Transcript: Nested Loops

>> Let's take a look at nested loops. Similar to nested if statements, loops can be nested. For example, consider a while loop. If the body of a while loop contains another while loop, then we think of these as, as nested. And in this case, we've got an outer loop and an inner loop.

For each iteration of the outer loop, the inner loop iterates completely. Let's look at an example. So the question is, how many time will the string here be printed? Here's we've got nested loops, we've got an outer loop that works on count1. And an inner loop that works on count2.

So we begin with the outer loop, while count1 is less than or equal to 10, it was initialized to 1. We go into the body of the loop, and here we initialize count2 to 1. And while count2 is less than or equal to 20, we print out here and increment count2.

After we go through this 20 times, in other words, this loop will end when count2, exceeds 20. We go past the loop here in increment count1 and that, you know, at which time we go back and loop again in the outer loop. So count1 then becomes two and so on.

So the question is how many times will Here be printed? The answer is 200. So each time the inner loop is executed, here is gonna be probably 20 times. And that inner loop is gonna be executed 10 times. So that's how we arrive at 200. Now let's look at a, an example that we might use in an application, of nested loops.

Here we've got, a little program where we wanna read a line of text from the user, print the words in reverse order, and then query the user to repeat, and enter more text if desired. So, here's the strategy. We're gonna have an outer loop to read lines of text and then the inner loop will store the words in an array list.

This is so we can get them back. We will print the ArrayList using its default two string method, in a print statement. And then we're gonna print the individual elements of the array, the array list in order using a while loop. And then we'll print the results of the ArrayList in reverse order, using a while loop.

And then we'll, ask the user if we wanna repeat, and we'll go back and, and check the outer loop, to see if, if the user indeed did want to repeat. So, let's look at the little program called ReverseWords. You'll use these concept in some of your projects. So, here we've class ReverseWords and main.

And notice we're importing the Scanner and ArrayList, we're using both of those in this program. In main here we create a Scanner on system.in, the keyboard and assign it to scan. We create an ArrayList called inputList. It's an ArrayList of string, and then we create a Boolean, getData, and set it to true.

Now this will be the Boolean expression in our outer while loop. Now this says while getData, which begins as true. We're gonna go into the body of it and we're gonna ask the user to enter a line of text. We'll scan that line in to input, just read the line in.

And then we're going to, create another Scanner called lineScan, on this line that was input. And then we're going to, with this new ArrayList, we're going to go through a while loop and scan each word in. So here we say, if lineScan has next, this becomes our Boolean expression.

As long as it has more words in that line, we're going to say inputList.add and scan next. So the lineScan.next scans for the next word in that line and then we add it to the, input list. And so we just continue through this loop until we run out of words.

And at the end of the loop, we have all of the words read in and stored in our array list called inputList. And then next we're gonna print out the list using the ArrayListToString, implicitly. And that is, we're just gonna say System.out.print line inputList. Since inputList is an object, an ArrayList.

The, its two string method gets called and we'll see the items printed out all on one line. But next we want to, print them using a while loop, and just separate it by spaces like they, looked when they came in. So here we set index = to 0.

And then while index is less than the inputList.size, that's how many words we have. Then then we want to go through here and print out, that item in the list. And we have to use a get, so we say print and then input, lot, List.get index and then we concatenate a space with that and, and we've printed the first one out.

Index is incremented we go back up as long as it's less than inputList.size, we, we print the next one, using a, get. When we're done with that, now we wanna print them out in reverse order. And so here, we say, System.out.println, we skipped a line there. But now we wanna set index to inputList.size -1.

So recall, the indexes begin at 0, so if we have five words in there. They're indexed 0 to 4, so 5 is the size and we wanna set index to size minus 1, or 4. And then we go through our loop as long as index is greater than or equal to 0.

We we print out the item that we get at the index. And then we decrement index and go back and keep looking as long as it's greater than 00 we print. Eventually we get all printed out, and then we check to see if the user wants enter more data there.

And we ask them to enter a y or an n and then we read that in. Actually, we, read it in here with a, scan line, and then we ask if more data, what was just read in, = ignore case of capital Y. And, and, this returns a Boolean value.

And so getData get set to true or false depending on what the user keyed in. And if the user keyed in a, a y, ignoring case, then getData gets set true, otherwise it will be set to false. And so we go back to the top of the loop here and we check to see if get data is true or false.

If it's true, we go through the loop again, if it's false, we jump past the loop and just print out done. So let's execute this in a Canvas. The Canvas file came with the example here for you. And so we're ready to, scan in. I'm gonna move this over just a little so we can see more of the program.

Let me move this up some, all right, so it says, we're about to scan in. So I'm gonna step, I'm gonna do single step this instead of play. Play would actually step into everything, take a little longer. So just single step and, it says, we, we set getData to true.

We can see it here and now we're about to read in our first line. It says, enter a line of text, so I'm going to enter, let's move this up a little and we can see it. This is a test of loops, and I hit Enter or Return.

And, and then step, then I should see it there. So input now just got set to the line who was read in, and you can see it here and also on the Canvas. This is a test of loops. And now we want to create a new scanner called scanLine based on that line of text, so we're just gonna put that in a scanner, and there it is.

We'll see it when we try to access it the first time. And we create our self a new ArrayList. And then we're gonna scan while lineScan has next. And notice lineScan is at 0 right now and hasn't, we actually haven't seen the initialized buffer, we're about to right now.

So we see its got the words, this is a test of loops in it, can't see it all but it will move over for us. And so inside the loop here we're gonnacall scan, lineScan.next and that's gonna pull off the next word separated by spaces, the, default delimiter is a space.

When we're scanning like this, when we scan next. So, its gonna pull off list and then it's gonna add it to the inputList. The inputList is our ArrayList here of strings. So lets just step there, so we've pulled off list, there it is and notice that the position in scanner has moved, so now we're looking at is.

We're back at the top of the loop, and I'm just gonna step through this and you'll see us pull off each word. There's is, a, test, and here's of and finally loops. So, now when I, click on the step button, it says while a line scan has next, we're gonna be at the end of them.

We've got them all. So this should jump out of the loop or pass the loop. And now we're just gonna print out the ArrayList. And again, this is using the, two string of the ArrayList. And there we see, they're the items in the array list. This is a test of loops.

Next, we want to use a while loop to print them out in order. And so I'm gonna move this over just a little so we can see this. We're gonna keep our eyes on inputList. Notice here we're setting index to zero, and then as long as index is less than input.size, which in this case is six, we're gonna go through the body of the loop and we're gonna use a get.

And we're gonna say input.get and we're gonna get the, value at the index, beginning at 0. Each time we go through the loop, we'll get the next one and we'll index as we go. So, we'll see these printed out down below and we'll see us march through the ArrayList.

So I'm gonna step, so index is at 0, I'm getting that first element there and I just, printed it out. And we increment index. We see over here now we're looking at is, we're about to get and print it out. There it is, index goes up. And I'm just stepping through the loop and as I step through the loop, you can see the, march through the index there or through the ArrayList.

Eventually we get to the end there, and we're gonna jump out printed the last one. And so now we're going to print them in reverse order. So this is a little bit different. Now we're wanna to set index to input size minus one. So input size is 6, right here, size is 6.

And, 6 minus 1 is 5 and sure enough this is where we want to start in our, in our, attempt to print these out in reverse order. So let's step through here and see this, this guy march. So there's index, and you'll see it just march back down to zero here through our loop as we get each one and print them out in reverse there.

So there's loops being printed out, of is printed out, test is printed out. We're going through here and, or about to print out, is, and one more, this. And it's printed out at the bottom. You can see we've got them out in reverse order here. So, one more step here and we're out of the loop.

And, now, we, we're, now at the point where we're gonna ask the user for more data. We're just gonna say enter a yes or no and print it out there and we're about to scan the next line for more data. The question more data or not, and I'm just gonna say no here.

And so we can see there that more data is no. And then notice getData, which is Boolean, is being set equal to more data.equals ignore case Y. So more data, ignore case does equal y this would be true, otherwise it'll be false. So, I've entered no, or n, so this is going to be a false.

And we see it here, it's been set to false. And now we go back to the very top of the loop and get data is false as we said. And so one more step, we won't go in the loop, now we're going to go beyond the loop. So now we're at the end of the loop, and we're about to print out, done and, and we're finished.

So this is a good little example. Be sure to go through this, and understand, a number of things here, how outer loops and inner loops work together. And also how we scan things in from the keyboard. We put them in a separate scanner, and sort of march through them, putting them into an ArrayList here.

And then we print it out, the ArrayListin, in forward order and, and also in, in backwards order.