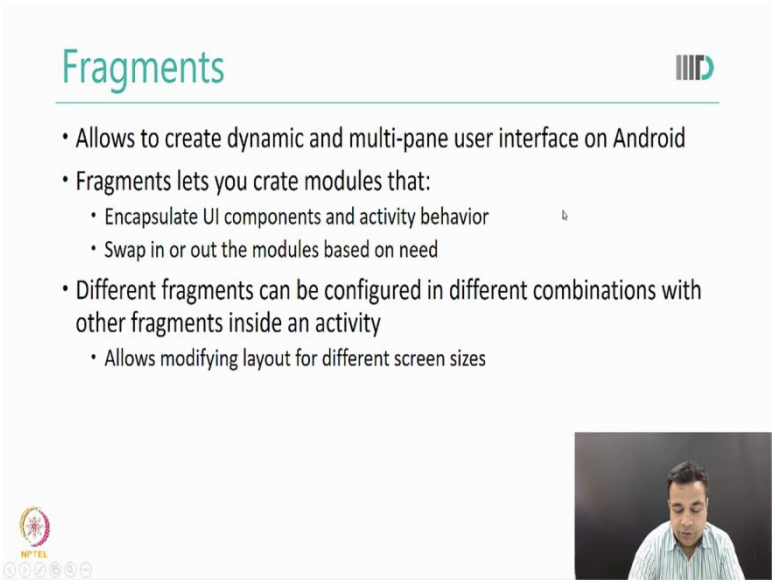
Mobile Computing
Professor Pushpendra Singh
Indraprastha Institute of Information Technology Delhi
Lecture 23
Fragments

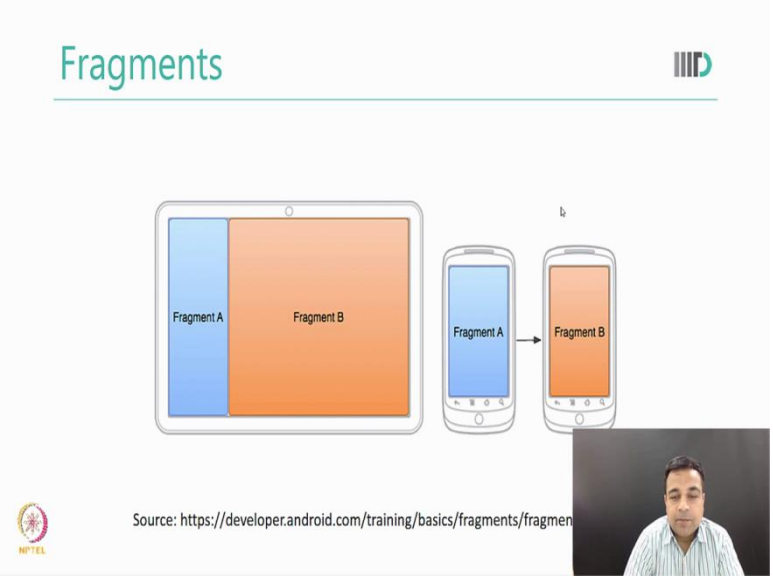
Hello, so today we will be discussing fragments. Fragments are very important in android programming and they were introduced from android (())(0:22) but later on they were extended to support even android devices which were released earlier. So what does a fragment do?

(Refer Slide Time: 0:35)



So fragments allow us to create, dynamic and multi-pane user interface on android, what does it mean? Let us see a image.

(Refer Slide Time: 0:48)



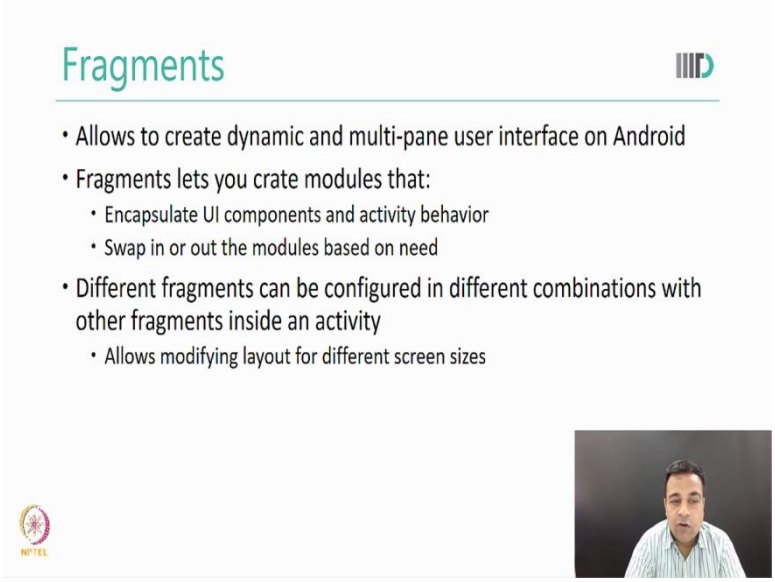
Now, you see, so when we started having tablets which had a much larger screen, there was a need to have different types of displays for tablets or anything which has larger screen and on phones which usually have smaller screens. The same logic applies even for phones with larger screens and the phones with the smaller screen. So suppose you have a very simple application which displays your name as a list and when you click on any of the name then it displays basic information let us say your photographs, your name , your roll number, your program that you are studying etc. So, there is a list of names and for each name there is detailed information.

Now, if I have to develop this android application just on phone, a good approach would be that initially my phone screen display is a list and if I click on a list, I move to another screen where I see the details then I can click on the back to the list click on another name and see the details of another name and so on. This is a natural interface for a phone; however suppose we have a tablet. Now, we can have the same interface that we have on the phone on the tablet as well but on tablet a more natural interface would be something like this, where let us say either the left or the right side displays the list and then there is a another portion which can be used to display the details, at the same time when the list is being displayed.

So, when the user clicks on any of the name on the list, they can see the, display right there. So, this looks like a more natural interface for the tablet. Now, whatever you have learned so far if you try to emulate that in presenting a display like this for phone, it has a straight forward you can create activities and one activity takes into another activity and you can create on on a list item here and that shows you another screen, you can comeback press on

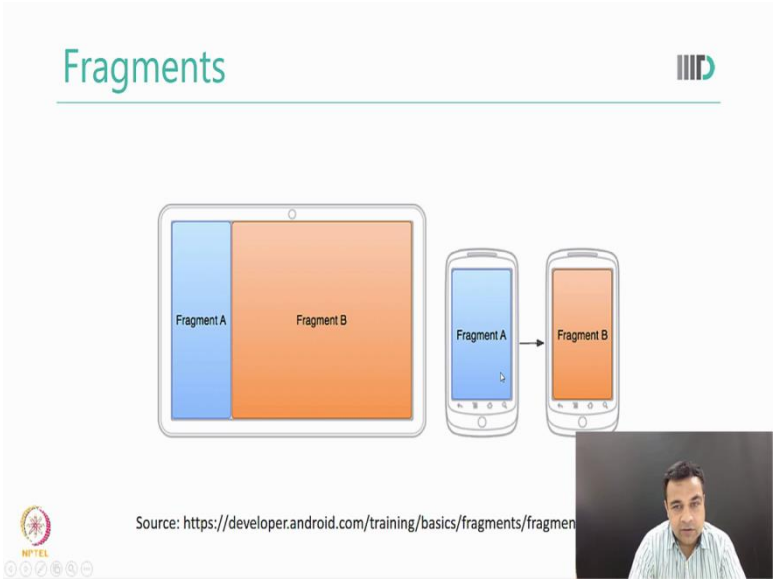
another list item and then you can see the another screen however; if you want to move this code to the tablet and want to show something like this, this is not possible because the activity life cycle in the way activity has been implemented in android does not allow you this functionality that you can display two activities where you can move back and forth between, between the activity and impact to the same parts. So, here what you kind of need is something like, there is a bigger activity and that it involves may be 2 or more sub-activities but this is not allowed in the activity, so android makes it available as a fragment and not activities.

(Refer Slide Time: 4:17)



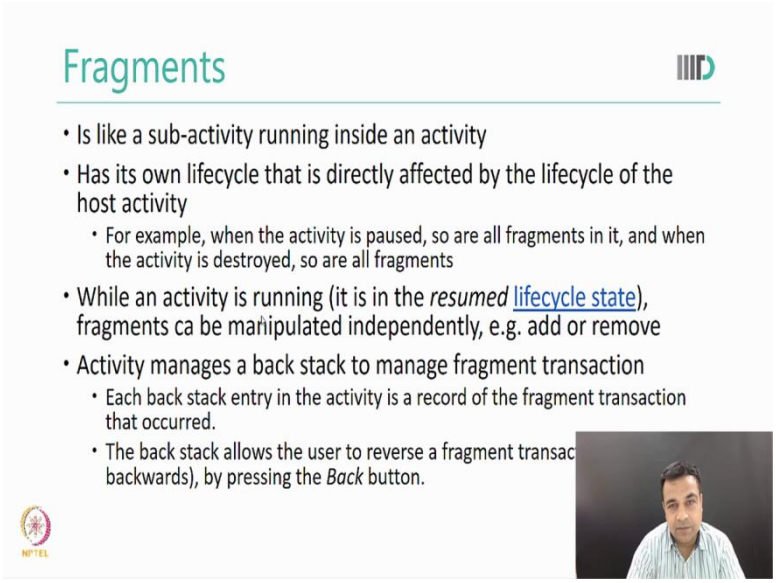
So, let us come back. So, fragments are which allows to create dynamic and multi-pane user interface just like the one we discussed. So, the fragment lets us create modules that encapsulate UI components and the activity behavior and then it allows us to swap in or out the modules based on our need. Different fragments can be configured in different combinations with the other fragments inside a single activity, which then allows modifying layout for different screen sizes, now let us comeback to our picture again.

(Refer Slide Time: 5:29)



So the way I will implement it is that I will have only one activity and my one activity will have, let us say my list is of 10 names. So, my one activity will have 11 fragments. The one fragment will display the list and the other 10 fragment correspond to the details of the individual items of the list. Now, when I am developing for the mobile at that time, my activity will only invoke one fragment and then another fragment but when I am on a tablet my activity will have two fragments at any given time. So, while we cannot show 2 activities as part of one activity, we can show 2 or more fragments as part of one single activity. So, in a way your fragment is actually like a sub-activity that is running inside an activity.

(Refer Slide Time: 5:59)



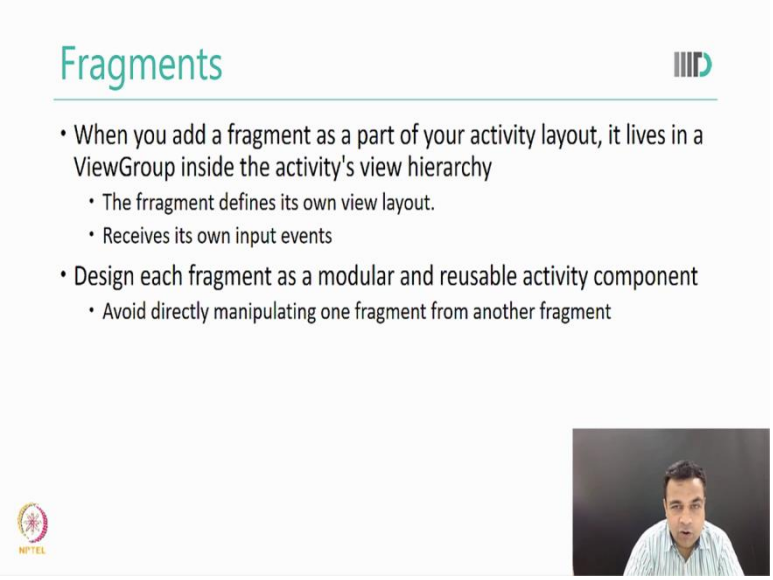
So, now you can see that this gives us a lot of freedom when you want to develop, suppose our interface is different and we want to display the small thumbnails of all the list items here, that is not very difficult we can display so many fragments inside the same activity or we want to make this just display do 2 or 4 or 6 on any combination, the fragments and the activity together provides us this functionality. So, always think a fragment like a sub-activity and that multiple fragments can be use inside an activity in different type of configuration as you need them. So, let us comeback to the details again, so each fragment like a sub-activity running inside an activity. Each fragment just like an activity also, has own life cycle.

So, in the previous classes we have read about activity lifecycles. So, activity lifecycles goes something like on Create, to start, create, resume, pause, stop, (())(7:14) similarly, the fragment also has its own lifecycle. Now, fragments lifecycle because fragment is running inside an activity. Fragments lifecycle is directly affected by the lifecycle of the host activity. So, if you pause the activity, fragments inside it will also be paused, if you destroy the activity fragments inside it will also be destroyed. So, where do we manipulate fragments?

So, we manipulate fragments when the activity is in the resumed state or what we called in the running state. So, when activity is after on resume if you remember from the activity lifecycle that is this activity is running at that time we can manipulate the fragments because that is when we have the full freedom, at that time we can add fragments dynamically, we can remove fragments dynamically, we can replace fragments dynamically, we can choose different combinations of fragments.

So, fragments along with activity allows us the flexibility that we want to have when we are dealing with devices which have different the screen sizes or when we want to develop UIs the multi-pane and (())(8:34). The activity which is hosting these fragments also, manages a back stack to manage fragment transactions. So, the back stack is maintained by a activity. So, suppose you add it fragment one, fragment two, fragment three. Now, you are pressing back button, the back attack will allow you to go back to the fragment that you get, we call it a fragment transactions and the back stack entry is a record of the fragment asked (())(9:08). So, when you add a fragment that is a fragment transaction and when you remove a fragment that is a fragment transaction and your back stack entries are record in these transactions. So, using the back stack we can allow the users to reverse the fragment transaction, by pressing the back button.

(Refer Slide Time: 9:42)



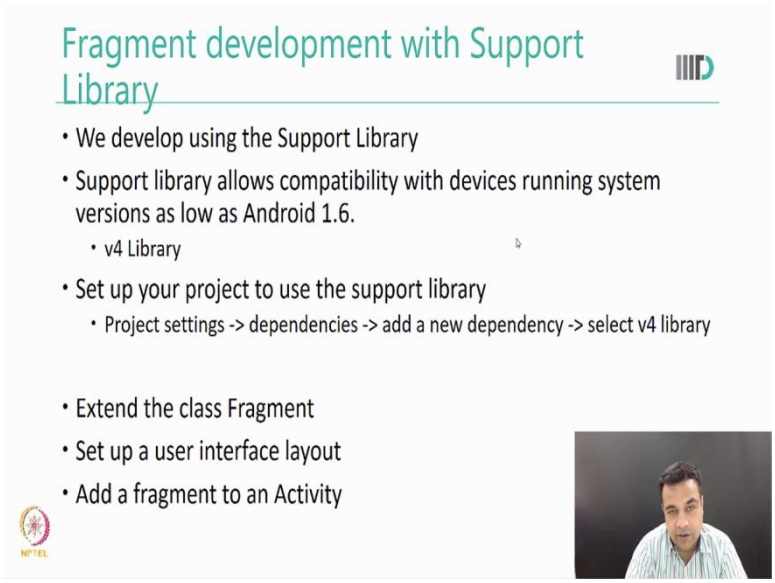
Now, when we define activity we define usually a layout and if you remember from our previous lectures a layout is a combination of view group and view objects. So, when we add the fragment as part of our activity layout the fragment must live somewhere and it lives in the view group inside the activity's view hierarchy, okay. So, the fragment defines its own view layout and it receives in own in input events, in fact when you add a fragment, the layout of the fragments is inflated and sort of inserted into the view group of the activity.

Now, the next part is more of a design considerations you should keep in mind when you are writing android application and it is that you should design your each fragment to be as modular as possible so that you can use it as a reusable activity component with different activities and without depending on something which binds a fragment to a particular activity. So, try to make your fragment always as modular as you can, they should be like an individual independent component that you bring in together for one activity or for another activity without affecting the functionality of the program. So, one important the step is that we should not engage in direct manipulation of one segment from another.

Now, let us look again, on the same example, here you can see in that our activity that we are developing, we will have a kind of view group for the phone and this will only display the list while for this, the layout of the activity would be such that that in that layout, it is possible to insert 2 fragments. So, for phone we will have to have a layout file which only inserts one fragment, for the tablet we will have to have a layout file which inserts two fragments. So, that is how you insert a fragment into an activity to create the overall view. Now, there are

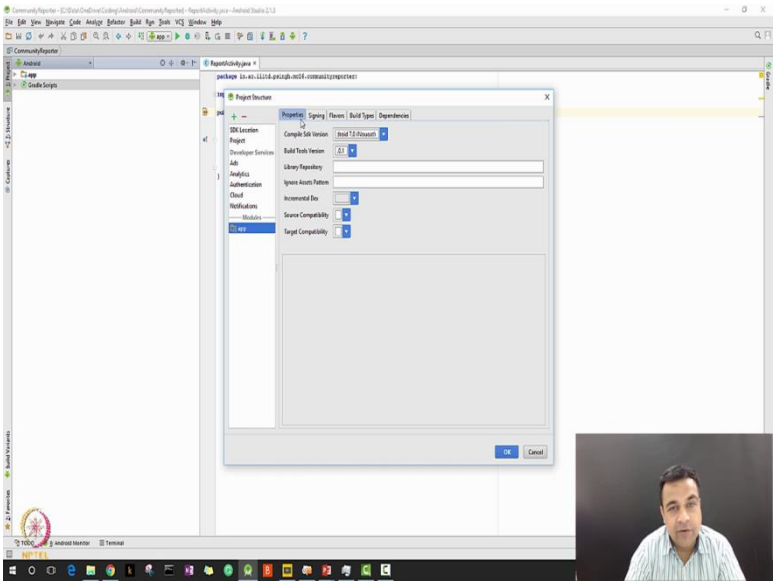
few more important things with fragments. The most important thing is that we, develop fragment what we call using a support library provided by android.

(Refer Slide Time: 12:18)



So, android provides 2 implementation of fragments; one in the support library and the one in the android OS and contrary to our intuition, we use the version which is provided as support library and there are two major reasons of doing this. The reason number one is that the support library allows compatibility with devices running system version as low as android 1.6. The name of this library is v4 library and if you develop your fragments with the v4 support library which is recommended by Google and android developer community you can support devices which are running as low version as android 1.6 that is very very old android devices as well. So, how do you develop your fragment with support library? You will have to do some extra work and let us see, what you need to do. Let me open the android studio and show you how to add the correct support library.

(Refer Slide Time: 13:59)



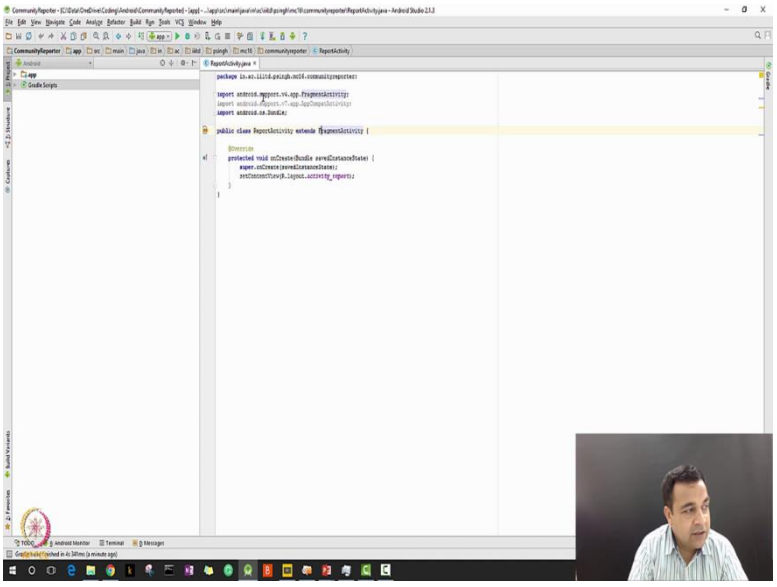
So, this is our android studio. Let me go, into the file. So, go on the top left, click on the file if you click on the file, you can come down and you will see menu item called project structure, click on the project structure. The project structure will open a dialog box for you. In this dialog box towards the end there would be something called app under modules, if you click on app, the dialog box will change into a tabbed dialog box and if you read the tabs, it reads as properties, Signing, Flavours, built types and dependencies. Currently we are interested in the dependencies, so I will go into the dependencies and I will see that what dependencies are there. I can also remove a dependency or add an independency. Actually, for this purpose I have already added the v4 library which I was asking you to add. So, I was asking you to add, the v4 library and as you see I have already added the v4 library.

So, let me first first remove it and show you so, if you are starting a program from scratch, you will probably see a dependencies window like this. So, click on the plus item the green plus item which is on the right side, if you hover your mouse the item will say add, click on this item, it will show you a smaller dialog box library dependency, file dependency, module dependency, I will go into the library dependency. Now, it will show me all the libraries that is I can add it into my program. In this if we look at from the beginning then the second item, which reads as support dash v4 com.android.support:support-v4:24. 2.g that is the v4 library that we were talking after, I select it and I press ok, now I can see that my dependency has been added and that is all I have to do.

If you look at this dependencies window you will also, see the other dependency which are already there. So, let me open it again under the project structure and the dependencies. So,

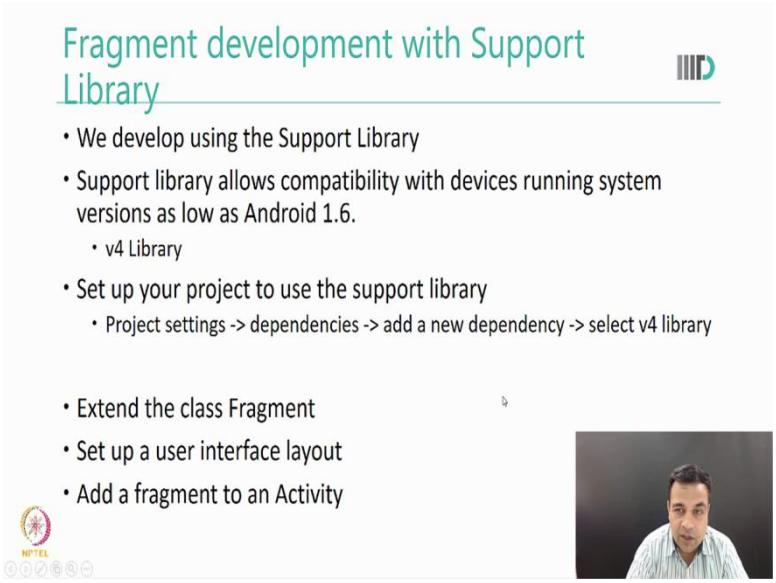
you will see that there is a dependency name here call app compact dash v cyber if you remember the name app compact, then you will see that for your activities, the app compact is necessary, some of you have put a question that they are finding this error that app compact library not found or something similar. This basically means that your dependencies has not been created properly.

(Refer Slide Time: 17:30)



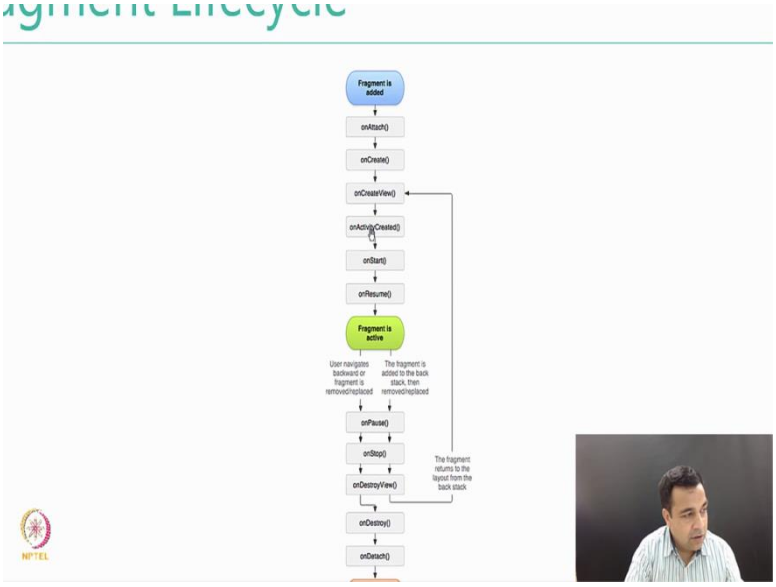
So, you can repeat the same procedure for adding dependency for app compact library. Now, because I have added the support library, when I extend an activity for fragment my input massage is of support.v4.app.fragment activity. So please after you have added the support library and when you before you start creating fragment make sure that you are also importing the same (())(17:47) support library for your program. Now, let us go back to our presentation.

(Refer Slide Time: 17:58)



So, as we develop using the support library for two major reasons, one that it provides backward compatibility with far lower version of android and number 2 that support library gets updated for number of times, so you will have more latest features in support library. Now, how we will develop a fragment? We extend the class fragment, we setup a user interface layout and we add a fragment to an activity. We will see all this in a program but first we want to cover some basics of the fragment.

(Refer Slide Time: 18:35)

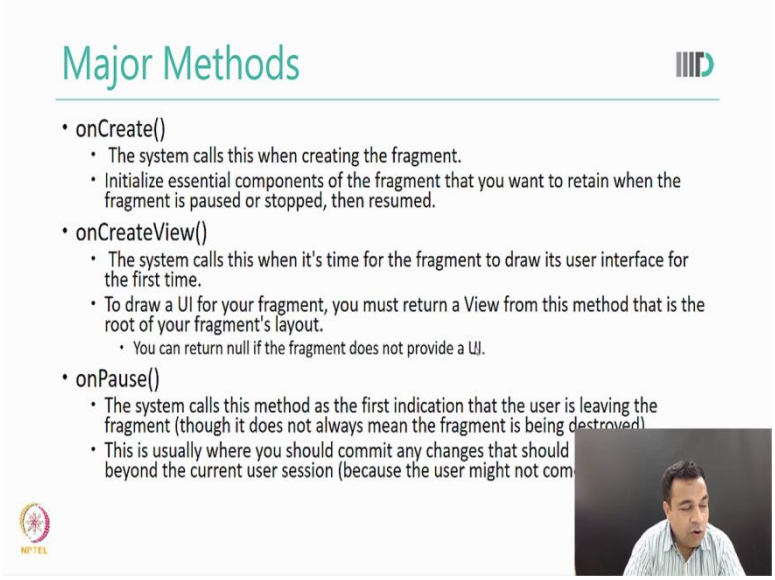


Now, let me first increase it little bit, what I am trying to show you is the fragment lifecycle. So, this is very similar to I hope now all of you can see, some of you have mentioned that you find difficulty in seeing the screens, so I have increased the size. So, this is fragment

lifecycle, it is starts when fragment is added, then it calls some methods, call back methods just like the activity.

The important method for us here is the onCreate view and that is where we will do our action. Similarly, onAttach, onCreate, onCreate view, onActivity check it, onStart, onResume, just like activity fragment also gets active after the (())(19:28) onResume and then onPause, onStop, onDestroy view, onDestroy onDettach. So, these are the call back methods which are the part of the fragment lifecycle. Later on we will see that how these methods corresponds with the methods of the activity lifecycle, among these methods, there three methods which are important, one is onCreate another is onCreate view and the another is onPause.

(Refer Slide Time: 20:05)



So, the onCreate method is called when the system is trying to create fragment. So, you must initialize essential component of the fragment in the onCreate method, then the second important method is the onCreate view. This is called when it is time for the fragment to draw its user interface for the first time and in order to do a UI for your fragment you must return a view from this method that is the root of the fragment or root of your fragments layout, later we will see through program how do we do this. Please note that a fragment need not to always provide a UI, in that case you can (())(20:54).

(Refer Slide Time: 21:31)

The third important method is onPause. The system calls this method when the user is planning to leave, this is similar to what the onPause method of their activities and because the user may leave, you must commit any changes that should be persistent. So, these are the 3 major methods as usual I have used following references. All of this information is available on the developer.android.com website and it has been taken from that. So, if you have any doubt you will go to the developer.android.com and treat yourself. In the next video we will cover more details about fragments, slightly see in how the fragment is made use in an activity and then in more other videos we will learn a program like fragments and see them in action. Thank you.