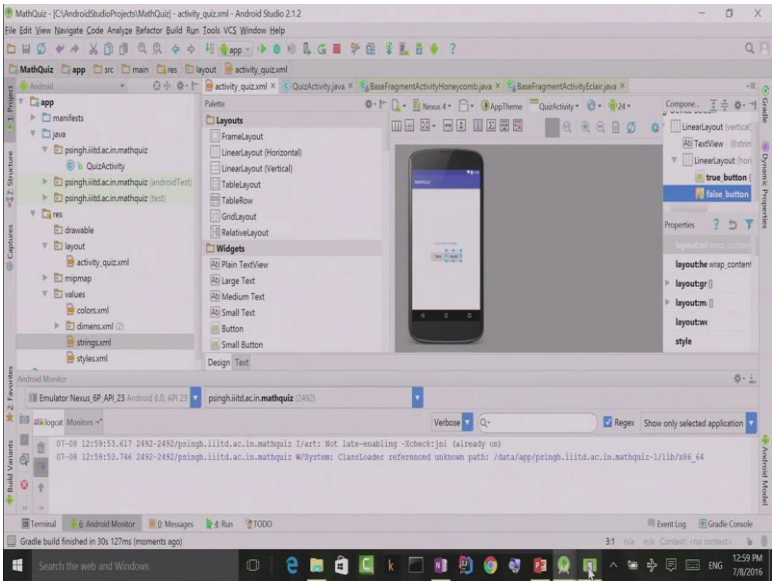


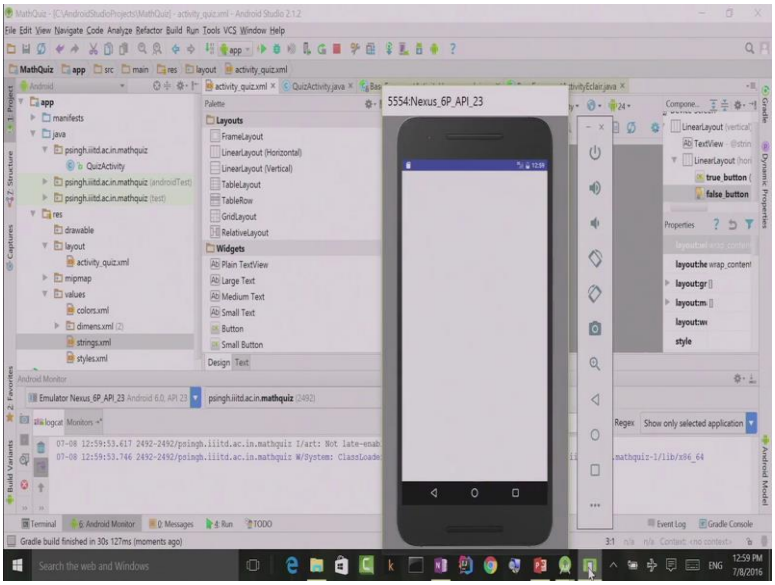
Our emulator is starting

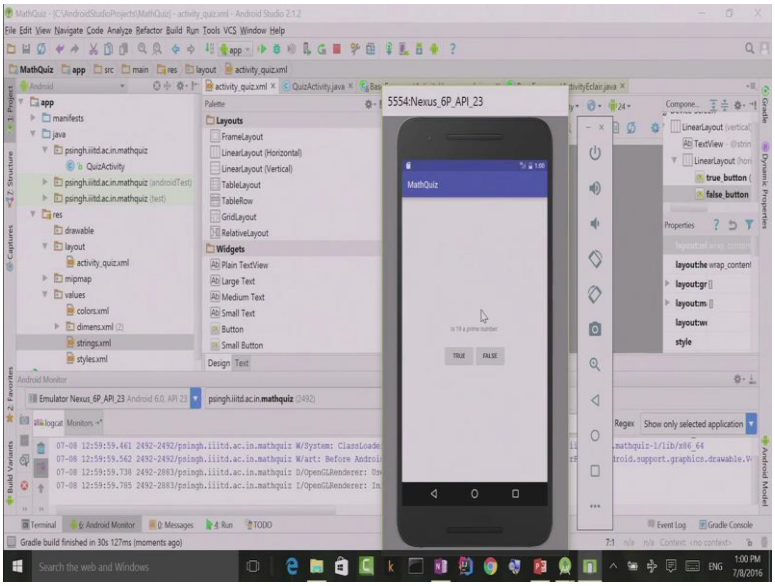
(Refer Slide Time: 44:16)





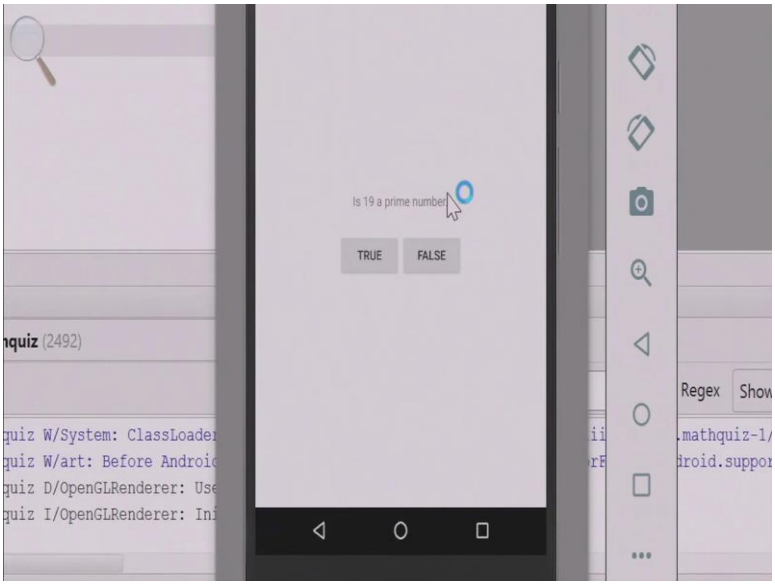






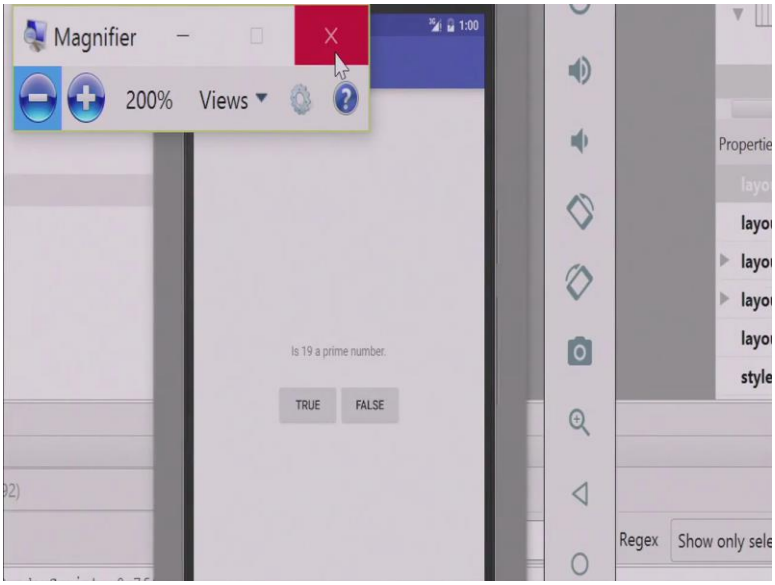
Turning same warning. We need not to worry this is specific to my device and here is our application. So yeah is 19 a prime number? Let me magnify it for you.

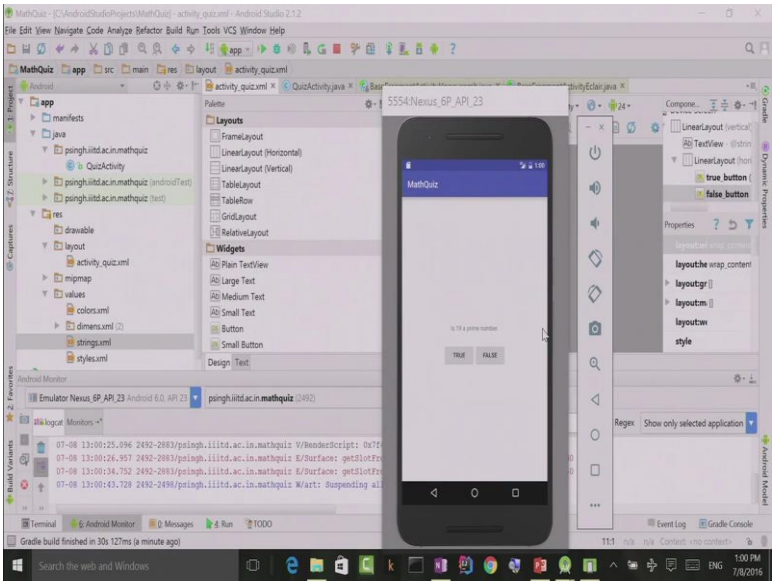
(Refer Slide Time: 44:33)



So this is the this is our application this displaying our question. Which we asked and it is giving us two buttons. Which we can press suppose I press true ohh displays correct. 19 is prime number. Just for checking let me press false it displays incorrect and as you see that after short duration our toast disappears.

(Refer Slide Time: 45:05)





You may have used many such applications which required you to interact with buttons displays some information etc. Today you have successfully created an application using a layout which was different than what was given by default by android studio. You changed it from xml file you create it resources in this string file directly.

You created two buttons you created there IDs. You check the IDs in the R. Java after that you created the behaviour of the buttons and then you deployed your application successfully on a emulator and run it. Congratulation this was your first step towards creating our simple very basic android application but which is usefull. In the next classes you will learnt to develop more and more advanced application in android studio. Thank you!