**Programming in Java**
**Prof. Debasis Samanta**
**Department of Computer Science and Engineering**
**Indian Institute of Technology, Kharagpur**

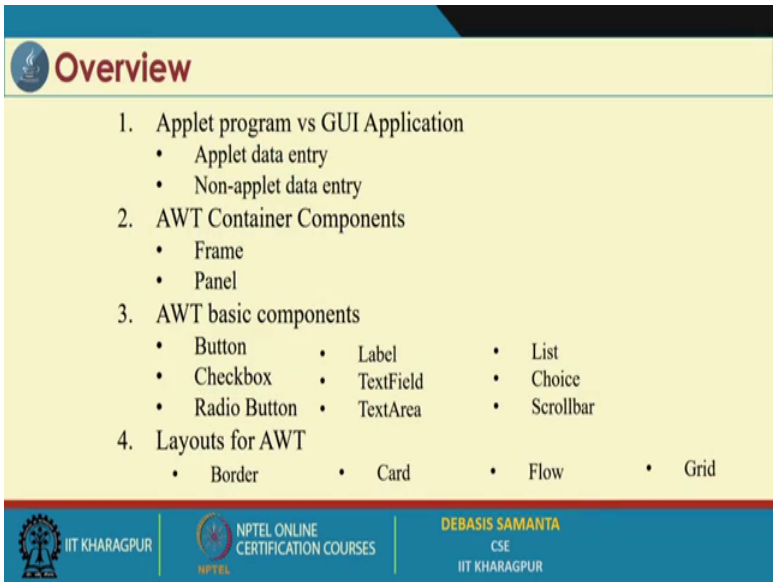
**Lecture – 41**
**Demonstration -XV**

So, this is a demo class, in this demo we will see exactly whatever the topics that we have covered in the last 2 modules related to AWT programming.

(Refer Slide Time: 00:25)


And in this demonstration we will discuss about few things right.

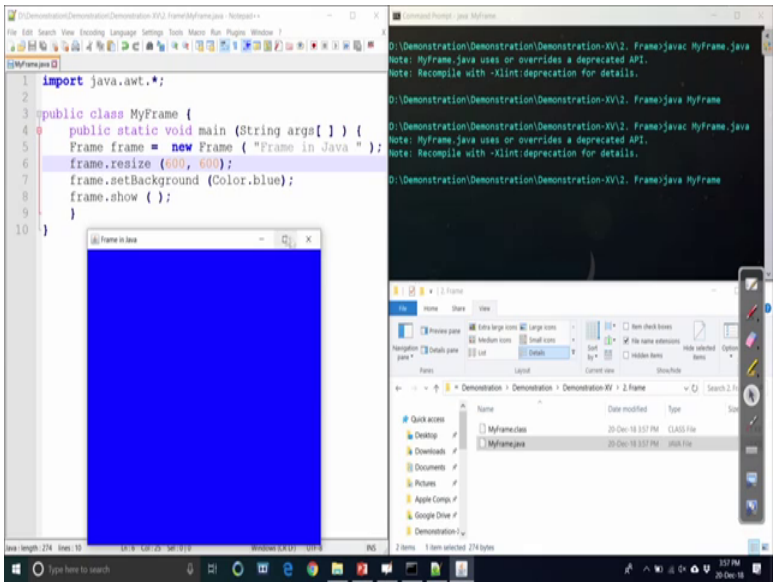
(Refer Slide Time: 00:27)




So, how the applets can how the different GUI elements can be included in a container in a components and the different containers also like say frame, panel and the different components GUI components like button, checkbox, label list, choice scrollbar menu item and everything and also how the layout can be managed by the user layout mangers that is there in AWT and finally, we will also see exactly an application point of view how these kind of things will be useful to develop our application. So, this is the overall what is the agenda for this demonstration.

Now, let us have the first program; in this demonstration you will be able to see how we can create a Frame and then the same Frame can be added and then the Frame can be executed and we will just discuss about non applet versus applet program I will just discuss about the non-applet program right now. So, here basically we want to create a Frame and the same Frame can be the displayed on a screen.

(Refer Slide Time: 01:35)




So, this is a program as we see here. So, this is a program as we see here in the first lines as we see we just higher the class basically we define, this class is basically MyFrame the name of the class as it is shown and then the main ex basically application. So, we have used the main method here and in this main method we create a class in this method we can see we create a Frame here and then this Frame is basically passed with an argument at String the Frame in Java; that means, as you know Frame is a container with a title.

So, it is basically in the title bar it will appear this one and this is the size we have to resize the Frame that how it will be there and background that we have discussed here, the color of the Frame will appear with the blue background and then finally, the Frame dot so, indicates that Frame is visible.

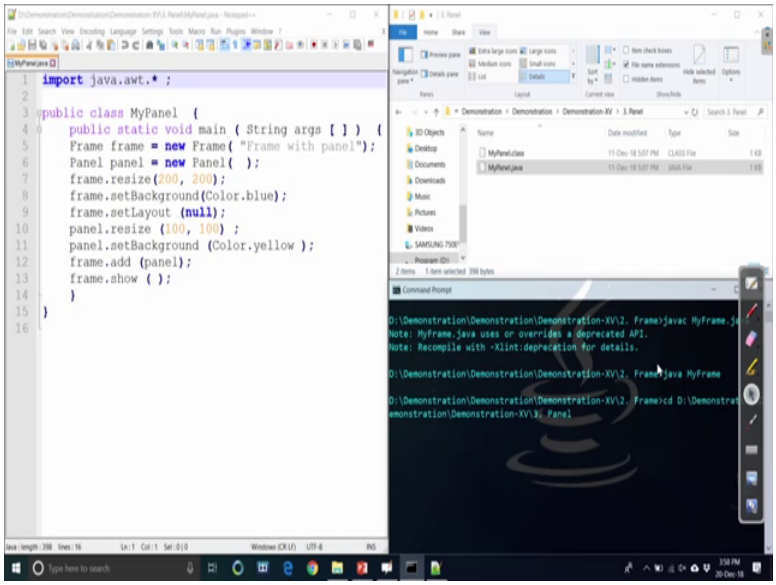
Now, so, if you run this program using java interpreter and then you will be able to see the output that it will show to us. So, this is basically a Frame. Now I am just going to little bit doing the changes here in the main program here in the resize method just 500, 500 that we have given it, now let us change it to 500 300 versus 500.

Now, we will see the size of the Frame will be changed automatically. So, you can see that size of the Frame has been changed it is like this. So, if we change it then size of the Frame will be changed again 100 versus 500 then you can understand how the size is affecting because it is even minor change. So, very small changes in the pixel, now you

have to shape the program class why it is not. So, giving the different sizes you have change the size save and then compile it they are not compiling right.

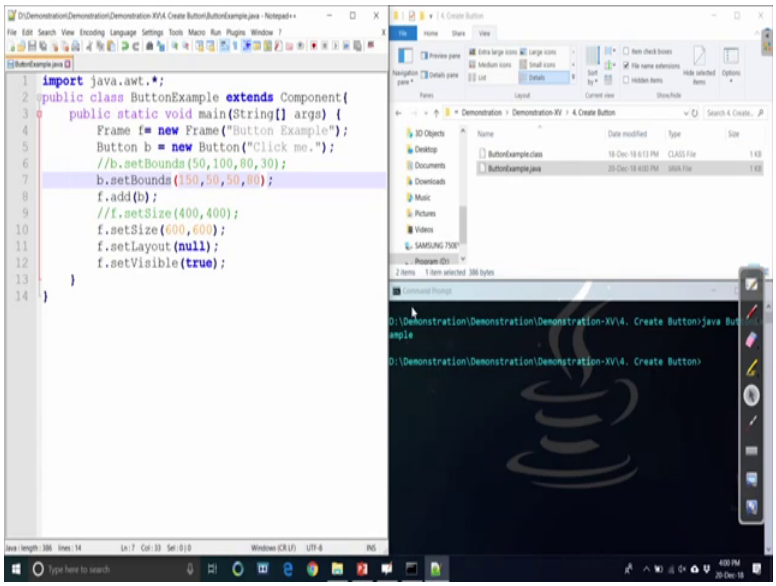
As we change the size and we have just seen that how the size has been changed. So, accordingly we can for example, if we see the 5 if we change this 600 versus 100, then you can see how the changes will be there. So, that just for you can test up to your own also you can see it how the different sizes of the Frame can be created and the same can be displayed it. So, this is for 600 100; 600 100; 600 100; 600 versus 100 if you give it then its fine 600 600 is not good. So, 600 versus 100 listen to me then yeah exactly it is like this. So, you can understand that if we change the size parameter it will change this one.

(Refer Slide Time: 04:14)




Now so, this is the about how we can create a Frame and our next demonstration related to the creation of a panel, just like the Frame creation will be panel Frame and panel basically the two things are mostly same except that the panel does not have any title bar and here we can see we first create a Frame and we include one panel inside the frame.

(Refer Slide Time: 04:33)




So, this is the basically method as we see first statement to get a frame and then Frame having the title say Frame with Panel and then next means create the panel with the only constructor it has there no string to be passed here like the frame.

And then resize Frame and then resize panel everything we are doing here and then if we run this program and here we can see we have not mentioned that the location of the panel where it should be. So, by default it will go to the left and then top most corner actually. However, we can paste the here you can see the panel the color background has been chosen as yellow and then background of the Frame is blue and we can change it. If we change anyone parameter accordingly the things will come into the change effect. So, this is about creating panel and then frame.

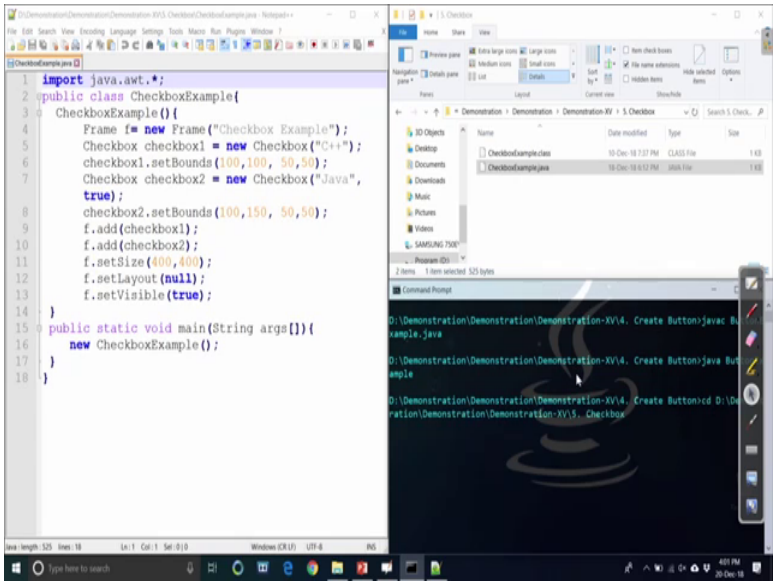
Now, let us have a very simple example from the components, as you know there are many components that is possible in java awt we will just give a discussion about the Button. So, Button however, a Button can be created and then how this Button can be resized and then the same Button can be added. Now in this example as we see we first create a Frame and then Frame inside this Frame we are planning to create one Button we create a Button b as a Button and having the level on this Button as Click me and then we decide the setBounds it is basically first two values indicate that x and y location and then the next 2 values indicate width and height of the Button if we change anyone value automatically Button will be changed.

And then size is basically setSize is basically give the size of the Frame actually because it is a related to the Frame only and then setLayout it is basically now we have to make it there we are not using any null layout manager. So, that is a null and then f dot set value it is 2 that this is visible actually; that means, the whole the things the container con containing a Button will be displayed to us.

Now, so, let us run this program I will be able to see the output here as you see this one there many more things also can be done a Button can be with different color different context and everything for this is the different constructor that we have discussed in the Button class can be used for this purpose.

Now, setBounds as we see the size of the Button can be controlled here. So, as we have mentioned the last two Buttons are re regarding the width and height if I change into 50 and 60 like. So, it is basically change of the Button size will be changed automatically say may be yes. Now, let us see we have changed the Button size is there, yes now we are save it ok.

(Refer Slide Time: 07:33)




So, compilation is done and then we are running it as we see here the Button is created compared to the previous example as we see size has been controlled there and you know the size of the Button also if we use a layout manger, that can be also controlled by h gap and b gap that is another example we will discuss while we will give a demo of the layout manger.

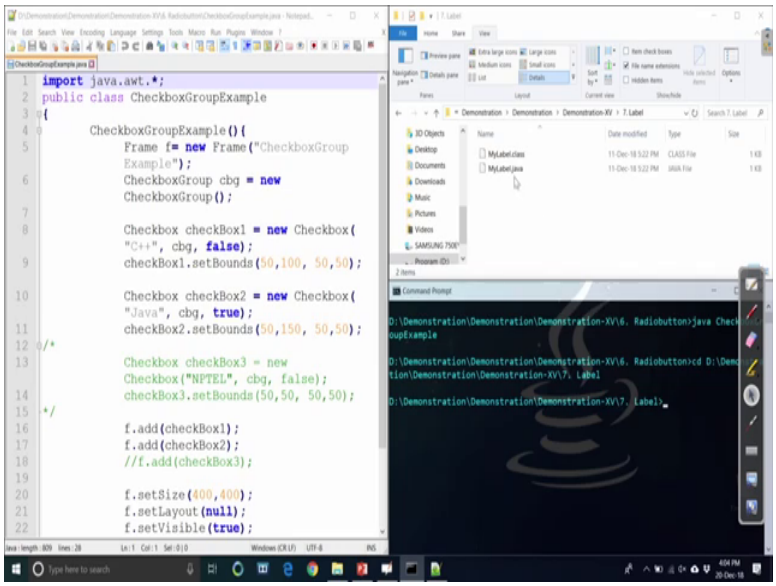
Now, let us have the next demonstration on the Checkbox. So, how the Checkbox can be created as you know the checkbox basically contains a number of items to the did there and then it will have the checkbox check window actually where the check can be either ticked or without untick that is basically check or uncheck can be there now here the say creating the checkbox as similar to the Button actually in that case we should take the help of checkbox plus.

So, using the Checkbox plus we have created a checkbox1button here and then title is basically C plus plus, then another checkbox also we have created like Java and then we just add this checkbox in to the frame. So, f dot f checkbox1 and f dot add checkbox2 it basically says that checkbox1 and checkbox2 is added into the Frame and the rest of the thing is same as regarding the configuration of the Frame.

Now, here you can see for the checkbox2 we make the second parameter as true whereas, for the checkbox1 we did not mention any second parameter by default it is false. So, if the true means this checkbox will appear with a tick mark like. So, now, if we see the output of this program this is basically a Frame base not an applet base. So, we can see exactly here. So, the two checkbox C plus plus and java where the second checkbox is tick; that means, it is true checkbox is true.

Now, if we click the mouse there and there you can see the tick button will be check changed there so, but here in this case we ticked it, but there is no event generated because we have not used any event handling mechanism here in this particular example. Whenever we add some event when handling concept then it will be basically do if we check this one.

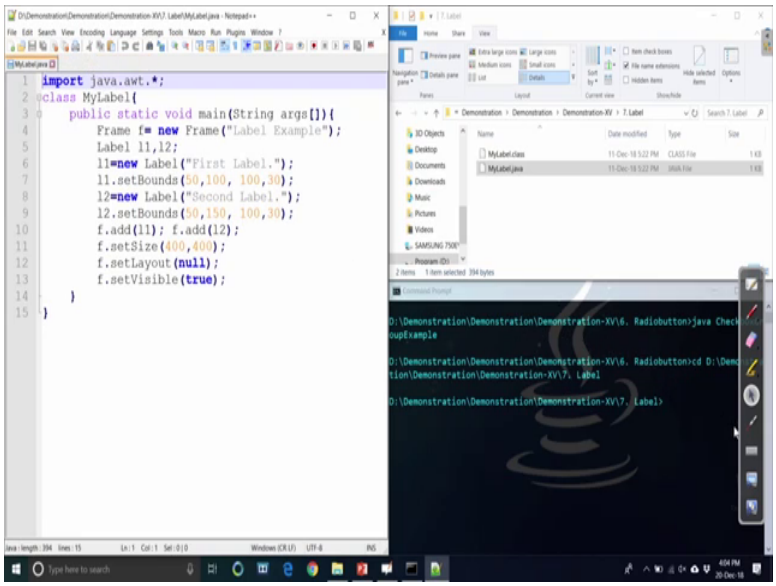
(Refer Slide Time: 09:40)




Our next example CheckboxGroup; it is similar to the Checkbox or itself, but is a group of items can be padded together here. Now here you can see we create a CheckboxGroup with 3 items 2 items one is fine. So, create the CheckBox1and CheckBox2 add these two Checkboxes into the CheckboxGroup. So, this is basically the procedure and we have done it and we add finally, all the CheckboxGroup into the frame.

So, here this is basically we adding the CheckboxGroup into the frame. So, it over and then if we run it, then we can see the CheckboxGroup how it will look like. So, as we see here and so, it is just similar to the Checkbox where actually, but it is a group this one and little bit difference as we see here in case of simple Checkbox the check window appears is basically square, but is a color as a round shape circle type and it is a difference between the Checkbox and then CheckboxGroup.

(Refer Slide Time: 10:59)




Now, we will see exactly how the Label can be included in a count container, in a Frame suppose. Here we create a first Frame and then we can see in this example we create a Label the 2 Label are created l 1 and l 2 and the 2 Label are Labeled as First Label and then Second Label and then we just position the Label into certain location which is mentioned here including its sides and then Second Label also and finally, we add the 2 levels l 1 and l 2 into the Frame and Frame is configured. And as we see the it will basically gives an output containing 2 Labels on the Frame. As you see here the First Label and the Second Label as these are name of the level here 2 Labels are the space on the container.

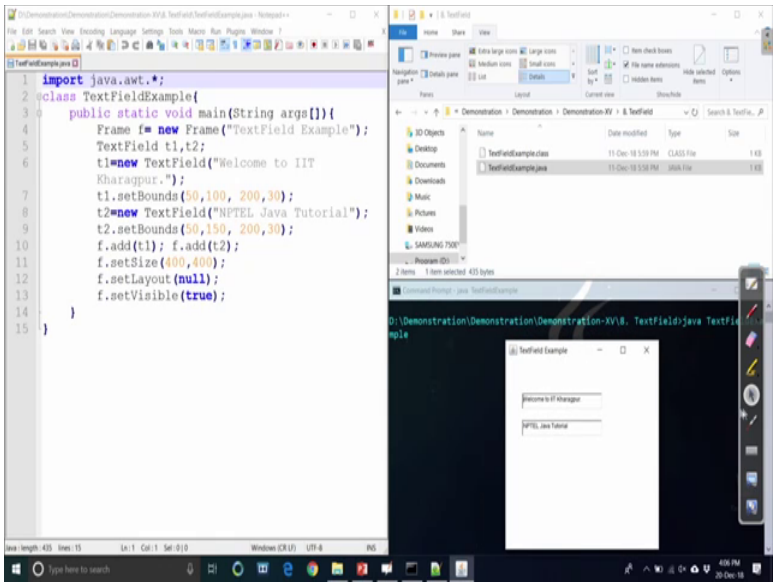
So, this way right. Now whatever the example that we have considered in a particular a program we can include all these things; that means, your container can contains many components; although here in every example we are using the different components to be included into the Frame. Our next example TextField. Now how a TextField can be created this example will tell us and then within the TextField something can be written also if you write it and then that component will be sensed by the event generator event handler and event handler will do could accordingly anyway.

So, event handling now skipped her e in this example simply we will see exactly how a TextField can be added into a Frame. So, for this we just write an application program here first create a Frame and then we create a text for there are 2 TextFields are created t

1 and t 2 are there. So, t 1 is basically text 4 with initial default ha what is called the levels are there will come to IIT. And then the setBounds is always there in which location this TextField will be floated and then what is the size of the TextField, as you can change the location automatically different positions it will float and then size also accordingly can be changed.

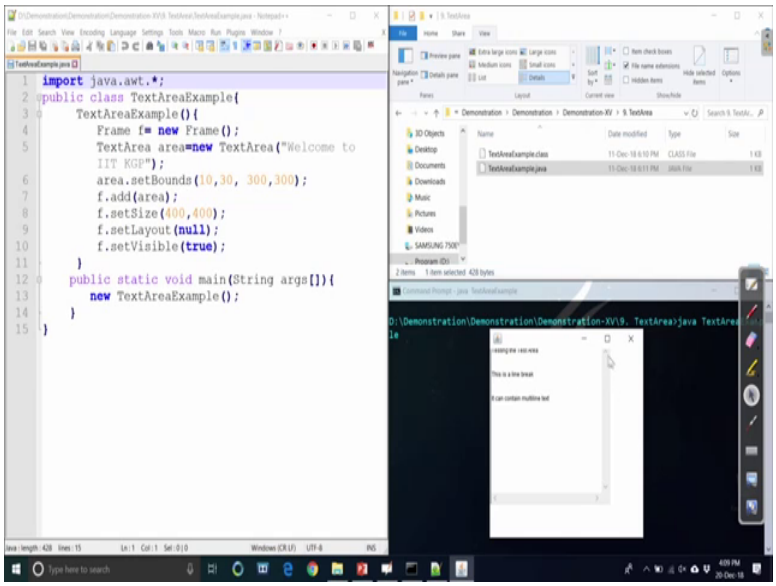
And then again second is a second is also used with different level NPTEL Java Tutorial like and then setBounds also decided the same as the first t 1 and then finally, we had t 1 and t 2 into the Frame and then Frame is to be configured and then that is all this is about the creating TextField in our program.

(Refer Slide Time: 13:20)




Now, we can run this program and as we see here yes. So, this is there and we can move the mouse pointer into any TextField and if we mouse it here and you can type anything. So, there is a type whatever you can type it is there and this type will go and then we can include it there. So, this way the TextField is basically allow a user to type something from the user using keyboard, positioning the mouse pointing first there and all this is the user there actually will be discussed or t in the different illustration that we will discuss.

(Refer Slide Time: 14:01)




Now, like TextField there is another one example it is called the TextArea; now text like TextField the TextArea is there, but TextField has the limited number of character that it can accommodate whereas, TextArea is basically is a sufficient area of the things are can be accommodate here. So, as you know if you want to include one para like say then you should consider the TextArea instead of TextField actually and so, it basically give a display a good volume of strings actually.

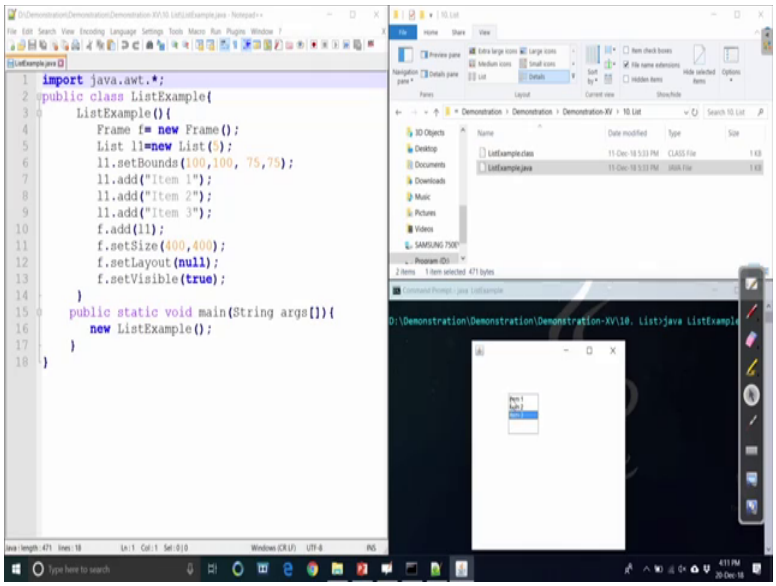
Now, here as the example as we see here we create a Frame first and within this Frame we create a TextArea a and then we just this TextArea include the default text like welcome to it KGP and then that size of the TextArea is decided as setBounds is there; that means, this is the amount of string characters that you can include into this area and then finally, we add this TextArea into the Frame and then Frame is configured and then Frame is visible as to so, that we can see the output.

As you see here these are default text we can write as much space it is possible total within the TextArea it is shown there and one more thing that you can note is that the TextArea whenever you de can define with the. And then TextField the difference between the TextField and TextArea is that TextField is a smaller in size of course, whereas, TextArea is larger in size, but there is one more difference that you probably noticed it where there is both side the two scrollbar area is coming.

So, if you go on typing much more text and everything. So, within the Scrollbar area you can I means scroll the entire area of the TextArea that is possible to have it there. So, Scrollbar automatically it is coming into your TextArea. So, usually TextArea is always accompanied with both scrollbar horizontal and as well as vertical we do not have to explicitly prepare it to be added, although scroll bar can be added in the independently other than this one also.

So, this is a very good example of TextArea and then it has many application in the context of many application since actually. Our next example is basically showing exactly List is ok.

(Refer Slide Time: 16:22)




So, List of items that can be included and those can be processed. So, here basically an windows program people many times use the List and then from the list one item can be selected it is like this is basically objective and here we can see there are 2 types actually List and choice item this example is basically List; we create l 1 we create one List is called the l l and we just create also a Frame because we want to include this List in a Frame and then; obviously, the List is size of the List is specified by the setBounds method and then l 1 is added into the List.

So, in l 1 List we add few items here as we can see if we go on adding it will just add into the items like in this example as we see 3 items has been added namely item 1 item 2 and item 3 although we have discussed that List of size 5; that means, two are

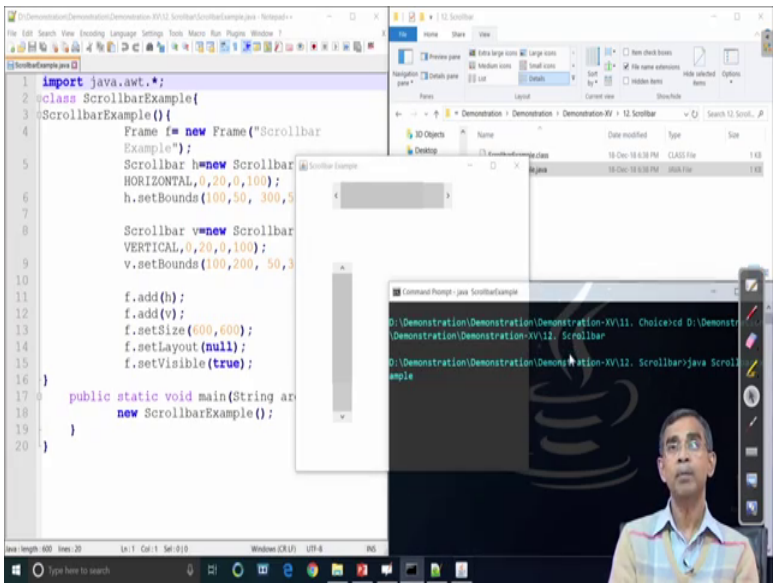
remained vacant actually. So, even if we do not do it also it is there; however, if we go for adding more than 5 actually. So, only it will consider up to maximum 5 and the rest of the things will be automatically discarded and then finally, we add this list into the Frame and Frame is size.

So, now we will see here this example includes a List of 5 items presently it contains only 2 items in it and it shows this one. As you see here if we select anyone item as we see it is basically showing the selection. So, automatically it is done within a very simple code you can see, how many things that we can do it. Now we will see alternate it to list items also there is another variations of this list is called the Choice. So, Choice is similar to the List, but the difference you can see if we run the example here.

First let us see the example run this one Choice, we can see exactly what is the different from the List and Choice here you can see. The Choice basically List shows all the items which basically it has whereas, choice will not show it, but if we there is a small scrolling see if we click it all the items which are there inside the choices will be included here and if we select anyone that will be highlighted into the choice item. So, if we like that ok.

So, this is the idea it is very similar to, but it is more elegant in case of choice than the list and after all you have faced many application where you can see this kind of examples in your example.

(Refer Slide Time: 19:07)



Now the our next example is Scrollbar. We have already shown that using TextArea how the Scrollbar can be automatically coming into there, but the Scrollbar can be used in a different way let us see exactly how the Scrollbar can be created and it can be included in our Frame.

So, first we create a Frame and then we create a Scrollbar for this the class Scrollbar is to be used. So, Scrollbar h is the name of the Scrollbar in this case and new Scrollbar and here we have Scrollbar object for this we created some values are there HORIZONTAL; that means, this is HORIZONTAL Scrollbar and then is basically starting and then location actually in which location it should come here and then finally, it basically visibility how much portion of the things Scrollbar will be visible. And finally, it is a range some from which value to which value it is there although values is basically not excursively to be there, but whenever Scrollbar is negotiated then the value automatically then be handled by the event handler actually.

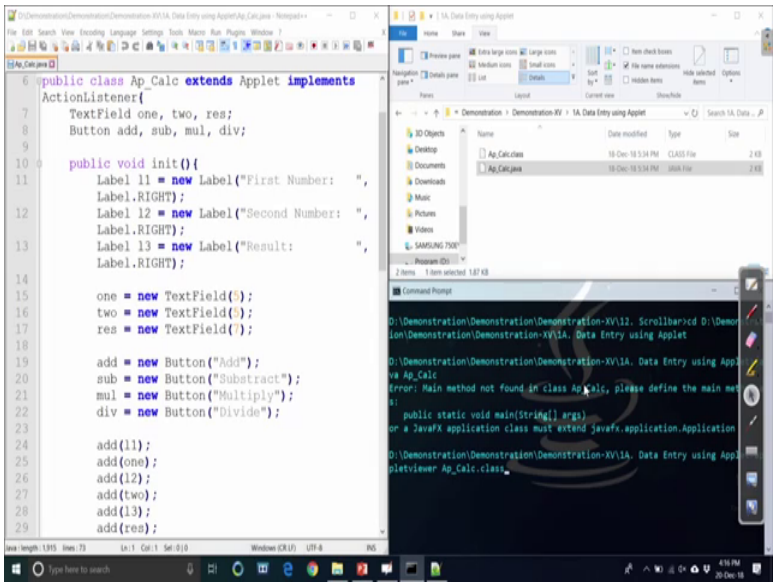
As we have not discussed the event handler. So, this part will be skipped here anyway when later on and then finally, the size of the Scrollbar can be decided in the setBounds and it can be added and then it can be added into the Frame. Now like HORIZONTAL Scrollbar another VERTICAL Scrollbar also we have created and the same VERTICAL Scrollbar is added into the Frame and the 2 Scrollbar. So, this example includes 2 Scrollbars one is HORIZONTAL and VERTICAL now let us see the output of this program this way you can understand exactly how this thing it is working.

So, here you can see the 2 Scrollbar the first one is HORIZONTAL and second one is VERTICAL. Now so, if we design one graphical user interface then accordingly you can position these two Scrollbar where you want to do it and then finally, you can use it and if can move it the using your mouse drag then you can move it and if you can release the mouse like at a particular position it basically says that in which location of the mouse drag is basically stopped if there location is basically what is the range that you have mentioned according to their range.

And then again size of the Scrollbar you can see the Scrollbar window actually those are basically dark one this is a size as a size is basically you can decide also very smaller size larger size depending on the your program. So, you can customize it like anything.

So, this is the ScrollbarExample and so, far all the examples that I we have discussed there are no applet is involved this is basically the Frame. Now we just discuss about one example where all these things will be utilized I am discussing about a simple example where a graphical user interface can be created by which user can enter the two what is called the name and then idea of the students and then it can be used and then finally, it can be stored in a file or like this one. So, this example we are creating using an applet.

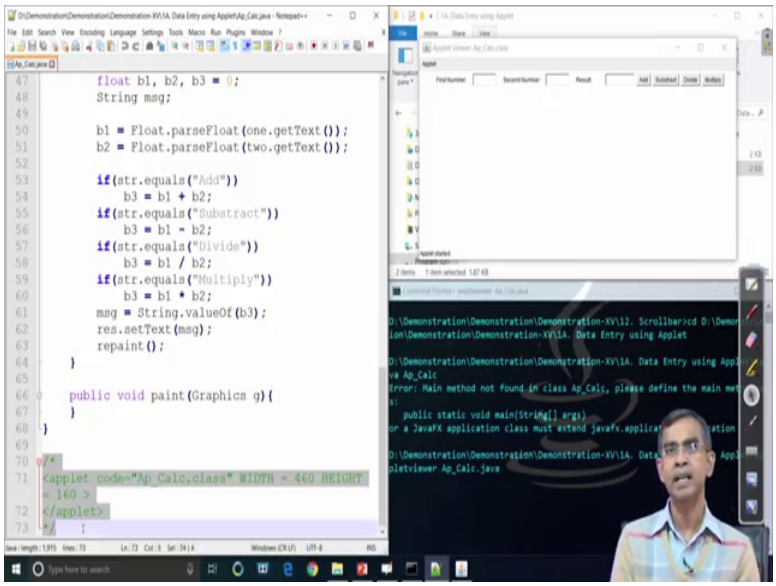
(Refer Slide Time: 22:21)




So, here if we want to create a GUI using an applet. So, basically you have to create an Applet. So, this is an Applet class extends Applet and here the event handling is ActionListener we can skip it right. Now here you can see in this example we create one TextField TextField we create a 3 TextField rather one two and result we like this and also we create 3, but 4 Buttons namely add sub mul and div.

Now let us run this Applet then we can understand exactly what you are going to do it. Now I am giving you the view of this Applet first and then I will explain the program so, that you can understand ok. So, here now you keep it little bit bigger in size please.

(Refer Slide Time: 23:24)




So, now, let us keep this not so, much big size fine. Now so, this is basically the Applet as we see. So, here we can see the acute TextFields which basically shown here these 2 TextFields also to understand that which is the TextField. So, 2 Labels also we have mentioned here 3 Labels actually 3 TextFields 3 Labels.

The first Label is basically First Number and the Second Label is Second Number and third Label is result and as we see for the First Label First Number there is a TextField and then for the Second Number is called another TextField and another is the another the result is another TextField and so, these are the 2 TextFields 3 TextFields are created and then labels also there now in the next example as we see here.

So, Label l 1 is created that is a first and then it basically Label dot right means right alignment actually so, the second Label and then third Label. So, 3 Labels are created and then also we have defined TextFields like say one two results and then Button also add sub mul and div and then for this one two three these are the TextField that we have instantiated here and this is basically size of the TextField area as we see this is basically the size.

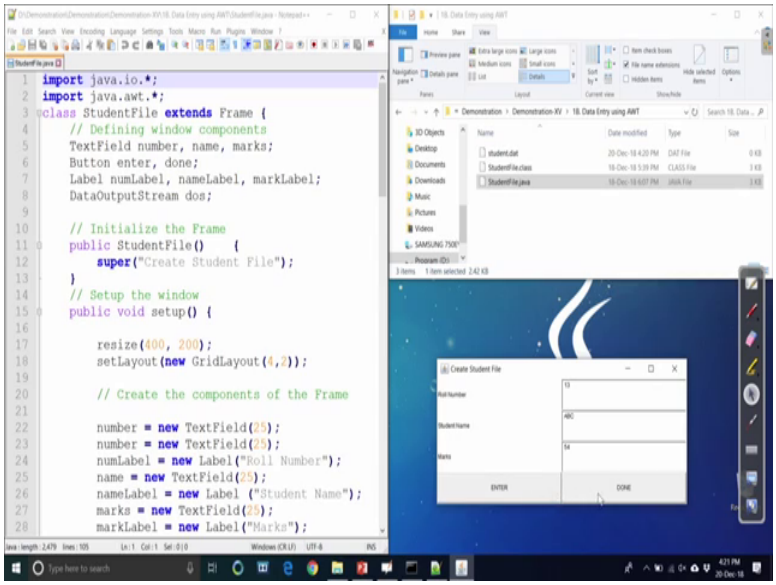
And so, this is basically the and then finally, we add 4 Buttons the Label as adds then sub mul and div and we add these Buttons. So, we define 4 Buttons here and finally, we see we add all the elements. So, add Labels 3 Labels like and also we add TextField. So, this is the add now here we are adding it and but adding you see in a flowing some order

because we do not have mentioned expensively where they will be added here by default they will be flow follow certain default what is called the layout management.

So, in this case actually there the different level management. So, this completes the look of the applet that will be there and rest of the parts is basically relate to the action handling. Now again go to the applet again that we have already created there if we type something go there in the first one say 25 and then next is say f15 and then the result if we see. So, the result add may be add. So, the result will be automatic

Now, this basically here the event is generated and the event some result is completed and then it will displayed. So, these are the next part of the program is related to this one. Now, here you can see how an applet can be planned or can be designed so, that it can include the components it is there. Our next example is the same thing, but not using applet, but using frame.

(Refer Slide Time: 26:29)




So, here again the same, but using Frame little bit bigger in size right so, that you can understand it. So, this you can see here the example is similar to the previous example only and here we create the three TextFields now namely number, name and marks and then Button enter and done.

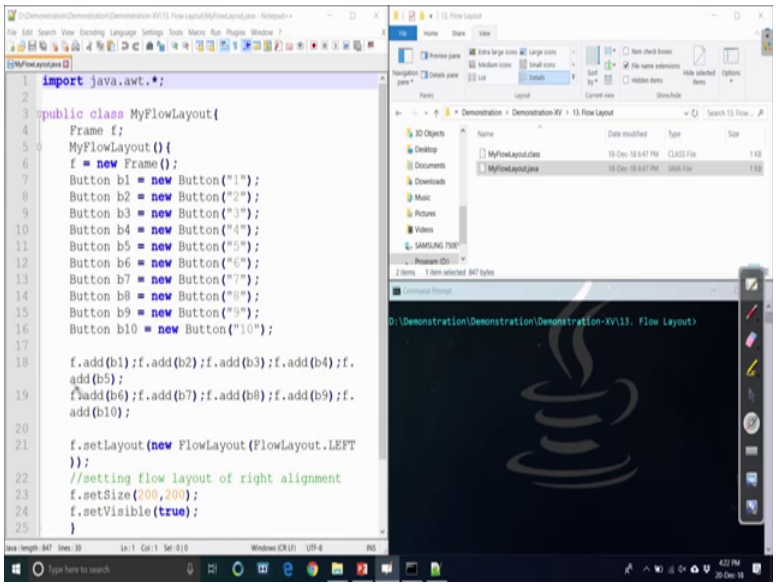
So, two Buttons are created and then 3 TextField it is created, but you see it is not an applet it is rather a component container window we can say and that window is created

in this program. As you see it is basically we create a Frame actually. So, this is a Frame based right and so, we create a Frame in this Frame we create 3 TextFields then 2 Buttons and they are added into the Frame and then finally, Frame is resized and rest of the things is relate the event handling.

Now, here if we type something here so, that type right. So, first name and then second second name and then done or enter means it will go there and then again clean it. So, this means one record is enter next you can do it like say a b c right and then done means it is complete and then the application will closed it ok.

So, fine. So, this is basically an example and obviously, it is the idea is that how a Frame can be designed and then how it can be handle and everything that is the most important idea to discuss in this demonstration.

(Refer Slide Time: 28:17)



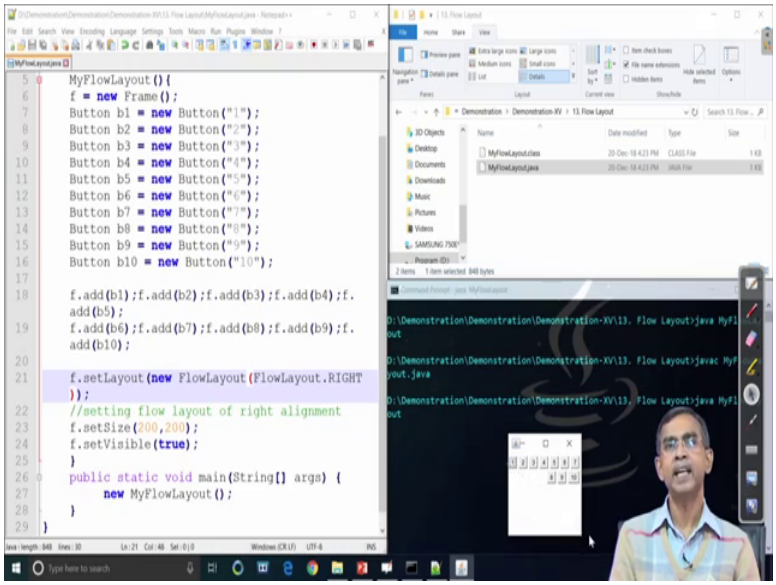

Now, we will discussed about. So, the layout management. As we have learned in the theory class there are 4 different type of layout the FlowLayout, GridLayout, Border Layout and card layout and. So, there are 4 examples that we can discuss here we can give a demonstration for the 4 layout management scheme actually, let us have the demonstration on the FlowLayout management.

As we know FlowLayout management by default will flow will arrange all the components that you can add into a Frame like here and they can be added from left to

right in that sequence only the sequence you have mentioned here in your program. So, here we have added say b 1 to b 10 10 different Buttons and then that Buttons are labeled as 1 2 3 right like and then we have added one Frame and those what Buttons are added into the Frame into the Frame in that sense actually it will go into the layout manager layout manger will place them from left to right; that means, Button one first then Button 2 and so on till the size of the Frame occupy then actually.

Then the next row it will come next row then alignment may be from left to right in this example we can by default is the right alignment and then something also can be done in the LEFT also here actually if you see setLayout new FlowLayout this is the way that layout manager can be involved.

(Refer Slide Time: 29:33)

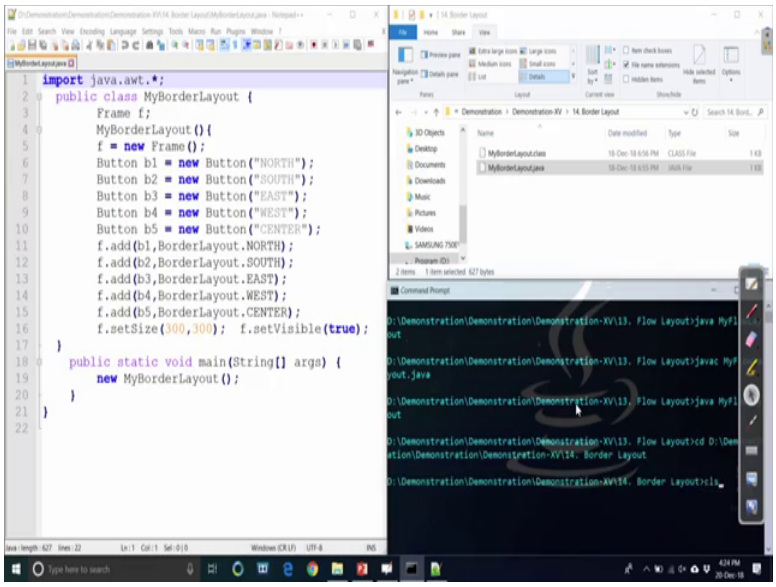



In earlier example we have not mentioned this one. So, default is the flow outlet manager, but here we are explicitly mentioning that we want to use the layout manager as a FlowLayout and this is basically left aligned. Now let us run this program and we will change the alignment the LEFT or RIGHT something like some sometime there is center arrangement is also possible here ok.

As we see here. So, let now if we change this right like I can you will see exactly how different output will occur for you. Now here if you see alignment is basically in the same order, but alignment is basically from RIGHT. If this how it will change as this is very small size whatever the portion it is possible to include it can be shown here, but if

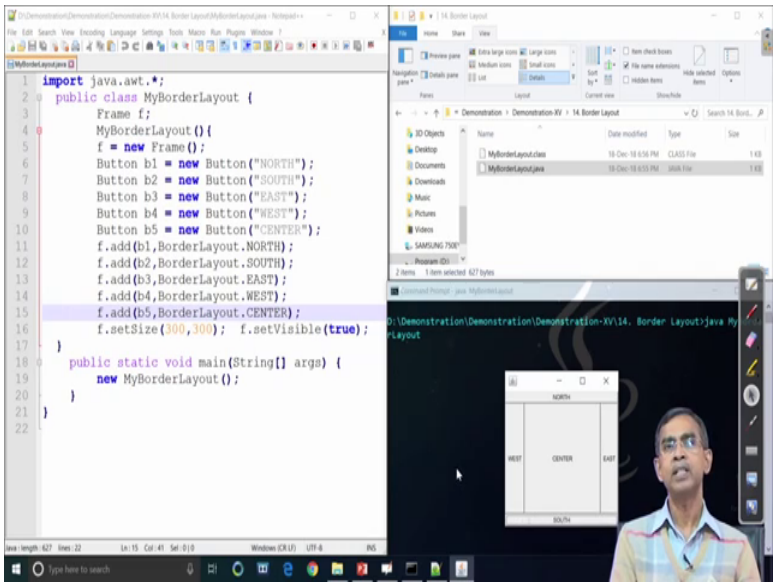
the change size is changed then automatically they can adjust automatically into the total area. So, in that sense it is basically little bit fluidic that mean it can maintain it, but is left to right or right to right that is the only thing it is you can see it. So, this is about the FlowLayout manager alternative to FlowLayout is the Border layout.

(Refer Slide Time: 30:50)




Here actually the total area canvas we can say can be divided into 5 regions center LEFT RIGHT and then NORTH and SOUTH like.

(Refer Slide Time: 30:57)

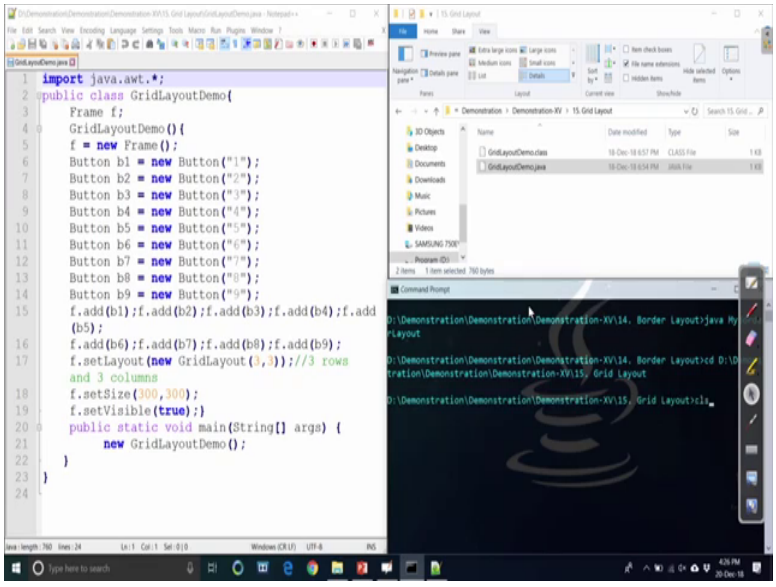


So, this is a convention of the layout specification according to the BorderLayout again we create here type what is called the components 5 Buttons actually here they are label as NORTH SOUTH EAST WEST CENTER and like this one and then we just adding them into the Frame and following the layout manager is called the border layout. So, it is basically we have to mention that in which location which particular component will go.

So, BorderLayout dot NORTH basically we are adding Button b1 into that region it is like this. Now if we run it we can mention it here as we see here. So, we have 4 components mainly the 5 components rather 5 Buttons are had been there, and you see the size of the Button in the earlier example where we have to mention the size of the Button explicitly by using setBounds, but we have not mentioned here.

By default whatever the area that is will be there this is basically default size is there. However, size also can be controlled either using setBounds or using the h gap and b gap of the this one. So, here for example, add b 1 BorderLayout NORTH then h gap in pixel and b gap in pixel also can be considered. So, that this is a gap between two neighboring either top or bottom or sides components the just the gap can be automatically mentioned and accordingly the size will be fixed there. So, this is about the BorderLayout and then next is basically the GridLayout.

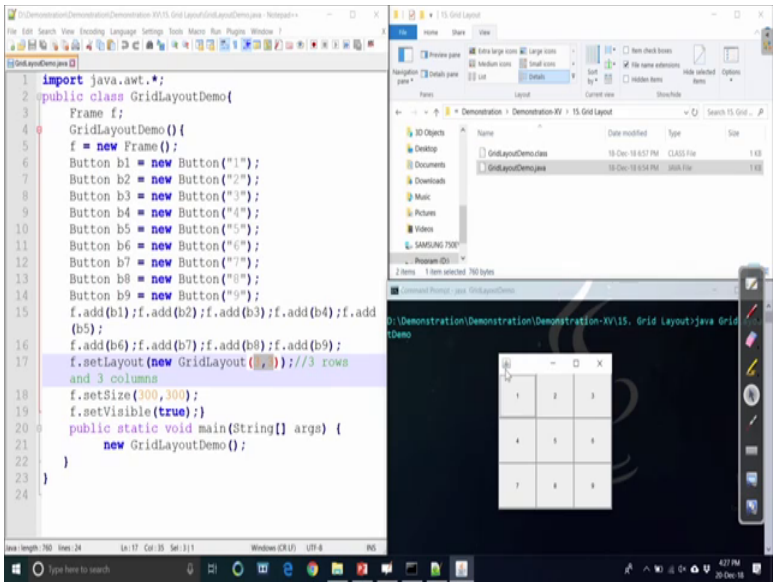
(Refer Slide Time: 32:31)



As GridLayout basically Border Layout has only limitation that only 5 components can be placed in the 5 distinct region where the GridLayout you can mention as many component as you can place there, but in the form of a grid.

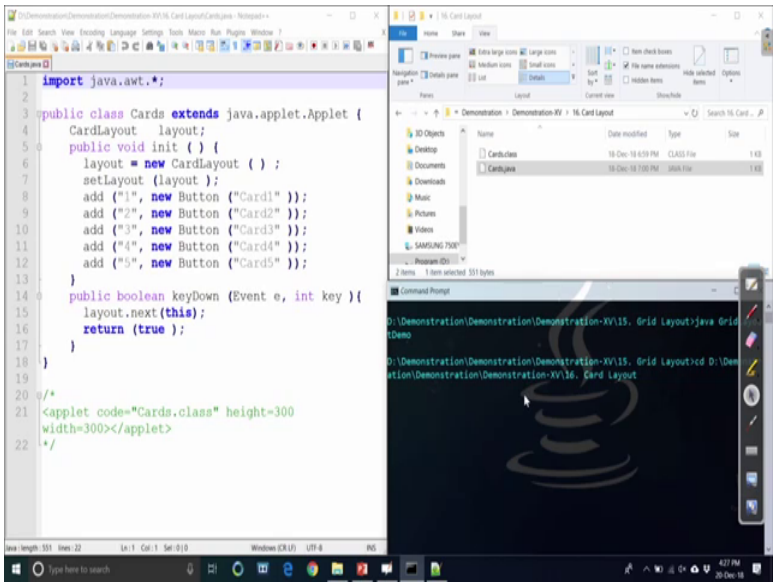
So, usually grid is basically having the number of rows and columns specification m cross n. And now in this example as we see we create a GridLayout of size 3 cross 3 so; that means, total line 9 is a example of 9 grids there we can mention what. So, 1 to 9 Buttons are there and they can be followed according the GridLayout pattern and then they will be automatically from order the order they have mentioned in this or the order we which we have added into the Frame actually, in the same order it will place it ok.

(Refer Slide Time: 33:33)



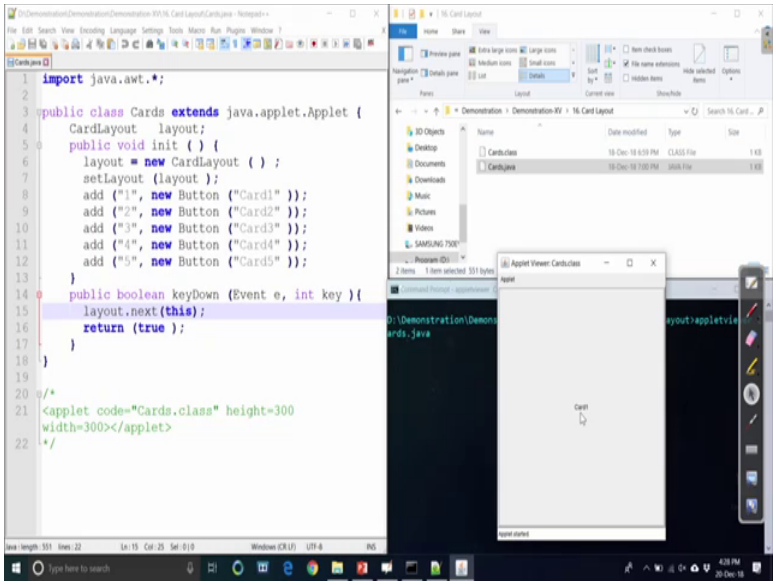

As we see here this is the display little bit bigger in size. So, I am just resizing by using the mouse zooming here right ok. So, we can see it here that the different Buttons. See in this case 9 Buttons are arranged into the GridLayout pattern following the GridLayout manager. Our next example is CardLayout.

(Refer Slide Time: 34:06)




So, this is the last layout manager in the example. So, CardLayout means it is basically 3 I mean you can add here in this example as we see 5 cards have been added, and all these cards can be clicked so, that the next card can be popped in this example as we see the layout dot next has been; that means, if we click the mouse then the next card will be popped and ok. So, this is right here now the ok.

(Refer Slide Time: 34:35)




Let us see example about it we have actually each Card in this case includes one full Button actually. So, little bit bigger in size fine. So, this is the one Card now if we click

on any portion here then the next Card will be displayed here as you see the next Card then again click it next card. So, the different cards are there and again go on; so, then fine right. So, Cards there is a 5 after that 5th Card is displayed if we again click on. So, Card1 will be there so.

If it is a next means, in the order they have added according to that order card will be there instead of next if we say previous, you can see the different result you can go obtain. So, instead of next you write previous. So, layout dot previous this ok. So, in the layout manger we can define next previous exactly ok. So, it is now created and if it is previous means, Card1 after this it will go to the Card5 and then Card5 then Card4 and so on. So, these basically gives an idea about how the different Card can be played actually.

Now, it is basically Card means a on clicking the different Frame will be popped up and then this will give an output like this. So, we have discussed although the different components individually, but actually in an application all the many components can be included in one application in Applet like this one then the whole thing say for example, in a calculator. So, there may be TextField, some label, some other Buttons and other checkbox and everything can be included and they can be added there and this way you can do it.

And finally, the manager so, layout management it basically helps a program automatically fixing in which way all the component can be managed. We have discussed about the FlowLayout manager, Border Layout manager, GridLayout manager and then Card Layout manager all these managers I mean classes they there they are defining AWT package as we have already disussed in theory and all the components are also discuss they are readily defined in the Java dot AWT dot component class. So, they are basically all Button Label they are the sub class of the component class and we can add it.

We can add them either into an Applet or you can add them into a frame. So, you have already discussed about how the particular component can be added, how a number of components can be added either into a Frame or into an Applet, how the components can be added into either Frame and Applet using say layout manager border card flow or GridLayout manager like.

So, that is all this is basically illustration about the AWT and our next part that we have to discuss about event handling because we are referring the event handling concept many times now we will discuss event handling that will be the our next module actually.

Thank you very much.