**Comparison Operators**

[00:00:00.49] [MUSIC PLAYING]

[00:00:08.34] RYAN AHMED: Hello, everyone, and welcome to this lesson on comparison operators. Comparison operators compare two values and return true or false. Comparison operators are represented in double equal sign. Remember that assignment operators were represented in one equal sign, and it performs a completely different task. Assignment operators simply assigns a value to a variable, while comparison operators compare two values together and returns true or false.

[00:00:43.89] Please note that comparison operators will become very useful when we cover if-else statements and while loops in future lessons of this course. Here are the key learning objectives of this lesson. Apply comparison operators in Python to compare two operands with one another. Understand the difference in syntax between comparison operators and assignment statements. So let's head over to our Jupyter notebooks and get started.

[00:01:18.32] [MUSIC PLAYING]

[00:01:25.44] All right, so right now, we are in the Jupyter notebook titled Comparison Operators. So comparison operators compare the values of two operands together and return true or false. Let me show you an example. Let's assume that I'm going to define two variables. First one is I'm going to call it revenue\_A. This is simply the revenue for a company A. And I'm going to set, let's say, $1,000 in it, as an example.

[00:01:57.03] And then I'm going to define another variable. I'm going to call it revenue\_B. And then I'm going to put $500 in it. And please note that as a quick reminder, one equal sign indicates that this is simply an assignment operator. All we're doing is we're assigning 1,000 to revenue\_A and 500 to revenue\_B. That's all what we're doing.

[00:02:21.91] Now what I wanted to do is I wanted to compare the revenue for company A, revenue of company B. To do that, I'm going to leverage comparison operators. And this is simply the syntax to compare two variables together using the equal equal sign. So here if I say revenue\_A equal equal revenue B, this is simply the comparison operator right here. And this is just one of many. I'm going to walk you through all the options coming up in the next slide.

[00:02:52.51] But for now, simply, I'm saying, OK, compare revenue\_A to revenue\_B. And the feedback, or the output, is going to be one of two options. It's either true or false. Simply, here I'm saying, is revenue\_A equals to revenue\_B or not? And because revenue\_A is 1,000 and revenue\_B is 500, both of them are not equal. And that's why the feedback, or the output, is going to be false here in this example.

[00:03:22.50] Well, what other options do I have? Well, if you scroll down, you should see all the different options available in comparison operators. So if I say, a equals equals to B, here I'm asking the question, is A equals to B or not? If the answer is true, I'm going to get back true. If not, I'm going to get back false.

[00:03:44.56] If I say A greater than B, this is simply a question. I'm going to say, OK, is A greater than B or not? If A less than B, this is simply the syntax here in Python. And this is a description. Here I'm saying, is A less than B or not? If I say A greater than or equal B, this is simply the question. And if I say here, A less than or equal to B. And finally, I have A not equal B. And this is the syntax-- simply an exclamation mark equal, which indicates not equal.

[00:04:19.57] All right, so let's go ahead and jump into the code and show you in details couple of examples. All right, so let's go ahead and define revenue\_A. So I'm going to say revenue\_A equals to 1,000. And then I'm going to say revenue\_B. And I'm going to say it's equal to 1,000 as well.

[00:04:37.17] And then I'm going to compare revenue\_A to revenue\_B. So again I'm going to say, is revenue\_A equal equal revenue\_B or not? So this is simply the comparison operator that I covered before. And one equal is simply an assignment operator. I'm assigning 1,000 to revenue\_A, 1,000 to revenue\_B. Here I'm asking a question or comparing these two variables together.

[00:04:58.68] So if you press Shift and Enter, then you will get the answer is true because 1,000 is equal to 1,000. I want you to go ahead and maybe change that number. Instead of 1,000, change it to 500. And run it again. And simply, you will get the answer is false because revenue\_A is not equal to revenue\_B.

[00:05:21.86] What you could do next is that you can use greater than or equal as follows. So here I'm going to define revenue\_A. I'm going to put 5,000 in it. I'm then going to define revenue\_B. I'm going to put 1,000 in it. And then you can say, is revenue\_A greater than or equal revenue\_B or not? Well, the answer is true because 5,000 is in fact greater than 1,000.

[00:05:46.51] What you could also do here is you can compare by using the not equal sign, which is what I covered before in the table. But before we do that, I just wanted to do a quick reminder because many of you might make this mistake in the future. You might kind of confuse the comparison operator with the assignment operator.

[00:06:07.25] So when I say revenue\_A equals to 10, if you just use one sign, one equal sign, that means it's a variable assignment. I'm putting 10 in revenue\_A. However, comparison operators, if I put equal equal, you will simply compare two variables together and end up with true or false. So in the future, if you simply get some error code when you're trying to leverage comparison operators, you might have just missed that and maybe used one equal sign instead of two.

[00:06:35.68] I just wanted to just quickly iterate on that. So if you say revenue\_A equal 10, check out what's revenue\_A. Simply, you will get the value of 10.

[00:06:43.24] OK, finally, you can simply compare revenue\_A and revenue\_B by saying not equal. So if I say 1,000 in revenue\_A and I put 1,000 in revenue\_B, and then I said, is revenue\_A not equal to revenue\_B, if you run it, then you get the answer is false because, well, revenue\_A is in fact equal to revenue\_B.

[00:07:06.37] All right, so that's it. That's simply all I have for this lesson. I hope you enjoyed it. In the next lesson, I'm going to have our coding lab or practice opportunity. So please go ahead, give it a try. Write your answer in here. And then I'm going to provide you with a detailed video explanation of my proposed solution. I hope you enjoyed this lesson, and see you in the next one.

[00:07:27.97] [MUSIC PLAYING]