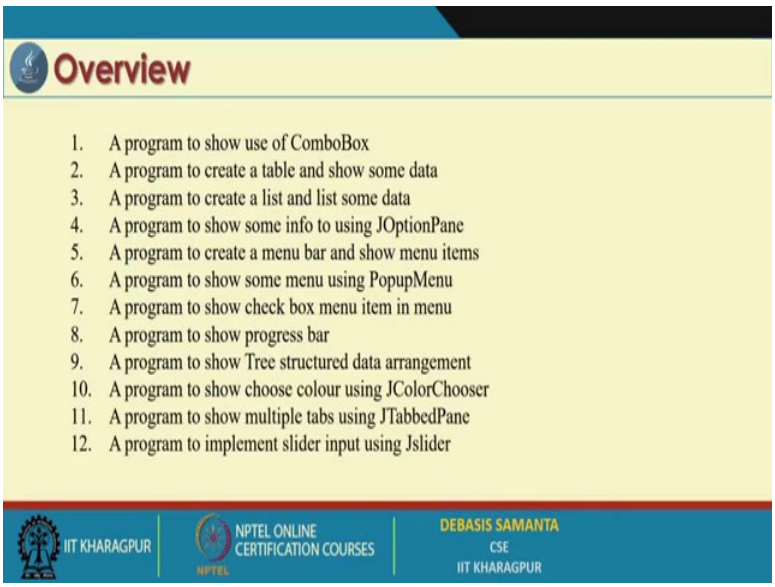
Programming in Java
Prof. Debasis Samanta
Department of Computer Science and Engineering
Indian Institute of Technology, Kharagpur

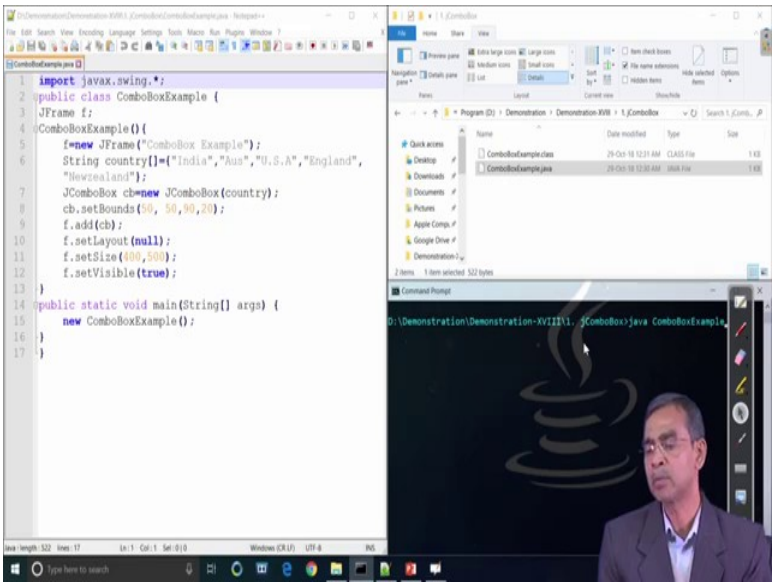
Lecture - 47
Demonstration – XVIII

(Refer Slide Time: 00:27)



So, this is a continuation of our demonstration related to the Java swing package in Java. These are second part of the demonstration. In this demonstration, we are going to cover few more components like ComboBox, then JOptionPane, then JColourChooser, Jtree, JTabbedPane and so many other things are there.

(Refer Slide Time: 00:43)



So, let us have the demo first. And this is the ComboBox. ComboBox is very similar to the CheckBox group actually, that we have discuss their in awt. And here how a ComboBox can be defined can be created this examples will illustrate us.

So, here ok, so first we create a Frame, because a Frame should be there in order to contain a ComboBox. So, Frame is created. And this class is the program itself ComboBox example, inside this constructor is a constructor base implementation. Then we create a Frame, the Frame name object name of the Frame is f. And the Frame will appear with Frame title call the ComboBox example.

And then, we define an array of strings like India, Australia, USA, England and everything. So, basically our ComboBox will include these are the items in to it into in it. And then we create a ComboBox passing this List of strings as an item in it. So, this is the one way that we can do. Otherwise, separately also add item, we can use it, and then add in to the ComboBox also can be done there.

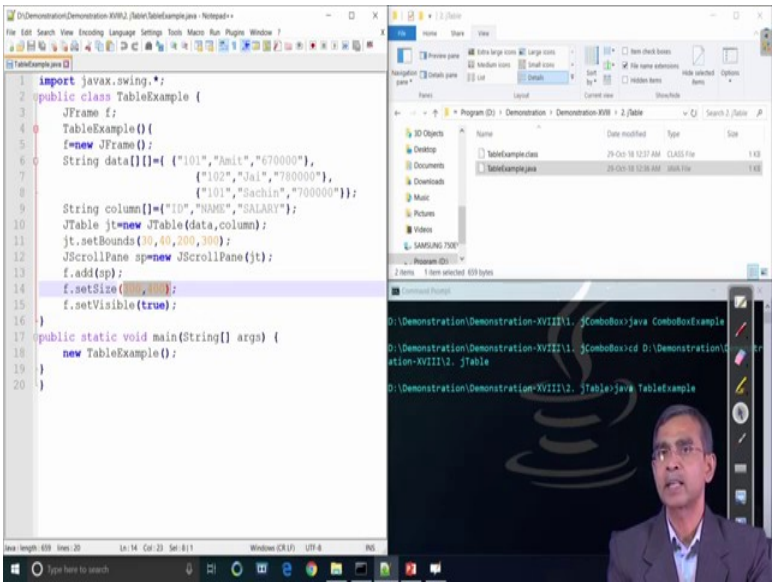
Anyways, so if then finally we can create the settings for this ComboBox regarding its location, where it should be floated, and then the size that mean width and height and everything. And finally, we add this ComboBox into the Frame. And then the layout size and the visibility of the Frame box has been set according to our requirement.

And finally, we call this object instance by means of its contractor only ComboBox

example contractor. So, this is the way that the ComboBox can be created here. Here we have executed this program, and we can see that this ComboBox is now already appear. The first item which is in the item List actually will be displayed here. And then if we click in the scroll bar there side right ok, then all the items those are there will be added here, and accordingly List here.

Now, here actually so the size of the ComboBox as per this example is concerned is the pipe. So, you do not have to discuss it. You can easily compare the ComboBox that we have discussed, while you are using awt. They are also one way. The same mechanism also, it can be followed here, but as it is a we have used a different example, so that it can give the different way, the different favors that a ComboBox can be designed, and can be used in your window program. So, this is the ComboBox.

(Refer Slide Time: 03:17)



And another thing is that a table, how we can create a table using you know the table. A table is basically collection of some rows, the rows contents the data, and there are set of columns, those columns are basically call the attributes. So, we are going to define on table which has three attributes. The three attributes are id the three attributes are id, name, and then salary. As you see there string column is a array, which includes the column heading for id, name, and salary right. So, these are the way that we can do it.

Now, here the table is a two dimensional array actually, so that array can be read from the either from the output stream or here which has we have store this table in the form

of a 2D arrays actually. For this illustration, however in different way also, it can be read from the other source, the way it can be there.

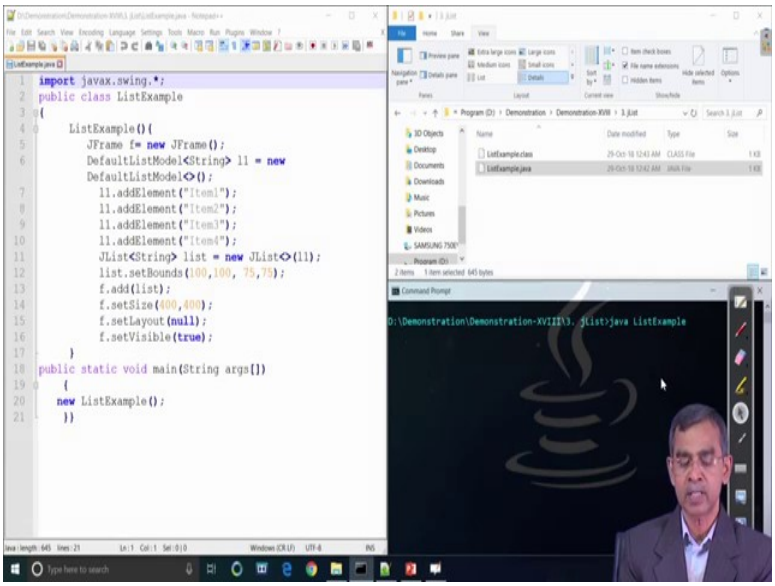
Anyway, so this table contents three rows as we see here. The first row include 101 has the id, the name is Amit, and salary is this one. Similarly, 102 and then 101, and all these things are there. So, this basically completes structure of the table, including the data into the table. And then finally, we have to create the jt. And then jt is basically the object that we have created of type JTable.

And then the argument that we have declared data, data is basically which includes the total records that should be there all the rows basically, and then column is basically column header actually. And then finally, set bounds that mean horse will be the size of the table that it should be, and then JScrollPane actually we have to include a scrollpane there, so that it basically scrolling is possible for the table. And then, we add all these scroll pane as well as the table into this Frame, and it completes the creation of Frame.

Now, let see how the output of this Frame will look like. As you see here yeah, so these basically size of the table, the table will look like, and it contains the columns as well as all rows. All rows correspond the all data that we have already populated into this example. So, there is again we can go adding few more table entries I mean rows and everything is just go on adding. Adding actually no problem, it is basically there is no limit of the size that you can restricted to the table that one.

And once this table is there, it can be processed by the different methods we have already discussed in the JTable. All these method can be used to access the different elements inside the table, how many rows are there, what is the number of elements, all these things are there. Even the searching also can be taken with the main different methods that is declare there. Now, this is the table example is very simple. So, for programming is concerned, absolutely that is not the big as you see here.

(Refer Slide Time: 06:17)



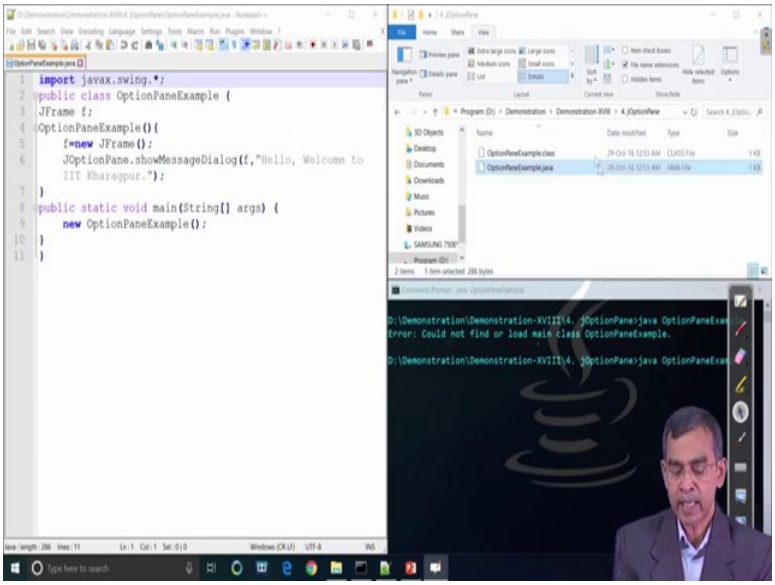
Now, our next example related the j List, it is very similar to the List that we have already discussed in awt. So, as you see the List is basically List of items that it can be. The same way we can put into the first array, and then array can be first as an argument, when you have to initialize say (Refer Time: 06:36) subject actually List object. So, here is an another way. So, you can add item independently one by one, and then it can be populated into the List only. So, this is the second version, second way of storing the different items into a List.

As we see here we first create a Frame, because Frame should include this List. And then we have created the List also here. Here List is created like default List model that we have created one idea about that ok, this is a string of List actually l1. Then l1 dot add element, it includes that we adding one item into this List. So, namely item 1, item 2, item 3, and item 4, we have added four different item into the List.

And then finally, we create the object of this List, the name of this is called List only. And then we pass this lll to this object as an argument that means, it will include all the items that we have added into it. And finally, these are the usual statements like set bounds, then set then f dot add List, adding the List into the Frame, and size setting, layout setting, and then visibility setting that is all. And then finally, we call create an object of this type of the program as an window, and then we create it one. So, this is the one example of List, it is there.

Now, we can select by means of placing a as we have highlighted. If we place the mouse, there on highlighting will be there. And here is that only one mouse can be there, mouse can be also over, and then many multiple List items also can be selected like this one. And that selection can be any where there, as it is known as a control button, and in addition to control the mouse. Base the focus of the previous selection will remain same, and the next also focus can be changed, both the focus can be merged, and then bigger focus can be made. All this kind of things that can be done here. Now, so this is the example of List.

(Refer Slide Time: 08:43)

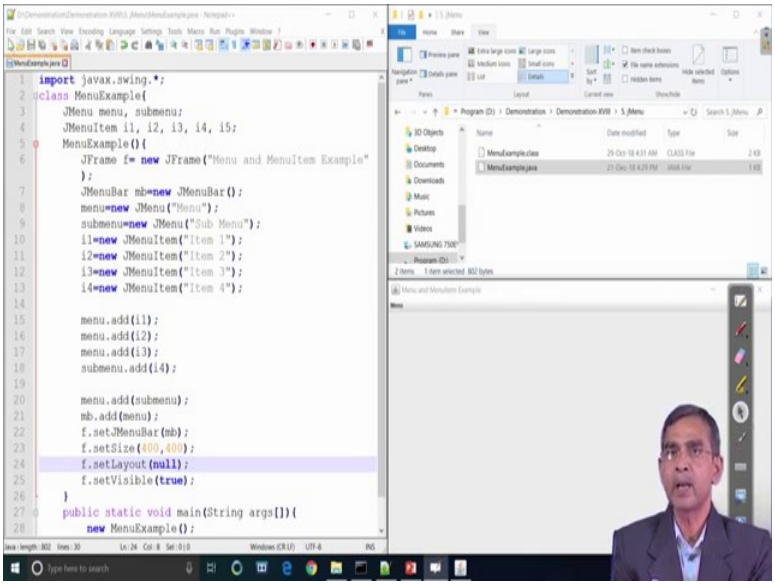


Now, next example that is we have plan to illustrate, the concept of OptionPane. And OptionPane is basically just look like a Frame, let us have a quick look of the OptionPane windows that it can be ok. The optionpane is basically the fixed size, it cannot be resized or cannot be changed here only a cross button can be used to close it or button can be press to select it like this one.

And as we see here OptionPane is created with a message like hello, welcome to IIT Kharagpur like. So, this is the idea of the OptionPane. As you know OptionPane usually sometimes appears in any our program, whenever you see today using mouse or everything regarding either progress of some development downloading of a file or if any error occur is basically error message or it is basically called the dialogue message to that.

Anyway, so this is an example how such an OptionPane can be created. For this things, we have to use the class call the OptionPane JOptionPane, and then for this thing an object can be created. And using the show message dialogue, and which dialogue you want to add into this that you can include it. And then finally, the another argument that in which Frame, and finally this object will be created, and it can be added into the final program. So, this is the program code very simple few lines of course are there, and that kind code is enough to display any OptionPane like.

(Refer Slide Time: 10:19)



Now, our next example is to create MenuBar. And so menu items into the MenuBar. As you know menu is basically create a List of items in to it, and then MenuBar is also basically showing the bar, where the menus are will be (Refer Time: 10:31). And if a particular menu is selected, then this menu can be chosen here. So, a typical look of a MenuBar and menu items as we see in this applet, only one menu here.

So, now we let us see the code, as you see the code here. So, JMenu, and submenus are two menus are there. If we select the menu, then menu will be selected, and then submenu will be there. And JMenuItem, there are items i 1 to i 5 the items have been identified there. And then menu example in this example as we see we create a MenuBar objects and mb and then menu new JMenu, menu is created.

And then submenu also another object is created like JMenu is again the JMenu object itself, but having the different level submenu. And then we add the item one i 1, i 2, i 3

into this menu item is basically calling the constructed there. So, we create and then finally we add these menu items into this menu itself. So, this completes the creation of a menu, which includes submenu items 1, 2, 3, 4 like. And then that is also this basically creates the menu item, whose output already we have experienced, it will look like this.

And then if I ask you to add one more menu, the same procedure that you can repeat copy paste and then just simply change the values, and the menu item bars and everything. So, another menu if it is there, it will appear here. So, many menus can be added, and then they can be placed into the sidebar or MenuBar actually. So, this is basically the MenuBar. And then for this MenuBar includes many other menus also. As you seen in any other windows application like word or power point a many menus like this, and as we see here also. The many menus are can be included in the MenuBar, and then the menu item can be added in each menu. And then each menu item from a particular menu can be selected by clicking all these things. So, this is also another example of MenuBar like. So, this is the one example, so that is all.

And next our illustration let us have the popup menu. As a name popup menu, it is also similar to the menu item only. But, is a popup menu, it shows that ok, let us have the quick output of this that how it looks like a popup menu first ok. As we see the popup menu, it is basically fine.

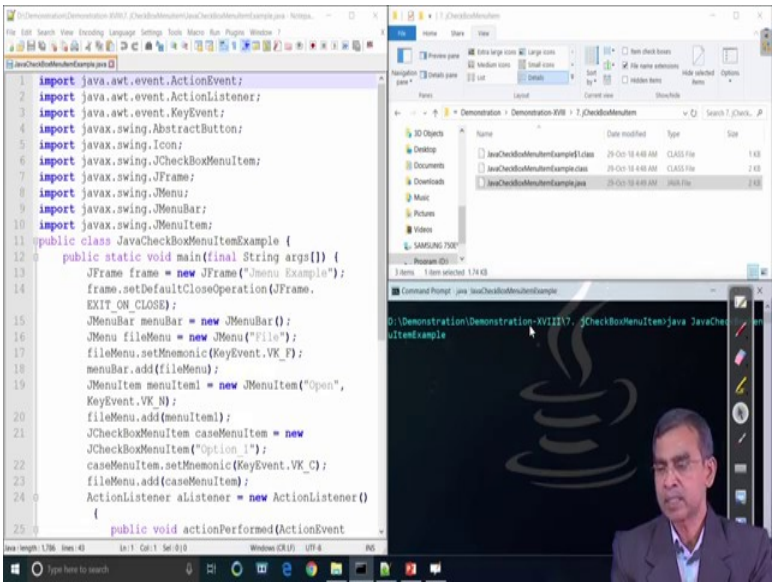
And whenever we click it, so pop up menu will appear. And popup menu will appear with some items in it, so this is the idea of the popup menu. So, it will not be there until you just click it, so that is why it is called the popup, whenever some events clicking is occur. So, pop up menu will appear there. So, this is something different than in the example, whatever you have considered about awt is a specific to this thing only.

And here we can see, we create a Frame first. So, Frame will include this popup menu, and the name is final J popup menu, final this is the popup menu actually yes, we create this one. And then this is the name of this pop up menu is edit actually. For this edit menu, there are item that we have added here cut, copy, and paste like.

And then finally, we just add all these item JMenu item into this pop up menu like cut, copy, paste as an object that we have created adding them into the pop up menu item. And finally, we just register the mouse Listener, because if we click mouse, then only this one. So, even needs to be registered here in this case, and this register basically

defined popup menu dot show method it is there, and then e basically take the event that it may occur, so then. And finally, this popup menu item is added into the Frame. And this completes adding one popup menu into your application program. So, this is the one example that is basically.

(Refer Slide Time: 14:54)



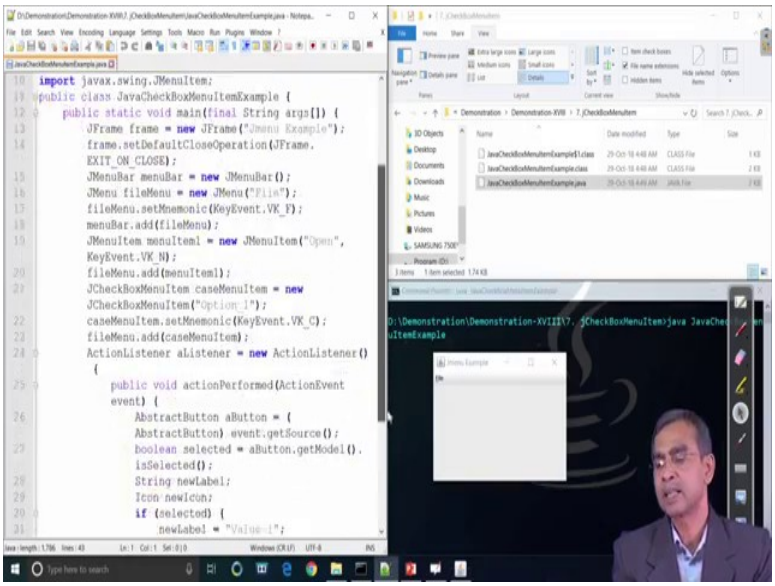
Our next example that we are going to discuss about similar to popup menu, but it is called the check box menu item in the menu. So, it is a menu item, but with checkbox, so that is the difference and ok. First see exactly how it look likes actually with a single on checkbox menu item in this example, as we see here yeah.

So, here if you see at a menu item is displayed in the MenuBar, and if this menu item includes and as a popup look like if you click it, then that menu item will appear. And then it has two values like open, and then value two ok. So, it is the two items it is there, and option one, what is this?

Student: Open and value two.

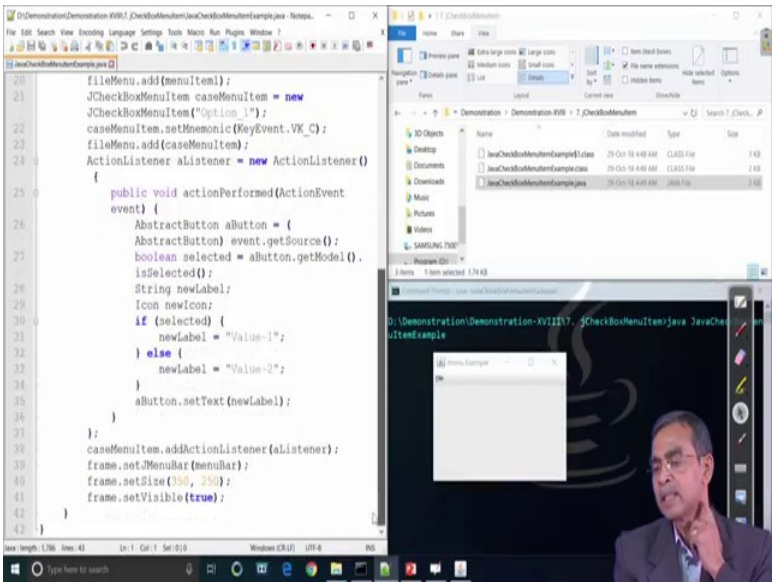
Open and value two. And value two is with the checkbox, where the open is ok. So, value two is the checkbox. So, this is the idea about that checkbox can be added into menu item.

(Refer Slide Time: 15:49)



So, you can select, and add it like this.

(Refer Slide Time: 15:50)



Now, this quote can be little bit ok, we can go through and understand here. So, here that event needs to be handled, because it is relate to the event ah associated with this one. So, without event, it will not give that kind of flavor actually. And as we see the program includes again Frame, and then Frame includes some default closed operation method, those are defined therein Frame class itself. So, for this Frame class, we called it. And that is the thing that is necessary.

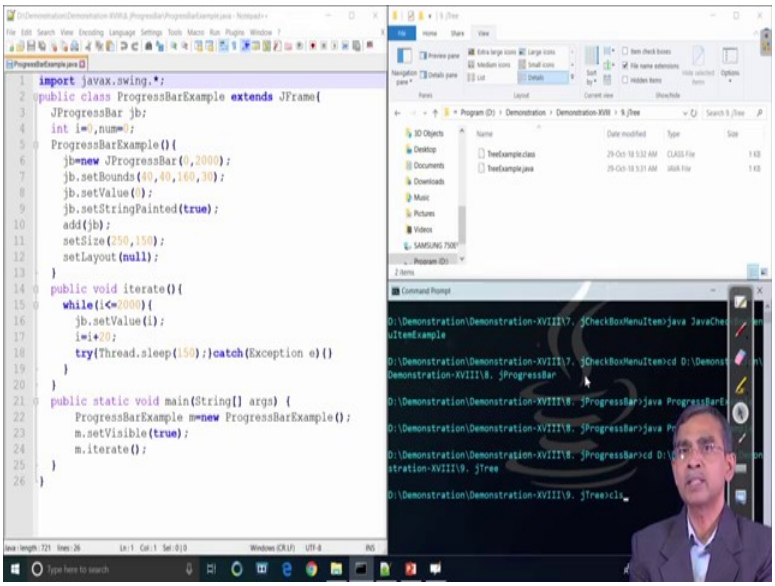
And this basically set as the exit on close means, if we just to I mean remove our mouse click, and there it will basically close automatically, so that is why you have to do it. Otherwise, once you open it, then it will remain, so that is why we have to set a exit on close. And then we create the JMenuBar that we have already discussed that how the MenuBar can be created in the MenuBar class. So, JMenuBar class, it is there.

And we create the JMenu objects, so there is a JMenu objects the name of this is file, and then the set Mnemonic key event VK that means, if the event is occur related to the function actually, it will takes place. So, this is the one definition, we have use it here. And then menu item, we have added like open, and then another menu item that we have appeared there is basically ok. So, yes so menu item that we have created here as the open one, menu item one which basically with one argument as the key event virtual key any number. And then we add this menu item, and then JCheckBox menu item, case menu item, new checkbox menu item like this, and fine, so these are the few many.

And now here is basically the action l in routine as we see here, the different action that can be handle here abstract button. We can create an abstract button type here a button is an abstract button, abstract button is created with an event as the get source. And then button get model, and if it is selected is true, then it will be displayed there.

And then there are few levels, we have used this string, and they are also icon image, we can created here. And then value 1 and value 2 depending on the selected true or false. It will be selected, and then it will add this button into the period, and then finally it will just complete this one now fine. So, now we can have the demo again fine. So, this is the one example that is related to the item the checkbox menu items in menu.

(Refer Slide Time: 18:55)



Now, let us have the next one illustration regarding the progress bar ok. So, as you know the progress bar, we will show the progress of happening of some event. So, automatically is a dynamically, we will change it. And we can have a look about it, as we see we have displayed the progress bar. And the progress bar with certain speed of progress, we have already defined in our program that will be explain shortly and it is there. So, it basically shows the (Refer Time: 19:35) it will stop it, wherever the compute is there, and then close it.

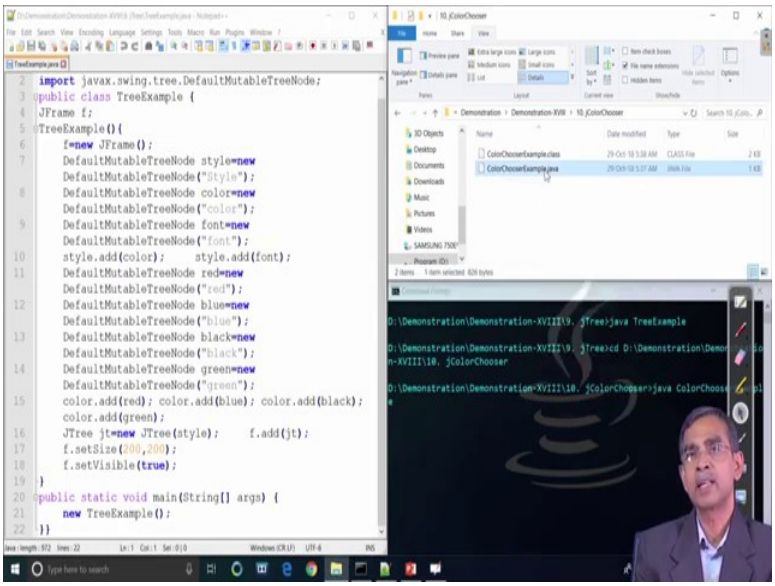
Now, let us come to the program which basically can be used here in this particular to give this kind of output demo. And we first create one object of class JProgressBar, here is the jb. And integer i equals to 0 to num equals to 0 is basically initialize the two numbers. And then we create an constructor of this example, so progress bar example.

Now, jb as a new JProgressBar has the range from 0 to 2000 is basically the range of values you can that presided, but it is actually the progress will be computed in the percentage calculation. So, no issue is there. Then size of the J bar JProgressBar is basically had decided with the location as well as the width and height. And then set value is basically 0, the program bar will start from the zero location actually, then j b set string painted. So, it is basically true, because it will highlights the progress right. And then finally, we add j b that mean the progress bar, we added into the Frame here is the constructor of the same. So, it is added in the Frame, set side set layout ok.

Now, here in order to have the progress dynamically changing, we have to define one more function here that is call iterate. And this function as we see just loop, and loop will continue till it completes from 0 to two 2000 there, j b set value is i. And then progress is basically, how much speed that here basically i plus 20, so 20 at (Refer Time: 21:16).

If we change it, then progress bar will right improve it faster like ok. So, whether slow movement or fast movement that it can be obtained like that way. And this basically the way that way progress bar can be designed. And then finally, it will include into the main, and the main an object can be created of this types and it basically will execute. So, this is the way that a progress bar can be design, and it can be used.

(Refer Slide Time: 22:01)



Now, the next one example is very important in the sense that a tree structure can be designed using Java swing. And use a tree structure mostly you are familiar with to the different, whenever you open a directory using command pond or some I mean using the command that is there in a computer. So, if you see the directory structure, so j t is basically showing a directory structure that is possible.

In this example, as we see that is a very simple structure, again it can be expanded also depending on if some other nodes are there in this tree structure. For example, here the style is the root in the directory structure here, the style has the two what is called the nodes namely colour and pond.

So, if we select colour, if some elements are there, then it will automatically expanded into that right, and then red, blue, black right. And if the red content some other, then if you click it can, but in this case it is not there, so it is like this. So, it is a tree look like, so it has the main. And from the main the sub main, and then from sub main is items like this one. So, there is no limit of the hierarchy from which we can go from top to bottom actually, so as we can see there.

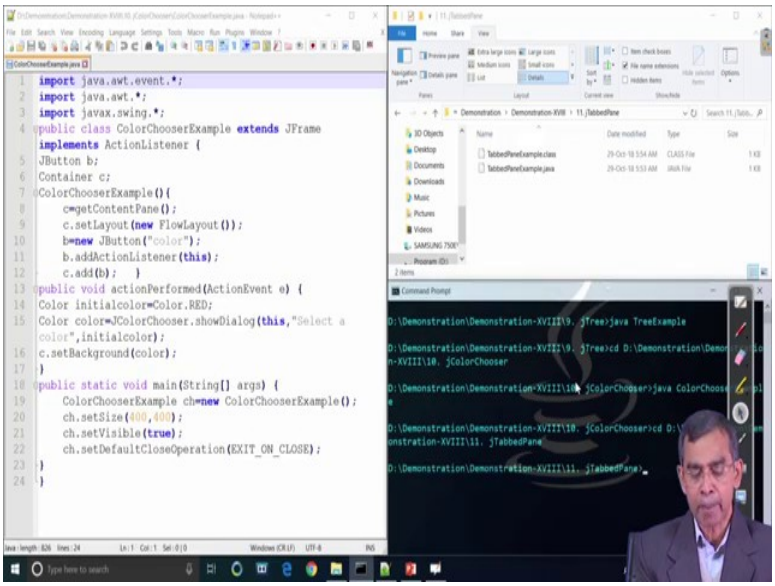
Now, we will see exactly how such a menu, and then submenu or directly or sub directly can be added into the component. So, this example shows, first we have to create a Frame, f is the Frame in this case. And then we just discuss the program as the tree example as a constructor, we will invoke this object as a constructor that is why.

And here the different items into the tree can be declare by default mu table, tree node, and then style. So, this is the first one style. So, we have label as style, so that is there. We create some other like colour pond also, because they are also table node t node actually. So, we have created here three t nodes; style, colour, and pond. And then we add colour and ponds to the style nodes, this is basically sub node of the style node, so that is why we add this one.

And the next we create few more nodes relate to the different colours like red, blue, black, and green, and finally we add this nodes into the node colour. So, this is basically sub node or children of the node colour actually, so that will appear as the children of the node colour. So, this completes the formation of different nodes.

And finally, we just create the J tree I mean tree, and passing the root node into it. So, here we see we create the JT has the tree, and root node is style here. So, passing the style. And then finally, we add this tree into the Frame, and whatever the other sizing of Frame, visibility, and resize everything is done. So, thus includes the basically how to create a Frame of our own, and as the output is shown, it basically will display the tree look like this. So, this is the example of tree, and many more items can be added, you can have the many other things are there.

(Refer Slide Time: 25:27)



Our next example is the colour chooser. As you know you can set the colour either using from the choice, as we have already experienced one such example in some other demonstration session. But, here the colour choice, the Java swing gives you to have the general tree of the different colours, and from there we can select a colour of our own choice. And then that colour can be set as a background or as a pond colour or whatever it is there, and this is an example can tell you.

Now, let us see the example here. So, this is the one Frame, and the Frame include a button colour. If we click this button, the colour choice will come automatically here. So, this colour choice is come, and then this colour choice has the different what is called the options MenuBar in the menu as I see RGB, HAS, and HSB all these things are there. So, these are the different way the colour can be chosen, and then it can be setting, and can be set automatically it is there.

So, everything is basically light weight like that means, they you do not have to bother about their direct implementation, it is there. You just can add it, then your in your program you can access all the facilities, it is there. So, as we see the different colour modes that mean RGB pattern, it can be shown. And accordingly, for them the different colour can be set also.

So, this is the way now here is the different way the colour can be chosen actually, it has give you the four different way the colour can be chosen. And also by clicking a colour

point there, we can add the set the colour as we see the changing the colour, we can change the colour point and everything. So, whenever you choose the colour, it will decide there which colour that we have chosen, and what are the previous colour that is the history. All the history colour also can be shown you in the recent pane actually. And then all the colours has been chosen, and then this things are there.

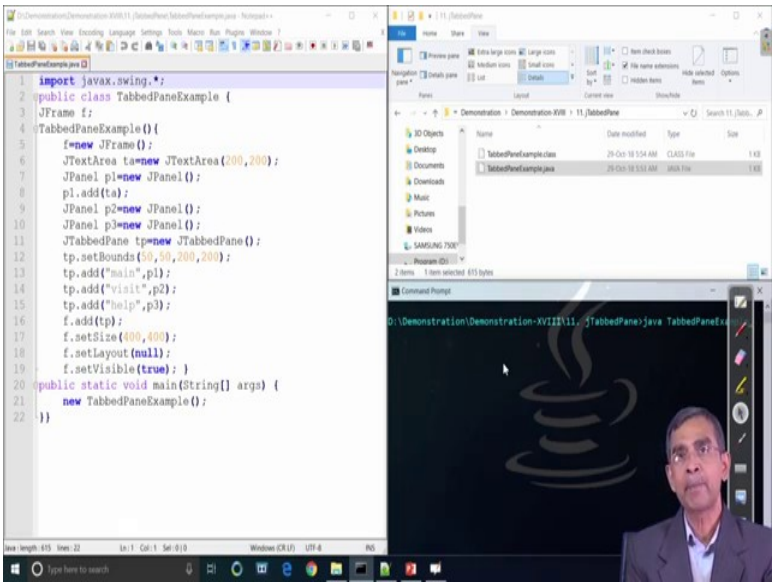
Now, so many things those are there in details, but can be done with a minimum number of coding possible. And here is the code that basically can deal with the colour chooser component in Java swing. As we see the code, it basically the name of the class that we are going to implement is colour chooser example extend J Frame, and as we have to input some events. So, action Listener to be implements. So, it is basically inherits the action Listener.

And then we create the button J button b that basically for the initially the colour that that needs to be created there, as we have started with the JButton colour actually, so it is there. So, we create the button, and the label as colour. And then add action Listener this one, c add button, so we add this button into this component. And then get content pane actually, so content pane is basically where this colour will content actually.

And then we defined the performance, so the event handling routine here action performed action event e. So, colour in initial colour is basically chosen as red, these are default colour. Otherwise, colour can be set as the J colour chooser, then show dialogue this select a colour as basically is a dialogue that will be there, and with initial colour actually. And then set background colour as a colour that means, whatever the colour that you will choose, it will be set the layout with that colour that is all.

And then finally in the main example, we create an object of this colour chooser. And then colour chooser will be placed into this component, and then it will be resize and Frame displace. So, this is the way that is there, it is not so big what is called the steps or basic complex procedures are there very simple, only you have to add the colour chooser example there. And it can be added into your system.

(Refer Slide Time: 29:11)

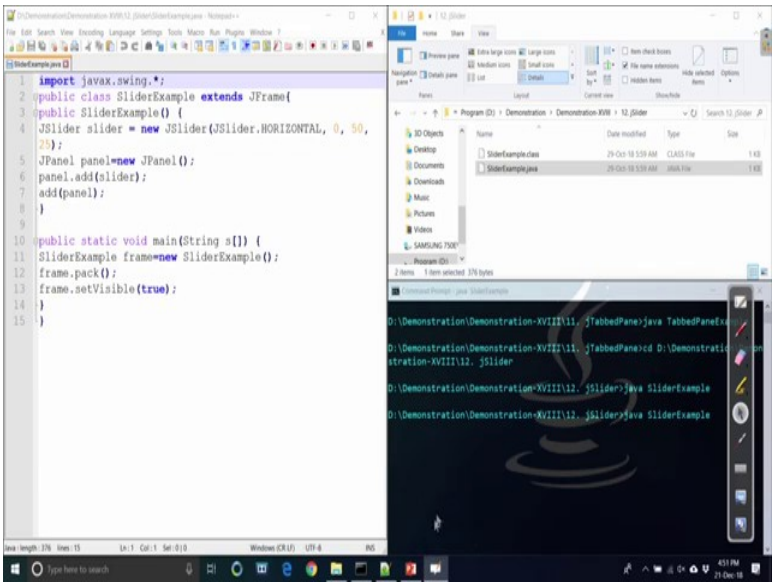


Now, our next example is basically TabbedPane. So, TabbedPane is basically let us have the view of a TabbedPane window. TabbedPane as you see is a tabbed, so there many tabs are there, as we see here in this example. A TabbedPane is appears with three menu main, visit, and help. So, likewise the different menus can be added into there, and this example explain the idea about it here.

So, we first create text area of size say 200, 200 here, so this means that this is the text area of size 200. And this text area will include as a final actually, so this basically we add this text area in into a panel. And there are two other panel p 2 and p 3 that we have added there. So, other than the original pane will p 1, p 2, and p 3. So, p 2, p 3 are there basically panel related to the different menus are there on this. So, p 1 is for main, p 2 is for visit, and p 3 is for help look like.

And how we can add this into the TabbedPane is basically, if as we have created the tabbed pane object tp, just we are going on adding all these menus there tp add, tp add main, tp add say visit, tp add help p 1, p 2, p 3 look like. And then finally, we have to add this tp that pane into the Frame, and it is completes the creating a tabbed pane the way it will see the output as we have seen. So, this is the way the tabbed pane can be added, and the tabbed pane also can include some other elements or it can include some event generation, so that if we click this TabbedPane, some things will appear like.

(Refer Slide Time: 31:06)



So, our next example is JSlider. We have already familiar is the JSlider, some glimpse of JSlider have been used in our colour chooser example. It is basically similar, but it is an independently one JSlider, how slider will look like as we see here is an output, let me bigger yeah. So, here we can see example of a slider.

And then there is a slide button is there, if we click, and then drag, then the slider will move across the vertical line some horizontal, orientation also possible for the slider, but usually it appears is vertical one. And so this is the slider as we see slider if we moved it, and and a tiny position, the slider is placed a value can be written there. So, there are the way that the slider value can also be read.

Now, here is an simple example that it can includes a slider. So, we give the name of the slider as JSlider as slider. And then is basically slider horizontal, so that is horizontal movement. If you write the JSlider or vertical that means, the vertical slider will be there. Now, the next three items are very important. So, first is shows that 0 and 50 is the location is basically, what is the 0 and 50 ok. So, 0 and 50 is basically range, and then 25 is basically the width right in the pixel right ok, so default is there default. Whenever it appears the default location in the range, that value will be appeared there. So, it is a 25, it is the default location actually, whenever it is I start the showing it is there yeah.

So, the default location is 25, and then range is basically 0 to 50. So, the minimum is 0, and the maximum is 50. And then we can add it into either Frame or panel. In this case,

we have added this one panel, so slider has been added into a panel. And then panel, finally add in to the Frame. And this is the way that slider can be there. So, we have almost cover all the different important JSwing components, those are there in the JSwing package, there are many more also. Those things you can write you can practice of your own and more practices required, so that you can learn it much more in a details way.

Thank you very much.