Programming In Java
Prof. Debasis Samanta
Department Of Computer Science and Engineering
Indian Institute Technology Kharagpur

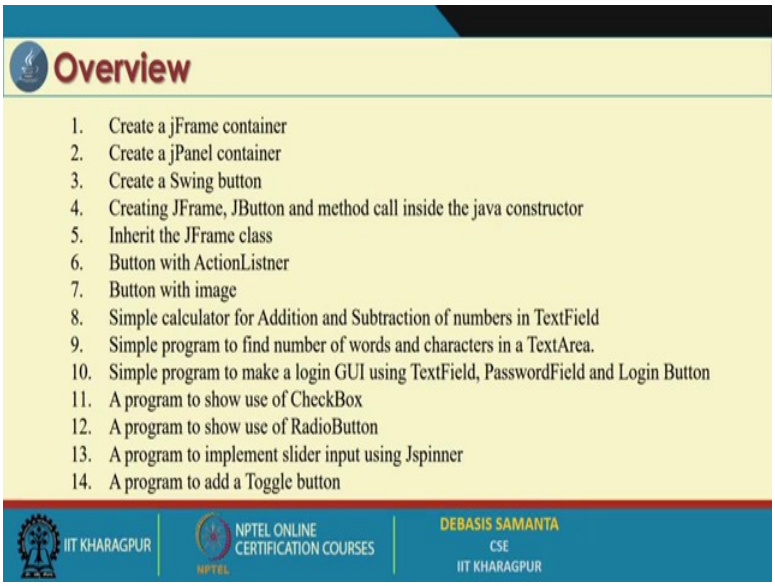
Lecture – 46
Demonstration – XVII

Now, let us have the demo on Java Swing facilities.

(Refer Slide Time: 00:21)

Now, java swing facility will be covered in 2 Demonstration modules this the first one.

(Refer Slide Time: 00:26)

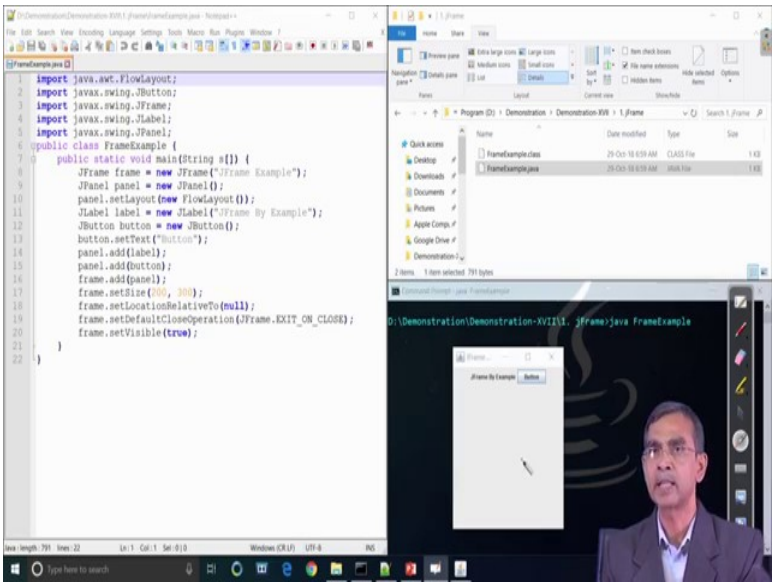


In this demonstration modules, we are going to cover how to create a j Frame, which means frame in java swings rather and then panel. And then Swing buttons actually different components are a swing and then here whenever you have to include this component.

There are a few things needs to be remembered, either using java constructor we can add some component or using a frame class or using the Action Listener method it is there and also we will see that which makes little bit different from a w button to the JButton that how a button can include an image, that means button with image actually. And then we will discussed about the simple calculator using addition and subtraction which we have discussed in our TextField, even handling on demonstration there, the same things will be but using java's swing only and simple program to find the number of words and characters using TextArea.

So, these are another advantage that we can have from the java swing this is how I swing is more preferable because it has a lot of built-in programs automatically it is there. So, automotive or account and then character count also can be facilitated from the GUI and then other GUI component like TextField PasswordField and then Login PasswordField or everything will be covered and then CheckBox RadioButton Jspinner and then Toggle button all these things that ever we will try to accommodate the demonstration of all these programs.

(Refer Slide Time: 01:59)



So, let us have for the first demonstration on the j frame, which means how a frame can be included in a container using the java swing and for these things as you know we have to import javax.swing.star, this is necessary if we want to include the swing component here and then other things are to be included like import java.awt.star and all these things are there those are related to this one.

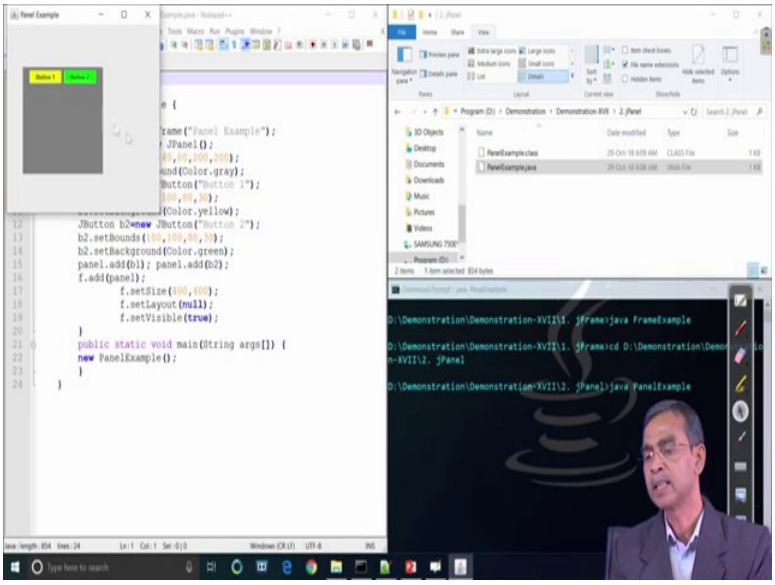
Anyway, so we have included those are the things are there they are javax.swing.star it will include all the packages that are there. And by means of that all classes those are defined there will be accessible to this program and this program is very simple as you see we create one frame the name of the frame object that we have created is simply frame and the level is JFrame example. And then also we declare one panel and then layout we have sit into the floor layout you have already learned about in the context of awt and then level has been initialized with the level button edge.

But a level has been initialized as JFrame by example that is all and then button one object is on a component is created here and all those things are added into this panel first and then panel has been added into the frame that is the policy it is basically that. So, panel includes the different Jswing components and the final panel is included in the frame objects and then so we have to just yeah.

So, if we see the output of this applet then you can find it the output that you can get it here as you see here a little bit will be regarding screen then ok, so JFrame by example

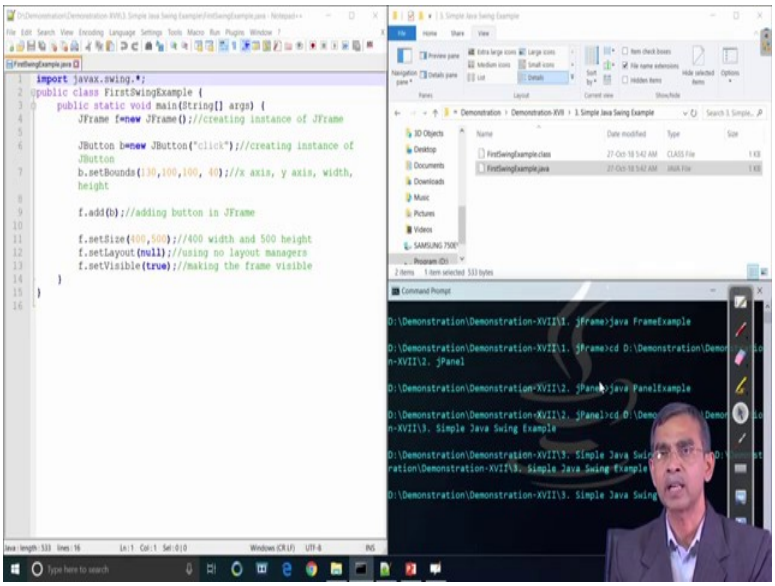
and then button it is there. So this is a simple example that a frame can include the different Jswing components how it is they are including here panel level and button everything has included in this one.

(Refer Slide Time: 03:59)



Now, the next example is basically the panel related, so how a panel can be created and the panel can be added we have already done the same example in the previous case also it is explicitly only paneled there. So, let us have the output of this program and here you can see the background of the different buttons can be changed here how the different colors also that is also possible here. And then a background of the frame is gray or white you can say and then whatever the panel is gray and all this thing you have done the setting it is here.

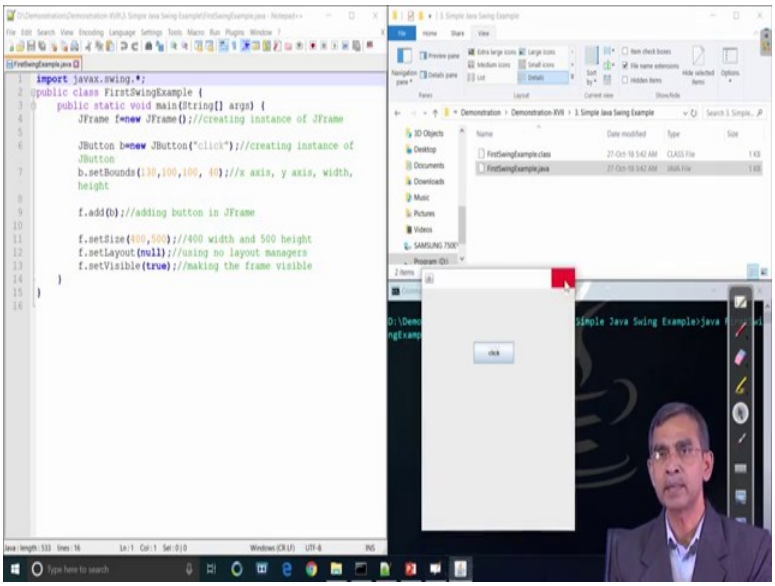
(Refer Slide Time: 04:34)



Now, our next example is basically Swing example, so how a swing button can be created and it can be added into a frame is very simple one, as you see here we first create a frame the frame is titled as not I tell is that in the frame it is here in this case.

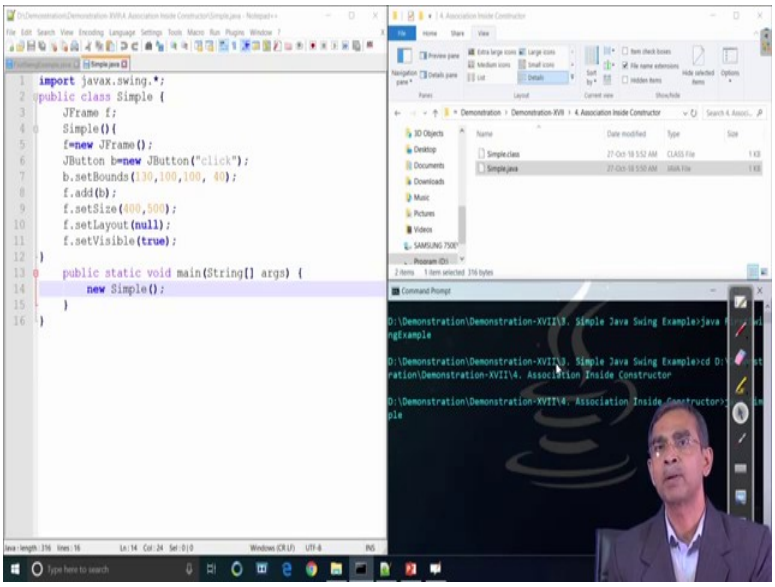
Then we create JButton the button is leveled as click and then we add this button b into the frame f and then finally we define the configuration of the frames regarding size layout and then visibility those are the standard for a configuration protocol that you have to do it. So, this is very simple an example for illustration which includes a one frame a frame contents one button the level edge click yeah that is fine it is very simple on example.

(Refer Slide Time: 05:28)



Now there are many methodologies by which a frame can be included.

(Refer Slide Time: 05:34)

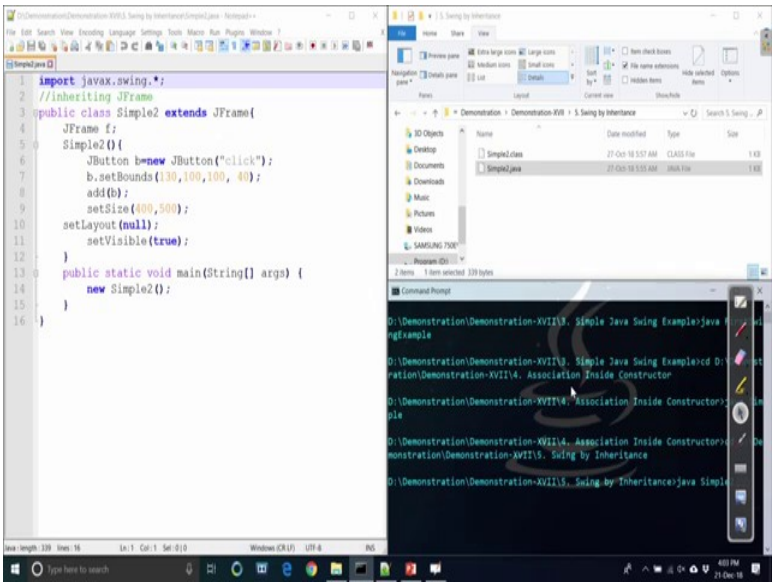


So, there are three ways that a frame can be and then the different methods can be included and finally, the whole GUI can be created. So, it is basically one idea about the creating GUI components inside the constructor method itself and here you can see the simple example here we give the name of the class that we are going to develop for a GUI, program the name of the class is simple and in this case and in the simple class as you see it has the frame declaration. And then the constructor simple and we inside this

constructor we define whatever the procedure needs to be followed in order to create the GUI. As you have done here create a JButton and that button is added into the frame and then frame has been resized and whatever it is there and finally in the main method we can just create the one instance of this class by calling the constructor automatically.

So, that concern only one constructor is they are simple. So this way actually what is the advantage of this thing is that code sharing is possible if you do the class in the different packages and then we can include it. And then from this class we can just create the object and then that object creation itself basically include the different parts of the GUI is to be included in the bigger one. So, it is basically the idea about the bottom-up approach, so the bottom level is all the class declaration for each GUI component and then the top level is the bigger content that it can be designed ok.

(Refer Slide Time: 07:15)



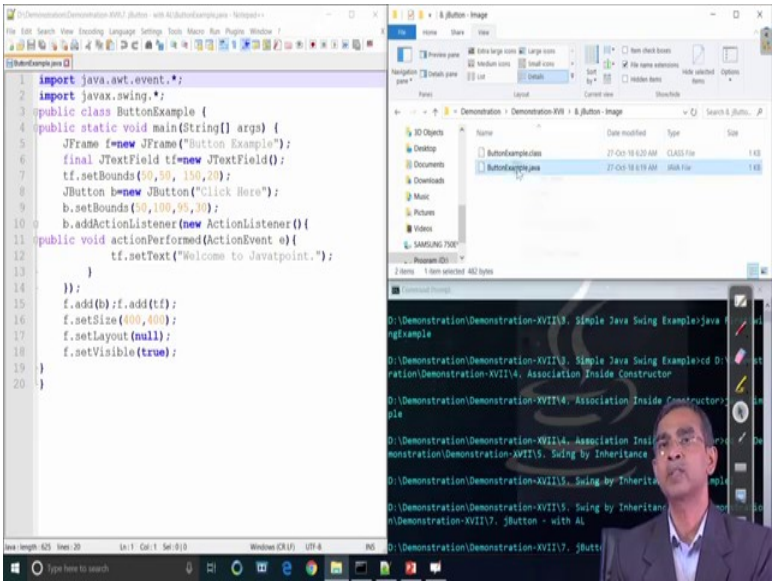
So, this is the one idea about it our next example is basically the same thing, but here we can create an instance and then for that instance we can create the objects they are here and then we can say that there is no need to create the instance of the same class explicitly. This is the one example where we can see without any creating the f object, however we will be able to add some component into it. So, this example basically illustrates this fact actually mostly it is very similar to the previous example. Now say let us see what is the difference here in so far the coding is concerned.

Now here we defined the class simple to which extends JFrame that is the only difference point here earlier is just simple to class, but you have to extend it. That means, it is basically this subclass of JFrame actually sample 2 in this case subclass of JFrame and then we create a frame object and then we define the constructor simple 2 and then we create the button b and then add Button.

Now here you see we add button and explicitly we do not have to mention implicitly that if this add button goes to this frame itself, because the frame is the subclass object it is there. So, this button will be floated on the frame itself which is the main which is included in the main method main class is there and finally, the main method is basically created of instance of this class itself by means of calling the constructor.

So, it is basically same as this one you may find a second the last one example is more preferable that is all this looks pretty simple it looks a little bit abstract, but it is also some programmer prefer this kind of coding it is there and another example is there button with ActionListener. So, even handling the most important part is there, so how we can add the ActionListener to each button like so for an instance for example.

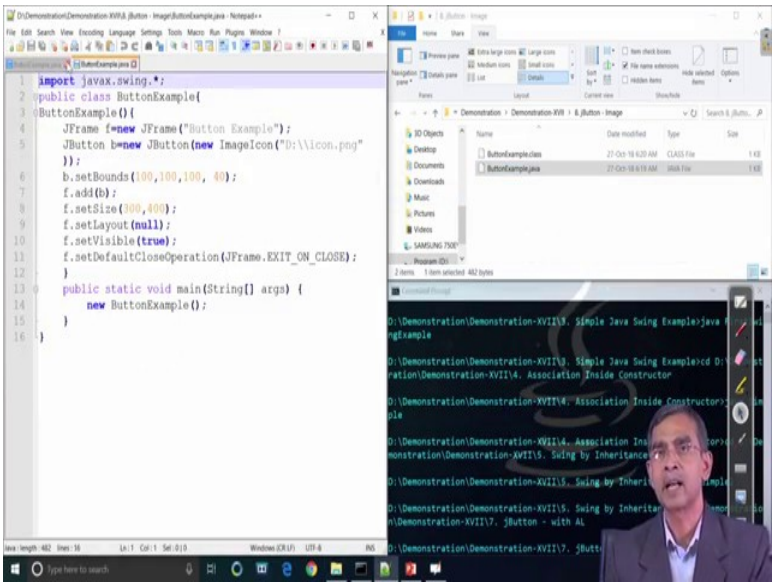
(Refer Slide Time: 09:11)



And in this example we see we create a frame fast and in this frame we add 1 Button name the name of the button that we have given as B and B has been set and configured properly and added into the frame and then we have to add the Listener method for that add ActionListener. New ActionListener is the method right and then it will basically it

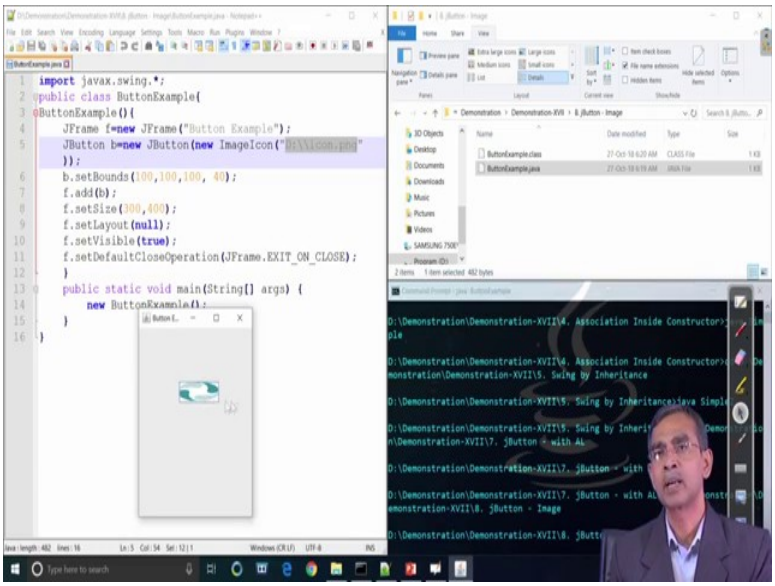
will print whatever the action that depends to is that means if we click the button automatically this event will be executed event code will be even united code will be executed and it will display the text welcome to java point like this one and then finally it will display the results. So, this is the one example that we have shown the Button.

(Refer Slide Time: 10:03)



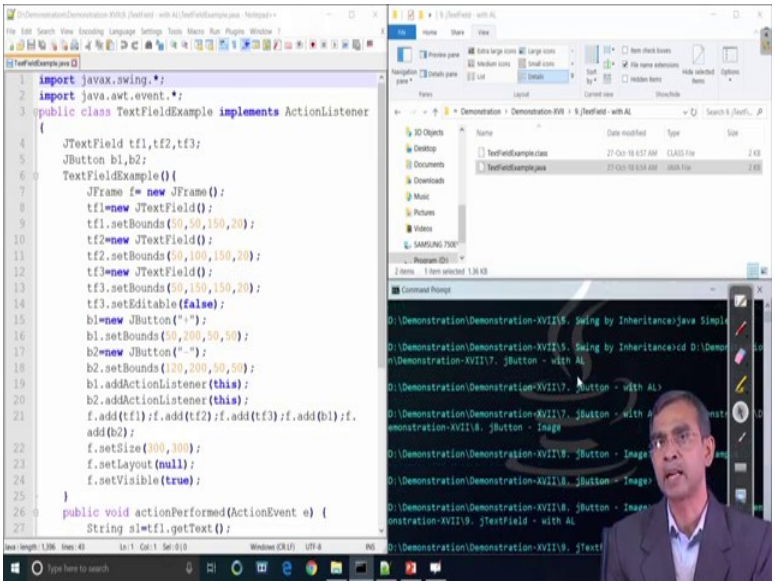
Now, here the next example Button with Image. So, there is one good point here so for the java swing button is concerned that we can add some button image on the button. So, this is the code there only you have to just simply instead of setting the level that we have done in there icon of the image type we have to add it there. So, that we did it here we have created JButton B and then we say then we create one image icon J.

(Refer Slide Time: 10:28)



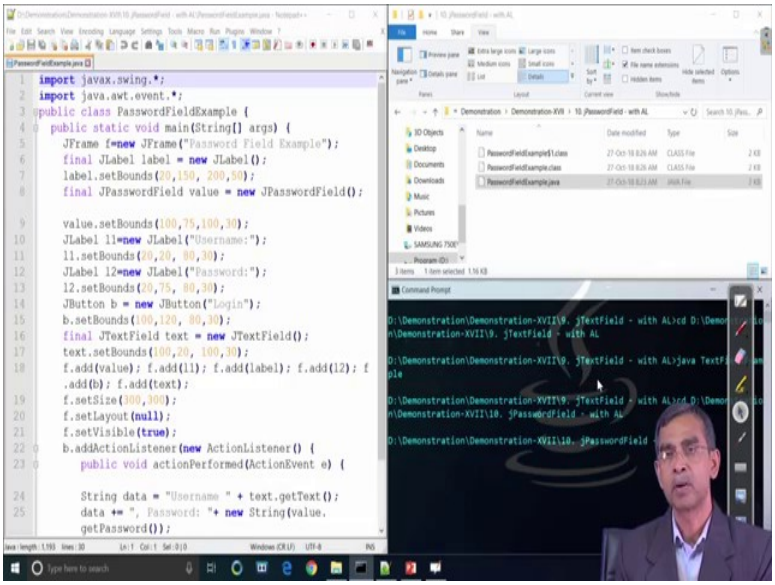
So, (Refer Time: 10:28) while we define the JButton B as a button that time we just create an instance of ImageIcon into it, the ImageIcon basically include the specific location of the image for from where it should get the image file. So, that this image can be included is a bottom as you see the output for this programs you can see is basically include one image that is here and yeah so this is the basically Button with Image. So, similarly some person phase or some other object flower or all these things or some other video click video symbols and everything can be included in this button like this. Now our next example related to the text field in JavaSwing.

(Refer Slide Time: 11:18)



So, how the text field can be added and after adding the TextField how the different activities can be handle real to this TextField.

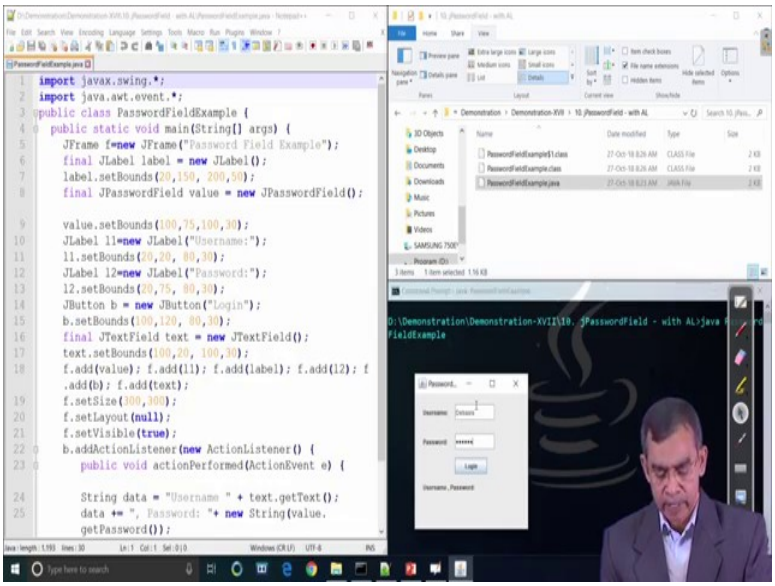
(Refer Slide Time: 11:25)



So, again it is very similar to the simple calculator that we have already discussed while we are discussing about even handling routine managing and they are basically here we will include same thing, but here we just use only the JavaButton and java text fill method it is there.

It is basically the same thing earlier it was without any swing pack swing package here only the thing that we have used in the swing package the output and display and everything very similar to the previous on an example of course ok. You can go through the code and then you can understand it is basically very similar to the previous code it is there and you can find a little bit different the Button looks like little bit different than the Button that is created that was created using awt bar components like this one.

(Refer Slide Time: 12:25)

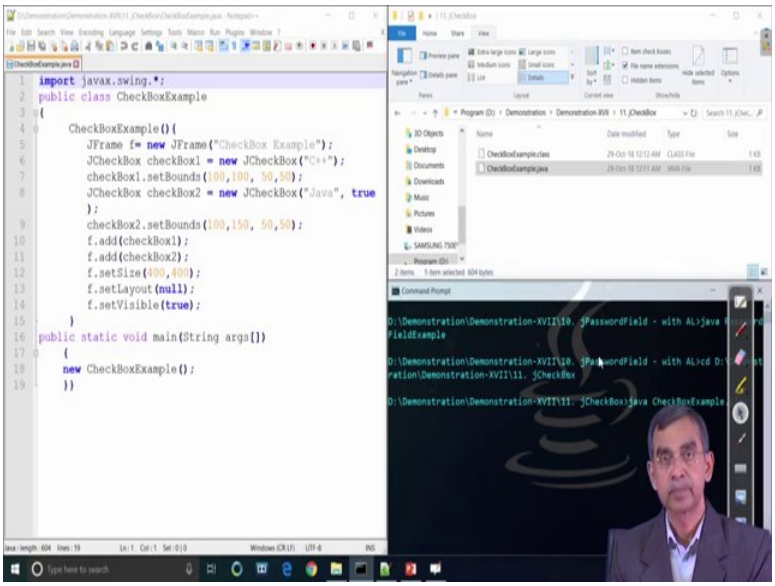


Now, our next example to discuss the other few essential components which is not use a explicitly with a particular specific to the java swing namely password field in addition to TextField for the login purpose as you know whenever user has to log in to some sites or server, they need basically display an applet or window were it includes login and then password.

So, login as a appears as a TextField whereas password has appeared as a PasswordField, so PasswordField can be treated by using java swing here as you see. So, login you can type anything something know even added here, so you can just login as you know you can write a bus is for example and the password is my password.

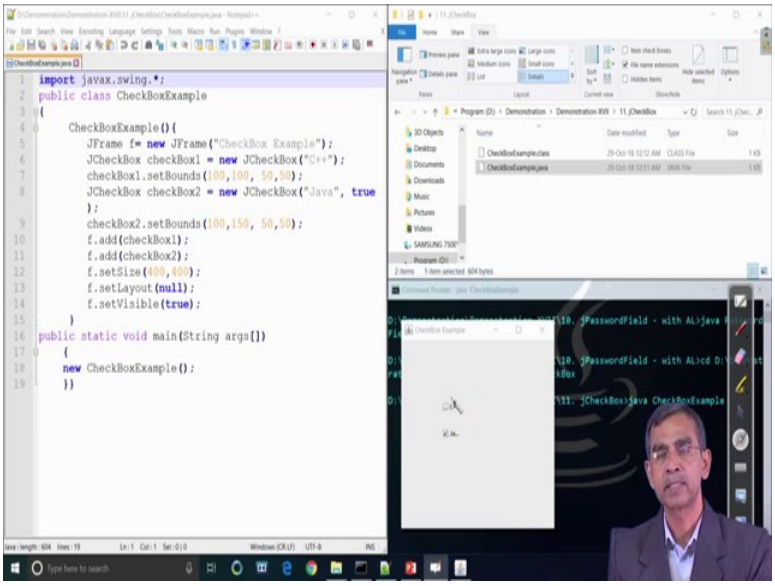
Now, if we click it and whatever the password that we have your program can sense it and accordingly it will pin this one, so password is this one whatever it is there. Now so these are the things that we can action can be generated and then even can be generated and for these event actions can whichever action that you can plan you can add into the interface, so this is about the PasswordField it is there.

(Refer Slide Time: 13:40)



Our next example is basically CheckBox CheckBox using java swing the similar concept only the thing is that JCheckBox JCheckBox and all this thing need to be included here. So, it is basically we create the CheckBox 2 checkbox as the JCheckBox 1 and CheckBox 2 and any CheckBox we add that that is all only we just create the 2 CheckBox and out of which 1 CheckBox as marked and the other is non marked. So, it is basically true and other is there so 2 checkboxes are leveled as C++ and java like.

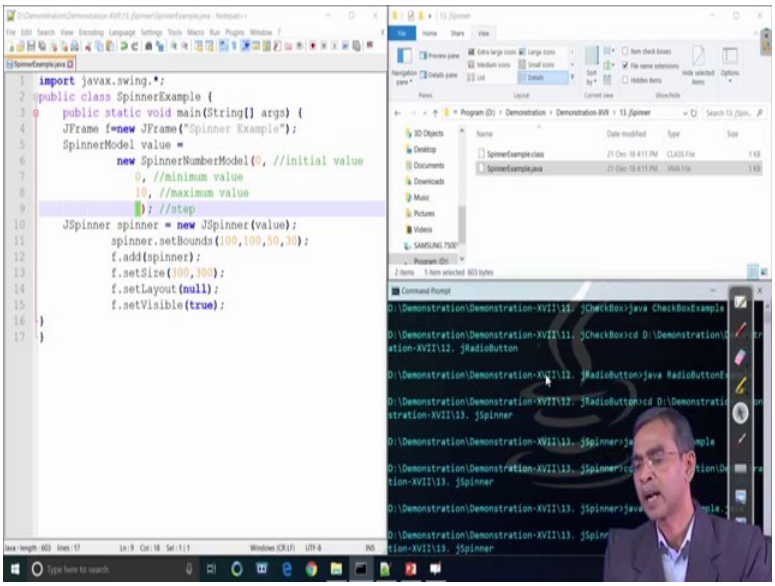
(Refer Slide Time: 14:18)



As you see the 2 checkbox here and C++ and java and if we click C++ you will see the checkbox selection will go from Java to C++ like, so automatically it will be added this one. Now radio button is another example very similar to the CheckBox as you have shown it, but CheckBox has the rectangular view of Checking Box. So, it has here it is called the CheckBox, on the other hand, RadioButton is basically the Button looks like a Radio, that means it is the circular type that is the only difference between RadioButton and the CheckBox otherwise the process method and everything is very similar to the CheckBox itself.

So, checkbox and radio button are the same programs in fact only changing the component type from checkbox to JRadioButton like and this is the code and as you can see this is the things are there and if we click that click will be highlighted here, using the mouse click event needs to be added here in order to sense the event and corresponding that event you can write our own flow code, so that the event handle general code can be written by the programmer.

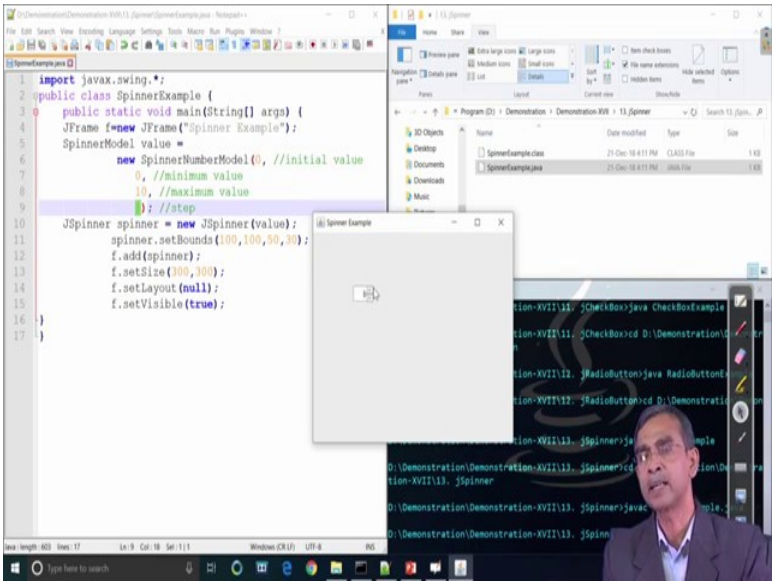
(Refer Slide Time: 15:36)



So, this is the one example of RadioButton example our next example is basically JSpinner, as you know that JSpinner is basically giving the choice of course but a little bit in a numeric form usually. So, here the numeric form and the range and everything can be specified as you see here and if we it is just not exactly the scroll bar it is basically incremental or decremental of the spinner value actually. So this in this

example as we see in this example as you see we create a Spinner JSpinner with the initial value 5, so by default we can change this 0 and then later let us make it 0 and then it has the range of values that it can vary so 0 to 10 and 0 to 10 like and increment is 1. So, if we can give that 0.5 increments that you can understand 0 5 and 0.5 and everything is there so incrementing 0.5 fine.

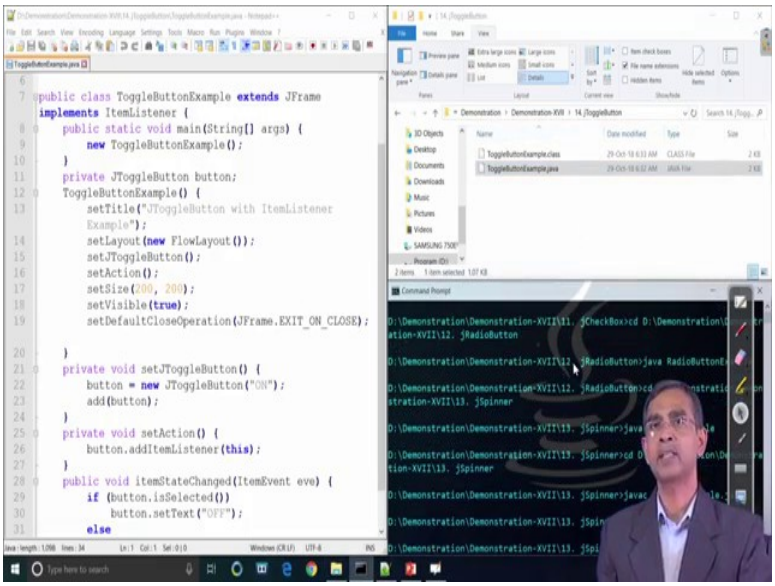
(Refer Slide Time: 16:41)



So, we can learn it yeah so as you learn it as you can see initially 0 little bit bigger in screen, so that we can see it a little bit clearly yeah fine little bit little yeah. Now as you see here now if we the upper click we can use it so it will increase as you see that we are clicking and automatically the spinner values is ranging from current value to 1 increment each time and then again decreasing at any after 10 if you right fine right. So, there will be no increment similarly if we decrement it will go than this one.

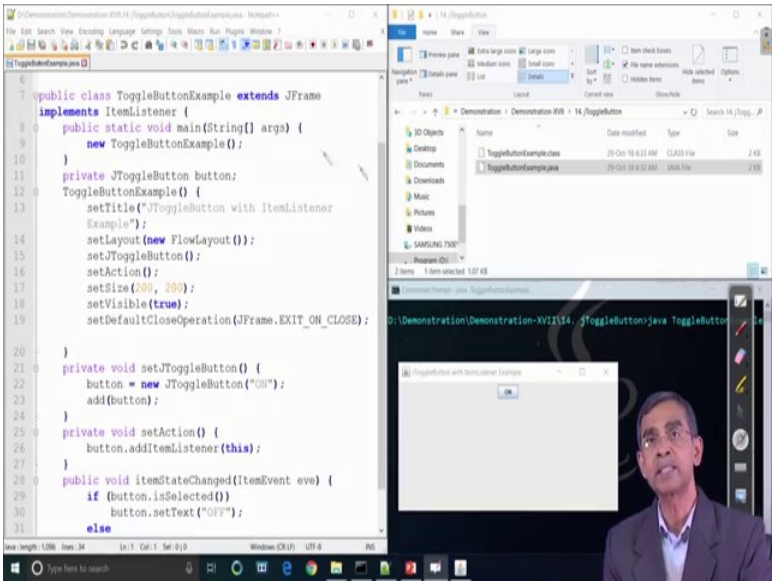
Now, so this is the idea about here incremental value can be any value here we have 1, so if I can to 0.25 so the range of values as precision will be automatically changed and then it will do work for you. So, it is only for the numeric range of course no other range can be allowed for it in floating-point and all these things not allowed there, so this is the JSpinner and our next example is ToggleButton.

(Refer Slide Time: 17:43)



Yeah so ToggleButton basically says button look like, but here actually the button value can be toggled from only 2 values either ON-OFF or whatever it is there.

(Refer Slide Time: 17:57)



Now, in this example that we see here so little bit bigger in size yeah, so nowhere the button is default is ON now if we click this ON then automatically change to OFF. So obviously, the routine needs to be written for that will just come to the code here we can see what is the code that is there in case of the JButton? So, here we have created one example the name of the class that we have is basically an extension of JFrame. So, a

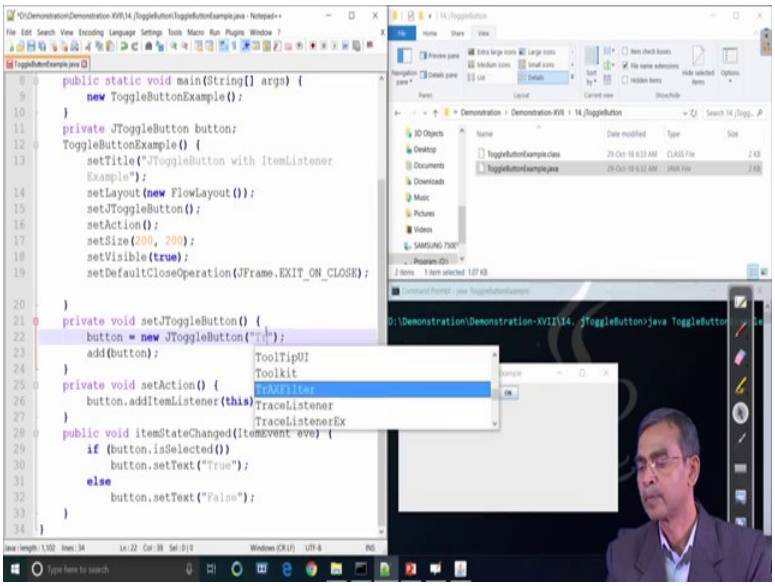
constructor of JFrame needs to be created first so public static void main string is the main method and then we can create the ToggleButton example as a constructor here to invoked this one.

And then the ToggleButton example constructor is now reset by set title, that means is a frame thus title of the frame is the ToggleButton with item Listener actually because, this Button can be is not that action event action listener it is item Listener and then we just defined SetLayout as a FlowLayout you can change any other FlowLayout or if you define the set layout null whatever it is they are. So, default layout planning will be there and set JToggleButton SetAction and then set size set visible true and set default closed operation these are the usual method for the ToggleButton that you have to add it exit or close so it is there.

Now here you just come to the set the ToggleButton method which you have used their how these basically toggling will be there. So, it is ON because it is ON and OFF between and then add button and then set action is basically add item is not is to be set here. So, that you can listen to the event if anything happens to that and then we decide to define the items state change event is there that is the event regarding the item event as I set already and then if selected then it basically set text OFF.

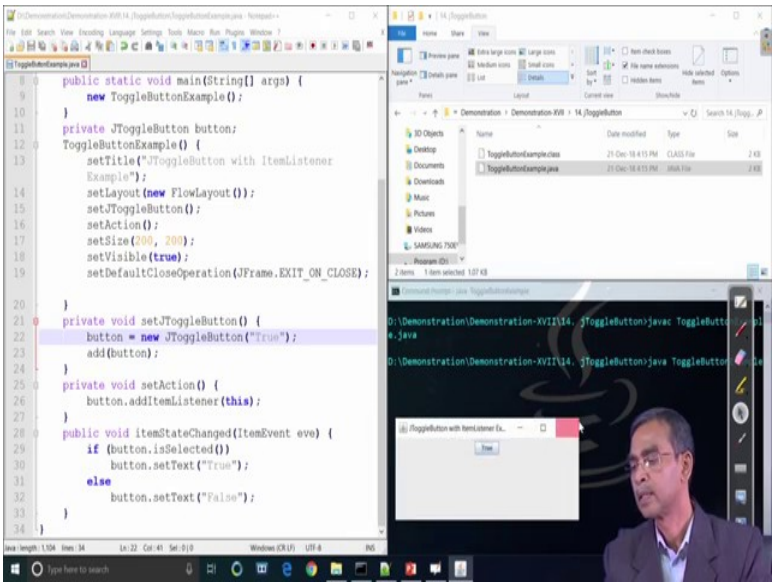
If you don't select anything it will remain the same thing as that now if I change it so on instead on True and then off is false actually we can change any value there right and then here True on instead of ON we can just write True.

(Refer Slide Time: 20:20)



Now we have changed it so you can toggle between True and False like ok.

(Refer Slide Time: 20:37)



As you see here yes now so Button is here we can see this is one, if we click it then you will see that change will be True to False. So, click the button yeah as you see it is there and all the other things are applicable here. Now we have understood few components there few more components are also there that will cover in our next demo class. The next demo class will include few more concepts about JavaSwing components.

Thank you very much.