Python - Module 6: Conditional Statements

Python

Module 6: Conditional Statements

Video: Conditional Statements Intro

Welcome to Python sessions.
In the last session we have discussed about
operations with all data types.
So in this session we are going to enter into
conditional statements.
Exactly. The purpose of conditional statements is
based on given criteria.
The creditor means condition.
Some block ops statements to be executed
if condition falls.
Some other block ops statements to be executed
here, a single block, uh, may single statement
or multiple statements.
If condition falls. Some other block op statements
to be executed.
I keep it for you in the notes.
So you can classify these e statements as below
a simple statement
and if and else statement.
The third category here, nest strips.
Nest strips and the,
the regular format of these three statements I will explain
it is simple.
If the format is like this,
if condition,

you know python coding rules.
If any line is expecting a sub statement,
the line should end with column then
the block of the statements.
This block my end single or multiple statements.
Example.
Suppose I'm taking a value as 10 B value 20
and I'm giving a condition if E greater than P.
If this condition is true,
I'm printing A is big.
Then one more statement. I'm printing in this block.
I'm giving two statements,
B small.
So this block will be executed
if the given condition is true.
Okay? Suppose if we given condition falls,
nothing will be exhibited here.
Okay? So try
to evaluate this one condition.
Evaluate and be valued. That means 10 greater than
20 is the condition.
So in this example, the given condition is false,
so nothing is executed.
For example, the value a hundred B value is 50.
Now what about this condition?
A hundred greater than 50.
Now in this case the condition, given condition is true.
Then it prints like a hundred is big, 50 is small.
So if condition is false, nothing is executed.
If condition is true, then only this statement is executed.
This is simply format.
Same thing.
Defend statement.
The second format,

defend else.
That means if the given condition is true, some statements,
condition falls, some other statements,
which format is like this?
If condition
think that is a block one.
That means all these are true statements.
Yes, some other
block of statements I'm giving treated as blocked.
All these are false statements.
So for these different examples we'll see
same previous values.
I'm taking a value 10 B value funding my condition is here.
If greater than B,
print Y is big,
yes,
print B is big
of course, in this example I have even only one statement.
If condition is true, a big's statement will be printed.
If condition falls, B is big will be printed.
So for this program, what is output?
What about year value then B,
value 2010 greater than 20 falls.
So that control will come to Y spot.
So what is the final output?
B is big B means 20, but 20 is big, is output of this.
Similarly, multiple statements.
I wanted for the same example. B value 10.
That's a B value 20.
So before printing you need to check the condition.
Then what would Small?
Small,
yes, I'm reversing it.
Print B is big.

Then Y is small.
So for the given data, that means given values of A
and B, what is output here?
Condition falls. That's why you're coming into yel spot
and the L spot, there are two statements.
So that uh, output will be like this, 20 is big
and 10 is
small.
In this way, conditional statement will be working
and there is one more format of it that is nas.
So what exactly the meaning of is if
within that other, if
one more.
If you can say if some condition,
if condition is true, you are checking
with one more condition.
Again, this is true. One more condition, you are checking
again else.
This is spo. This.
If this is IL four, the second if,
and this is I lpo the first.
If again I'm checking with other conditions.
For example it all the things based on the,
if the condition is true, you
are checking with other condition.
If condition is true, we
are checking with other condition.
Same thing with condition falls.
Also, you can check with our condition.
So that format in the next screen we'll see.
Look into this. I'm giving a condition here.
Think that it is true.
Some statement is executed if condition falls yells
and then if you are writing in stuff of that in Python,

the AIF combination,
the ALIF combination is AIF
Again, this condition also falls.
Again, one more LF than condition three.
In this one. This is
what simply NAS and alis.
So based on this nas some condition, some example will.
Once this explanation is done in the next session, we go
to the practical part of it.
This time I'm going to take 10, three values, value
and B, value 20, C value 30.
Uh, finally I want to find out what is the biggest value.
Also, I'm keeping EAB into the
competition, which is biggest.
I'm checking if you greater than me.
This is my condition.
If this condition is true, definitely A is big, B is small.
Then A needs to be checked with C.
Then if the condition is true, one more if condition, I'm
if you get it, then C.
Okay. Now if the second condition also true means definitely
A already bigger than B.
Now A is big, bigger than C also.
Now finally A is biggest rent.
A is big.
For example, if the second condition got false,
second condition falls means what?
A already bigger than B. Now C is bigger than A.
Then
print C is big.
So it's all that robot. Your first condition is true.
For example, first condition itself falls,
then come to this one.
Yes, I will do in the next line.

Look into this. The S were
is primary first.
After that, immediately I need to check
with one more condition.
In Python, the S and combination is yellow.
That's why here I'm giving yellow leaf.
The no need to say this one, right? Yeah. Okay.
First condition. False means what? B is bigger than you.
Then B needs to be checked with the competition with C.
Then bigger than C,
if this condition is true, print B is big.
Every is false. Print,
definitely C, S, P.
Now for the given values of 10, 20, 30, try
to evaluate this condition.
What about 10 greater than this value is 10
and B, value is 20.
So condition false because of falsity is coming
to it is reaching LC.
Okay. Now, B, value 20.
C value 30, condition two
or false 20, greater than 30.
That is false. That's why again, it is reaching L spot.
Now what will be printed funnel output is
30 is big.
Let's test with the other different values so
that easily can understand.
Now let's give the values in a place of uh, yeah,
I'm keeping a hundred in the place of B 250.
In a place of C I'm giving it 240.
I'll try to evaluate this
with without given values, not latest values.
Yeah, a hundred B, two 50,
a hundred greater than two 50 condition falls.

Again, it is reaching yes spot.
Now, B, value two 50 C value 2 40, 2 50.
Greater than two 40 condition is true.
That's why what will be printed two 50 is
okay, let's make it thousand this time.
This time it is thousand B value. Thousand B value two 50.
I think that it's two nine. What is the biggest 1000 value?
This yay thousand B two 50,000 greater than two
50 condition through it reaches this one.
Understand now competition with the C
thousand C value two 90,000 greater than two 90.
Again, condition through so that what we printed B,
that means panel output thousand days.
Big focus, especially if you look into this
example, nest.
If within that, if that is covered.
Again, both the things have been covered in this one.
Okay, simply these are the three formats of the regular.
So what are the three for points I try to recall.
The first one is simple statement.
Second one is if and else. Third one is nest, dips and L.
Okay, so in the next session we see practical part
of all these things step by step.
Okay, thank you much.

Video: Simple IF & IF ELSE Lab

And the types of uh, conditional statements,
like a simple if statements and
before that, uh, if and L statements.
So in this session we see practical, simple, if
and else
good plan.

So currently with the runtime time, five
possibly demonstrate.
We see demonstration about simpler.
I'm taking a value this hundred.
My question is like this, pay greater than
print
bigger value.
If condition falls, I'm not giving anything.
Just check this. A hundred, a hundred.
Greater than 50 condition True
because of two bigger value it has printed.
For example, I'm changing the value, okay,
in the next cell I will change it
and value.
So 40 greater than 50, the condition is false.
Here, condition is false, means nothing is executed.
We don't get any output of yet.
Okay, but condition falls also I want to execute something.
Then you have to give if and l statement.
If L statement, if condition
through some statement may be singular, multiple.
If condition falls, some other
statements keeping the same.
For example, a value a hundred D,
value one 20.
If you greater than B, print A is three.
If condition falls, then print
B is now check for this values a
hundred greater than one 20.
Actually this condition is false because of
False it came to the L part.
The print BS begins one 20 big.
Okay, let's do a trial
with other different values in the next Excel.

I repeat the same
code, it's 900.
We value 900 B values one point.
Now condition is through this time 900 greater than one 20.
The condition is true. Then it prints 900 is big.
For example, we both are equal values. I have hundred.
Now what is the output? What? What are the first condition?
A hundred. Greater than a hundred.
The condition is false here
because of false again, it reaches,
suppose you want to display three categories
of output like uh, a big B big or both are equal.
Okay? Such kind of things you want to print how to go
to the nests.
Okay? So in the next examples we discuss about nest labs,
but now let's continue with this one.
Now even the values are equal.
Definitely a hundred greater than a hundred. That is false.
Then it is printing 103.
Of course in these two values,
a hundred hundred is definitely a big, got it.
Similarly, if condition through multiple statements,
you can expect that is a block of the statements.
I'm just modifying that statement.
If this condition is true, print A is big,
then print B is small.
It also, I'm adding one more statement.
If condition is false, A is B is big, then A is small
like
C this at the same time,
in fact both are equal.
To handle that kind of equality, you need
to add one more manage.
So in the nest steps that is possible.

In the next session we discuss about nest steps.
Now with some scenario based example, we'll see
I'm taking a
student marks,
let's say have three subjects,
young one and three and three.
After this a pass mark in each subject
50,
the pass mark is subject is 50.
Now based on the marks whether student passed
or fairly one to five,
let's say student name something soldier
and M1 score is 70 M two score
is 30, M three score is nine.
Now I'm checking the condition if yum one
greater than hundred, but the power is the pause mark.
We said 50, minimum 50, greater equal,
two 50.
Then M two should be, should also be greater than article
equal two yam three greater than article two.
So here more than one condition, more than condition.
You need to give operators here
every condition must be true.
And then if
all these conditions are true, I'm rent
is passed,
which is my true state.
Yes. Print
name is five.
Okay, now check it.
Try to evaluate the condition for the surgeon here.
What about the first condition?
The monster score is 70, 70, greater than equal to 52.
Second condition, M two value 30,

30 greater than R equal to 50.
False one. False enough total will become false.
Got it. Now what is the final output here?
Surgeon is fake. Just check this.
Got it. Let's test with other data.
Let's say name is score is 60.
Now condition rule, second condition, rule,
part condition also all conditions are true.
Final result is now
what will be printed is passed.
Got it. So this is
what simply defend state
and one more example, let's take
your bank example.
You think that you are a bank customer,
your account number is one zero, okay?
Your balance amount is 10,000.
You want to withdraw
11,000.
I'm taking the condition.
Now, if a balance is greater than
with the dry
balance or greater than
or greater than a equal, then
I'm printing some statement.
Account number
has sufficient balance.
The memorial statement, I'm printing multiple statement.
Then immediately balance equal, equal to the two minus SQL
to automatical assignment.
Okay? All we discussed in the operator sessions,
balance minus SQLs, et
and also one more print statement.
I'm giving amount with the draw. Successful

is success.
It's all a sorting if the given condition is, got it.
For example, if the given condition is false, right?
It is printing you in tion.
Once your balance is
just balance and display.
Okay? Now here one account number is 1 0 1.
His balance amount actually 10,000.
He wants to get the draw 11,000. Check this condition.
10,000 greater than Oracle equal to 11,000 condition falls.
So what it'll show you inception for your balances.
10,000. Okay, now check this.
What is output in sub?
Let's do the test with other credit.
Now you are giving 9,000.
You want to withdraw 9,000. What?
What is your balance amount now? 10,000.
Now what is happening?
What will happen here? What is the condition?
10,000 or greater than to 9,000. True or false? True.
Because of rule it is displaying MS
assumption balance
and immediately what is happening from your exit balance,
it is directing with the dwell.
That means that 10,000 is a balance from
that 9,000 will be a deducted.
So that our latest balance now thousand and
after that we are printing confirmation as amount
with the dollars is successful.
And now immediately I'm displaying print the latest balance.
Your latest balance.
Again, we are 20 grand balance.
Okay, now check this output
actually as per given that I have 10,000 balance

and we are trying to withdraw 9,000.
Now here what happened? The first message
you have tion balance the amount with are successful
before that amount is with,
and then your latest balance is what all on
that way in real time you can use if and L statements.
Okay? So more,
more complex scenarios in the coming following sessions will
be seeing about this one.
This is simply a a small demonstration about if event else.
Okay? In the next example, in the next session
we'll discuss about nest statements.
Okay? See you in the next session. Thank you very much.

Video: Nested IF & ELIF Lab

Welcome to Python sessions.
In the last session we discussed about
a simple statement and if
and all statements with some scenarios, bank scenario
and one mark example.
We have seen that. So in this session we are going to work
with nest steps.
Depends if one condition is two
other condition to be checked.
If condition falls, one more condition to be checked.
So these kind of things are called negatives.
Let's say the practical part of it,
starting in a new notebook
is connecting with one time.
Now please wait.
This time I'm taking three variants.
I want to find biggest among the three variables.

So a value a hundred, B, value one 50 C value one point.
So for this I want to find out the biggest
first I'm comparing
and if greater than B is my condition.
If this condition is true, one more among A is big,
but again needs to be still need to be checked with the C
for that one more condition.
If within that one more, this is gone.
If you greater than C, if this condition is also true,
then big equal to yeah.
For example, if the second condition is false, then
greater than C, false means definitely C is big.
Why not with B? B,
B is already defeated in the first condition on That's
what the equal do we'll C, right?
Resolving will happen if the primary condition,
if if the first, if condition is true,
suppose first if condition got false, yeah,
then I'm checking with one more condition in Python Ys.
If that combination simply, you can write it as, okay,
suppose first condition falls means B is big among A, B.
Now B need to be checked with C.
If this condition is true, biggest being
big is C,
because competition between now B and C.
Okay, now finally I'm going to print the biggest
base, big
check the values given values.
Hundred one 50 to hundred, which is the biggest, which 50.
So definitely one 50 is output.
Here the biggest is one 50
and try to evaluate this condition.
What about this? I'm not greater than one 50 condition.
Let me explain. Clear

is what the given condition.
Check it the first step.
What about this condition?
I'm not greater than one 50.
This is false because of false.
It goes to L spot. What is the L spot for it?
They sell now be greater than C. What is the B value?
One 50 C Value one 20. Condition two or false? True.
Because of rule this will be executed,
then it'll come out of the state.
Then final printing is the biggest is one fifth.
For example, if I have given the values like this,
think that its values one certain.
Now check the first condition 100
greater than one 50.
This is false. So control comes to
I live now.
B. Value one 50 C, value
one 70, condition two or false again, false.
So that it goes to yes fault. What is the statement here?
B, equal to C, then it's value is one, seven.
Okay, now let's make uh, as biggest,
let's try with the sun value, which is 200 for the
now what happened to this condition?
200 greater and 1 52.
Because our route is coming to this spot,
now we value one per 200 C.
Value one 40 again, then yeah,
then it'll come out of the state.
Now finally, what is output? The weak is 200.
This is one more trial we do
and taking your name,
supposed name care
taking your salaries, your salary,

one lack 20,000.
So 12,001 lack 20,000 is the salary.
Now I want to find out salary grade.
The condition is like this fine
grade of employee
based on salary.
The rule is like this.
If salary greater than articles to one lack,
then that is grad eight.
If salary greater than articles to one lack
or greater than articles to 75,000
and is less than equals to one, lack less than one lack,
then grade is big.
If salary greater than articles two 50
and less than 75.
70 5K.
Then grade is C.
If salary less than 50, everything
then grade is C.
Now I have four options.
Okay, so based on the salary, I need
to find out what is a grade.
Let's say in the first example, I have one. One lack 20,000.
So for this I'm going to write the condition.
If I have only two options, a simple if
and yes that is okay,
but if I have more than two options, I need
to use master stepss here.
First I targeting grade here. Past time targeting grad A.
What is the condition for grad?
A salary greater than oracle Equal two, one lack
if salary greater than Oracle equals two.
One black. So let me write in the plain text letter.
I will keep it to here. If salary greater than oracle equals

two, one lack, then grade equal to,
yeah, suppose condition falls.
Definitely the value is less than one lack.
Then else if you know EL and if is here, yeah, leave.
Then I'm checking for the B.
If first condition falls, I'm checking for B grade.
What is the condition greater than equal to 75,000
and salary less.
Then one like no need to give the second condition.
Why? Because if foster condition is false,
then only you're reaching the second condition.
Foster condition falls means definitely the value is
less than one, like a lot.
That's why save S greater than equal to 75,000.
Then grade equals to B grade
still L save salary greater than Oracle equals to 50,000.
Definitely that is less than 75.
Then grade equals to C grid.
Suppose first of all, second, all third falls.
Then else definitely remaining all values.
Remaining value is definitely below 51.
Then grade equals to D grade
name salaries.
I'm printing that salary and
the grade is the grade.
Hmm? What is the salary for the name? One. Like 20.
What about this first condition? Two?
One like 20 is greater than one, like two.
That's why the grade eight. But now grade contents, yay.
Then it'll come out. What is output? Supposes name is Karen.
Karen Sal is one like 20
and grade is, yeah, for example,
our SAL is 90,000.
Then what about this condition? Condition falls.

Then it is reaching there nest the next condition
salary greater than 75.
Now it is true. Then it'll be executing this.
Then you'll get grade B.
For example, the value is 60,000.
The salary value first condition falls
or reaching the second conditions.
That is also false. Of which the power condition.
Now 60,000 greater than another equal to 50. That is true.
Now grade is C. The C will be printed here.
For example, we have 20001st falls,
second falls, third falls.
Then finally it is reaching yes here.
What is that grade D that is printed here.
Now let's check with this data with all possible values.
We'll test it
for Karen.
Actually we have given one lack 20.
Okay, now let's more for the value.
One more person name.
Let's say salary. 90,000.
From 90,000. What I got great. P.
One more test. Let's
say 60,000.
Thesal is 60,000. Which condition is two?
First, false, second, false third is three.
So that grade will be
C, C grade.
The last test, let's think the SAL is 30,000.
Now first one is false, second one is
false, third one is false.
Then finally, what is option grade D?
Check this. I'm running six.
Got it. And one more example,

I'm taking your details like name
and your department number also.
So those name money, but department number is that.
So I want to transform the department number
into department,
department number into department.
So was here. Rules are like this.
If the department number is 11, that is department.
And if it is 12, HR department.
If it is 13, think that there is a finance department.
So remaining
that 11, 12, 13.
For all the remainings, I'm expecting the department. Ms.
Got it. So this kind of transformation I wanted to perform.
Now plan this condition poster target
is marketing department.
If DNY equals to, this is equal check level.
If it is level, what is the department?
Department is going to market it? Yes.
If the first condition falls,
I'm checking with other department.
DNY equals to 12.
Then dynamic equal to hr.
Still I say D equals to
30 equals question.
Finance not,
not all, not then
Printing and message name is from
is from Department.
We have given 13. So name is money
and department number is 13.
What about foster nation here? False.
Then it reaches second. Again, false. It reaches third.
Now this is true. What is the DM value? What finance?

What is the output? Now money is from
department finance.
So this is all, okay, let's check this.
What is output money is from department finance
because 13 value.
Let's let's test with the different value I'm giving.
14 freeze. 14 past the second.
Third, all the conditions falls. Then finally other.
The output is money is from department other.
Come on, test the department number level.
Now it's value from marketing
testing with 12
output hr.
So this function is working fine.
Okay, so in this session we have learned about how to use,
how to write the code with the initiatives and nalytics.
Okay? So in the next session
what we'll discuss is different ways to write your
conditional statements about the same task in different
how many different ways that you can code it
and which is the best performance way of the coding.
Everything will decide. So how to do that.
In the next session we'll discuss about it.
Okay, thank you very much. See you in the next session.

Video: Different Ways to Write Conditional Statements

We discussed and did, uh, lab work with uh,
NA defender sif, okay?
And multiple examples we did with uh, nas.
So in this session, NA related only,
but what are the different styles of writing the code?

Writing the conditional statement. So we'll see.
So straight away we jump into the practical part of it.
Part in the first notebook.
Same previous task I'm taking here among pre values.
Find out the biggest value using nests.
First suppose A value 10. B, value 15.
The C value for, I want
to find the biggest value among these things.
First we see wave one, which is in
If you greater than because we are comparing.
And if this condition is true among AB A is biggest
now year two B compared with the C.
If you greater than C,
this condition is also true.
Then A is the weakest, then big equals to A
suppose the second condition falls.
If second condition falls means A small B is big, C is big.
Then biggest is big, equal to C.
It's all apart. The first condition is true.
Suppose first is also false. First condition is false.
Then B is big, GA is small.
Then B2B compared with C, Y, C, Y.
In Python elevated big, greater than C.
If it's true the gig was true, B,
still else will do.
Yeah, so see now finally you are printed.
Biggest is B.
Now check the things here. So we have the values stand
15, 14.
Now among all these things, 15 is the weakest.
That is expected answer check. The output biggest is 15.
Same program in different style.
We second way
this time without writing a steps.

We try to do that if he greater than B
and he greater than later within the single condition.
We are checking that one. If this condition is true,
big equals to, yeah, I'm not writing L
and u writing a separative statement.
Now this time I'm targeting B.
If A B2B has a big,
the requirement is B should be greater than A
and also B should be greater than C, then
B equals B.
Similarly, if this time I'm targeting
C, C greater than it
and C, greater than B
equals to six.
We did not write an E, C even else.
Also, we did not write. Now how many statements?
This is separative statement.
This is separative statement
and this is separative statement.
Now finally, I'm printing. The biggest is,
is now check this.
I got same reply,
which is 51 more
modification I do in the group.
Now if this is third way, one condition,
I would like to try reduce here my technique,
I'm reviewing the lost condition before starting.
If only I'm fixing some default values.
A big chase, big equal to see.
Now observe this how this logic will work.
Try to understand.
So this is what the statement, what we have.
So what the values we have again,
B, value 10, B value 20 C value.

Suppose B value 15.
We have again, let's say
what is a C value we have taken previously.
14. We have again, let's skip 14 more.
C has 14. Now initially big value,
big equal to CV set.
That means the big value is 14.
Now check the second condition.
The festive condition, yeah, greater than me means 10.
Greater than 50
and 10 greater than 40.
Both are false here. So entire cost condition is false.
Now come to this one. That means 15 greater than 10
and 15.
Greater than 14. Both are true.
So finally, this is so latest value of the weak,
previous old value of the weak is 14.
Now latest value have the biggest 50.
Now what it means, six 50 on
that where this logic works.
Okay,
and one more way, very simple way.
I will present. We have three option either whether A,
B, C among these three, anyone is the weakest.
Okay, first time fixing anyone as a default value
as I'm fixing C as a big.
Now my condition, if yay, greater than big
then big equal two.
If CB greater than big,
then big equal two.
That's it. Only two conditions, two separative conditions.
I'm giving here. Final.
I'm printing the biggest, is
now still the same answer we got Now try

to understand how it works.
I I will explain here
to this same previous values, the value 10, B,
value 50 C, value 40 in the beginning.
What is a big, big equal to 40?
After what is this condition? A greater than big.
A greater than big means 10. Greater than 40.
Now current value of the big is 40.
Now this condition is false so
that the statement won't be executed.
Next condition. B value
15, big value.
14. Between greater than 14 condition through.
Now this part will be executed.
Now what is the latest value of the week? Big equal to D.
D means 15. So finally 15 will be correct.
Okay, now
let's check with other different values.
Example, this time let's make A as big.
Let's say A is 70, B is
35, and C is 40.
The biggest should be 70 in the beginning.
C value big value 40 because big equal to C.
You said that's why it is 40 now. Now check this condition.
A value is 70, 70.
Greater than big, big is 40 enough condition. True.
Then the statement will be executed.
Now big value big equals big value.
70. Now 70 will be correct,
but this is operative condition.
Once this, if condition is executed again,
the control goes to the next step.
Statement. This is operat.
Separate in vendor statement, directly won't,

it won't reach this statement.
So that's why. But up to now, latest value
of the big is 70, previously 40.
Now big is war in with the 40.
Now check this condition,
B value 35, latest value of the big.
So here it is 70, right?
Let us value of the week 70, this condition false.
Still the big value is 70.
Finally, what will be printed 70 will be printed.
Same thing. This time we try to make a Cs B guest try
to evaluate this condition
that say this time you value 70 B, value
90 C value one,
10, yeah, C value.
Now in the beginning, B, value one 10,
the first condition 70, greater than a value 70,
greater than one 10 box.
This won't happen. I check the second.
Second condition,
B value 90, 91, 10.
Again false. This won't happen.
So that latest, so what is the current value
of a big, which is C?
That is what will be printed. One 10 will be printed.
Understood. This is working very simple ways.
The last one. Okay.
Now among the about three ways,
which is best in this situation, we try to evaluate.
So look into this. The first way is this one.
Second way, I'm keeping here.
The third way I'm keeping here.
So first way, if you greater than B,
if you greater than C, B equals

to DY,
big equal to C.
This is four positive yel, C, L, F,
B, greater than C, then be equal, equal
to B.
Yes, be equal to C. This is fast away.
So how many conditions will we checked here in this,
for example, first condition is true.
First it will check the first condition.
This is condition number one. Check number one.
If it is true, it is checking this.
Then if the condition is true, the statement,
if condition falls, the statement will be executed.
So if the faster condition is true, I have two options.
So here, number of checks, only two checks.
For example, if first condition falls,
then directly will come to this spot.
Then this will, this will be checked
in case of condition two.
It is checking one, two conditions.
If in case of condition falls, it is checking one
and three conditions.
Okay? Always it is checking number of checks here.
Two, in this example, look into the second way.
If you greater than B
and greater than C,
then big equal to Y.
If you greater than, sorry, be greater than Y
and be greater than C,
then big equal to B.
If C, greater than Y
and C, greater than B, then B, equal
to C.
Now how many checks will happen? Condition?

Check one condition, check in this way.
I have three statements here, two here, two here, two.
Finally, six conditions are checked.
Think that in the case of first case, in the first way
in first way, I have young options.
I have three options. That's why number of checks, two,
if I have young options, number
of checks will be and minus one.
If I have 10 options, number
of checks will be 10 minus one nine in the first way.
Hmm. Whereas in second.
Second way, if I have three options, number of checks,
three into two, if I have an options, 10, two and minus one.
If I have 10 options,
10, 2, 10 minus one means nine totally,
90 checks will happen in second way,
which is best if I have more number of options.
First way is best and compared with second way.
But third way is still more simple
because what is a statement we have kept
and the thoughts are third way is equal to B or C.
Then you are targeting, you're comparing with the
if a greater than big then big equal
to if C, greater than big,
sorry, big, greater than big.
Then big equals two big.
So here, condition check one, condition,
check 2 1, 2 options.
And the first way, two options in the second way,
two options, okay?
In the second wave, two options.
So both of us, so, so wave one, wave three, both are best,
but wave three is more simplified, especially
for this kind of situation.

I can use thought because more simplified code.
So this is what like, uh,
today's discussion in this session, discussion
in the next session, we go with the
real time bank scenario with initiatives.
Okay? Let's see. Thank you very much.
We meet in the next session.

Video: Real Time Bank Scenario

Welcome back to the Python sessions.
In the last session we discussed about the different ways
to write conditional statements.
Okay? Now we see a banking example scenario.
So by using these, I'm opening a new notebook,
just connected me the wrong time.
That here, the use cases
I taking a bank accounts, for example, bank customers.
So I'm taking help of a dictionary here.
So initially bank is not having any customers.
That's why this dictionary isn't,
no, I'm giving something I code number.
Let me write in a plain text, lateral sub, substitute
that into the notepad, cell notebook cell.
Then we'll execute it.
So what account number one zero
and the amount something 10,000.
That mode, that means type of operation. He wants to pay.
For example, he wants to deposit.
So account number,
bank account number 1 0 1.
He wants to deposit 10,000. First of all, what is the rule?
The account number should be existed in the bank.

If it does not, uh, if we,
if it doesn't have the account number,
the account number to be created.
Okay, so if
that 1 0 1 account number is not available in the
dictionary, first we check in the beginning,
we don't have any account numbers in the dictionary.
If not available, we need to insert that page.
So for that pay account number as key that the amount
as value allocate.
Okay? If account number
in, in operator is
the bank, bank is a dictionary.
If we is available, we need to add that amount.
Suppose if it is not available,
if account number not in the bank, then bank
of account number equal two.
Up to now what happens? Just second.
This is simply my condition
of this.
Print the bank. Hmm? Compare the values.
1 0 1 available in the dictionary.
Not available if not available.
The given condition is true
because we said account number not in bank,
if not available available.
We are making account number as key amount as value.
Just check it. What happened here?
10001st time inserted. That means his account is so bad.
Got it. But the same thing I'm giving 1 0 2.
Same
still 1 0 1.
Let's say 20,000. Now if I run
this, what happens?

Nothing happens. What is the expected functionality?
1 0 1 has already account his balance amount is 10,000.
Once again, he's the doing deposit.
That means if the account number available in the bank
to the balance amount, the 20,000 should be added.
Now his latest balance should be
10 plus 20, 30,000.
Suppose if not available.
If you're available then else
the given condition is if not available.
If that false means already account number is already
available in the then else bank of account number
to the existing value, we are adding that amount.
What is that amount? That
20,000 is available in the variable amount.
Now check it now.
Latest balance of 1 0 1 is 30,000.
Got it. So all this operation
to be done if the operation is deposit.
Got it. That's why I want to add one more condition for
that after this.
If mode equal to deposit,
if mode is equal to deposit
there only, I'm checking if
that account number available in the bank or not.
If not available. We are inserting if available,
we are adding bank account number plus equals
to Yeah here if within
that one more if necessarily.
For example, post condition falls.
Post condition falls means what is the meaning?
That is not the deposit.
That might be withdrawal,
that might be check balance like the remaining operations.

For example, if fast condition is false, then I'm coming
to the else yell
and then immediately if yell, if you know else
and if you know that is yellow I mode equals too.
This time I'm checking fur with the drop.
If mode is withdraw once again to withdraw money,
what is the first condition?
First of all, you should have account come on, check
the account available or not.
If the account number not in bank
or if account number in bank first you are checking if
account number in bank, then you are allowed to withdraw.
For example, your record number is
1 0 1 1 0 1 already there in the bank.
Got it. Then to withdraw the money,
what is the under requirement?
You have balance of 10,000.
You want to withdraw 11,000 in sufficient funds right
now in that case, I'm checking our balance availability.
If the amount
is less than a equal to the available balance, how to get
that available balance back off.
I contact them. If they given our requested amount is less
than a equal to balance amount,
then only with the drug to be done.
So if fit the drum means previous balance 10,000,
suppose you are withdrawing 9,000, 10,000 minus 9,000.
Now latest balance should be thousand.
The piece of bank of amount, bank of account number
minus equal equals to,
okay, for example, this condition is false.
What is the meaning? If this condition false means
you don't have sufficient funds.
Then I'm printing a confirmation here.

Print
insufficient funds.
Your balance amount is I'm printing that
bank off account
up to now just check it.
The center processes were fund availability.
If fund available you are withdrawing.
So that amount will be directed from your balance
and amount if not available.
It is showing you some error message to the user points
and balance showing current balance and models.
It's all a story. If this condition is true, for example,
if that condition is false, if this condition is false,
I mean this condition, what does the meaning the account
number not available in there then?
Then it is printing invalid account number.
Let's create account some confirmation message
today, right?
So this entire work will be done
if this condition is true.
If mod is with the truck for example,
mod is this condition is also fall false.
That means mod is not deposit, mod is not with the truck.
Then mode might be balance check.
I'm giving a if mode equal to balance.
Okay? Then if more equal to balance also whether
that account number available in bank or not.
I'm checking if account number in bank.
If account number in bank
if available I need to dis display.
I need to print the balance amount, right?
Current balance,
bank of account number passing.
If I pick pass the key, you will get value.

The value is balanced amount. For example, condition falls.
Again definitely account number not available in the bank.
Then the message invalid
account number, which is the A CM.
That's it up to not in the high level.
There are three conditions.
Condition one if false, more equal with the if false,
more equal to balance.
Still, if it is false print
invalid operation.
Think that only allowed operations are three a deposit
with the drug balance set.
So for these three things we are going
to test with our application.
I'm replacing the center part,
the previous value.
Second number one, zero the amount,
let's say say 10,000 more
equal to initially deposit.
So before this sale, I'm keeping bank as empty additional
in the last every time.
Let's bring that back a protectionary. Now run it.
So first time 1 0 1
with 40,000 amount the account is created.
Got it.
Why? It is what?
First we run this sell because already 10,000 is added.
After that 20,000 we added 30,000.
Now this time 10,000 is added. That's why it is showing 40.
Now let's start it from the beginning. Now bank is empty.
Now run this first time 1 0 1 10,000.
Just to validate this condition in this moment.
1 0 1, not available.
First model, equal deposit condition rule.

Okay, then it is checking that account. 1 0 1 not in bank.
Yes condition rule because not available.
Then you are in setting. Okay,
now run it and see it.
1 0 1 10,000 is inserted.
Bring that next time the account number is 1 0 2.
I'm giving one like 20. He is trying to withdraw money.
Hmm. What about this condition? More equal deposit. False.
Then it'll come to this one more equal to
with the drug condition.
True the account number in bank
1 0 2 not available in the bank condition.
False Because of false You'll come to here.
What it'll say invalid account number,
please create account.
Check it. Now check the message.
Invalid account number please create accounts.
Okay, now he decided to create account
making this as a deposit.
Run it. Now check the condition.
First condition through 1 0 2.
Not in bank. True. So that bank
of 1 0 2 equal two, one like 20.
That means account is great.
Run this.
So double time we ran it.
That's why this became one like 21st time
after that one like 20,000 is added.
So two like 40 because we did run two times.
I want to withdraw. This is condition.
This time. 1 0 2 wants to withdraw.
Withdraw one like on now check it.
Condition falls. It'll come to here. Condition through.
Then it is coming to here.

Now account number 1 0 2 available in the bank.
Then it comes to here.
Now whether the requested amount is less than Oracle equal
to balance amount or not your check requested amount,
one lack balance amount or two like 40 we have.
So condition is true. Then from the two lack 40
from the two like 40, that one lock should be detected.
Then it is coming out.
Then it is showing you, it has to show you latest balance.
Just check this. What are the latest balance
now one LA button.
Okay, one more time. 1 0 2. Wants to withdraw dry again.
This time it is trying to withdraw one lack 50,000
but his balance is how much one lack 40,000.
Now check that condition again.
Condition, false condition through condition
through condition.
One. Lack 50,000, less than
or equals to one lack 40.
Condition. False because of false.
What you'll get inception, balance. Balance.
Your balance amount is the running balance.
It'll should now run it and see the result.
Inception funds your balance amount is one like 40
because we requested one like 50.
Okay,
no 1 0 3 wants
to check his balance amount.
Forget about that amount.
1 0 3 wants to check his balance,
but in fact 1 0 3 account number is not there.
So where condition will be failed, this condition falls.
This condition falls. Now this condition true
again this condition 1 0 3 in bank, not available.

So that falls. It'll reach this point.
The message will be invalid. Account number.
Okay, invalid account number. We did not run it. Not run it.
Invalid. Account number 1 0 3.
Got it It now 1 0 3 decided
to create account with the 10,000
is trying to deposit
condition true condition true.
So that account number will be created.
Check again. Now third account holder will see 1 0 3.
1 0 3 amount deposit,
invalid operat.
Check the mode again.
Account number 1 0 3 amount 10,000.
Mode deposit if mode equal to deposit condition through
previously account number 1 0 1 not in the bank.
That is also true so that it has to be displayed. Why?
It is saying that invalid operation
somewhere, the spell mistake we have
invalid operation is taken.
Look into the L spot now.
First time one zero to create it.
Once again, retest all the operations.
1 0 1 first time. So deposit to be done.
Now 1 0 2 wants to withdraw either 5,000.
Now check the condition once second please.
False mode with the drop.
I want to withdraw mode to be with the drive
with drop this condition.
True account number available. Yes, true.
5,000 less Oracle to 10,000. True.
So that balance should be detected.
5,000 will be detected from 10,000.
So the latest balance of 1 0 2 should be 5,000.

Perfect. Similarly, 1 0 3
wants to check the balance.
Balance. It should say as invalid.
Uh, like uh, invalid account number
because 1 0 3 not available in the previous yes.
Invalid account number 1 0 3.
Think that now 1 0 3 is going to deposit.
Now deposit a deposit of 50,000.
Check the operation. 1 0 3 is created.
Now 1 0 3 wants to get the drop 60,000
now inception balance to be there.
Check it. So we did not change
that mode.
That's why another 50 60,000 is added to 50,000.
Now 1 0 3 latest balance, one lack 10,000.
Now he wants to withdraw with the draw.
Let's say six lacks,
but in fact he has only one lack 10,000.
Now check the output
in tion funds or balance amount is one lack 10,000.
Okay, suppose
1 0 3 wants to do transfer,
but we did not write the logic for this.
We have it in the logic only for deposit
with the draw and check balance.
Now the first condition is false.
Second with the drug condition falls, third one,
balance condition also falls.
What is the final option then? Invalid operation.
We got invalid operation in this way.
You can use these nest steps in our real
time regular applications.
Okay, so in the next session we'll discuss about
10 array operators.

So same if condition, you can write in another style.
That is ARI operators. Okay? Okay, thank you very much.
Meet you in the next session.

Video: Ternary IF Statements

With the initiatives in this session we are going
to discuss about tari if statements.
Let's do the practical step of it.
So tari format, uh, of if statement is like this.
First approval, true expression,
then if condition,
yes.
False expression
after hearing in this statement.
The true part or true expression in the left side
and the false expression is after health.
Okay, just let me keep it into that comment.
A simple example we see taking general value,
which is the, now my true expression, I,
I want to written mail.
The condition is yeah. Yes.
I want to written return other
here the value is young,
then it is written true value, which is male.
If is not am it is reding false.
For example, if I give some,
so it is giving the condition is false, so that's why
female, okay, so I want to handle that kind of situation.
So what if gender values,
yeah, I want male.
It is, yeah, female other than I'm men,
like other, other characters out there.

I want in this way. I want to transfer.
Now plan the condition. I want male.
If gender equals to, yeah.
Yes, you need to check one more condition.
In the case of ball, I need to check with other condition.
Again, I want female if gender is equal.
Yeah. Still else I want invalid.
So here I'm, this is my statement.
Previously I'm giving something gender, young,
not check this exactly.
I got married. Suppose I'm giving,
I got female other than I'm in family.
Let's say X, it is inval.
That's a y. Inval.
For example, capital. I'm giving, I want
to handle this house, but here
I'm checking with small letter.
So in this case, what even whether it is upper case letter,
lower case letter, transform into,
then check it again Here also dot lower.
Then you get mail.
If I pass small, small, first capital FI got failure.
Okay? So how this condition is working
is checking this condition, this condition falls.
Then it is coming into else
and the else again, there is one more condition.
Okay? Now check this condition. This condition is true.
Now because of truth, it returns
your first condition falls, second condition falls.
Finally, it returns inval in this way. I have three options.
I can plan like this. Let's break it down
to the multiline.
What happens a second?
Now please like this. Try to say backslash

here, backslash.
Now we can do the same thing.
So backslash is for like a two split state window.
Got it. Left side through value, right side falls value.
This is the format. Based on this example,
you try to think about it.
I want to transform one thing.
I'm taking department number,
department number two, department name.
I want to transfer. Suppose 11.
11 is marketing
and always
HR 13.
It's remaining other than 11, 12.
But remaining all the things I want this as other.
Now using if statement, I wanted faster target
marketing, I want to marketing.
If department number equals two level,
yes, one more condition to make it easy.
Come to the next slide by saying, okay,
previously department is not there.
Let's skip some department for our testing department.
Okay? If it is false, now target hr.
What is the condition for this?
If department number is equal to 12, yes.
Okay, if it is not 12,
Again, I'm checking for one more thing.
Come to the next, next finance.
If department number is equal to
then it returns for finance still not false.
That means still condition. False. Then this is the fault.
Okay, try to do this now
for the 11 I got marked it.
So how this condition is true left side true values there

that stated understood.

For example,

I'm giving 12.

What about the first condition? Falls then come to the L.

In the ls there is one more condition.

Department number equal to 12 condition is two.

That's why I left outside. I got H.

Next 13 I got finance.

So first of false, second false.

Then else else the finance department number equal to 13.

The condition is two. I got one.

Suppose I'm giving 50, it's not 11, 12, 30.

Definitely 1, 2, 3.

All the condition spots definitely will get other.

So this is another step.

One more cost per I want

to transform salary into salary grades,

for example, greater than articles to one.

Like I want A grade A

greater than to 70 5K

and less than a hundred K.

That means one like then grade B greater than articles

to 50 K and less than 70 5K.

Then grade should be C.

If all the values which are less than 50 K

than grade is using tari.

If statement, I want this now faster target.

Yeah, yeah, agree. What is the condition for it?

First take some salary.

Salary put, let's say salary is one like 20,000. Come on.

Target the A grade, what is the condition for it?

If salary greater than equals two, then yeah,

yes, come to the next level.

Now this time target B grade.

So for this, if salary greater than article 2 75
B Ys target C grid
IT salary than article equal two 50.
Then C Ys and fix as C grid.
This is a 10 hour apparatus.
Now check this for one LA 20. The first condition is two.
That's why it written a grid. Examples is 90,000.
Then first of all, second is true, then B grade,
then I'm giving something 60.
First of all, second condition falls auto condition is true.
So that you'll get C grade.
I'm giving 41, 2, 3.
All the conditions falls. So final option, take grade.
Okay, so the center thing thing,
you can direct them into one variable card.
Great.
Upgrade. Your salary is
salary with
grade, just check the output.
No, your salary is 40,000.
With grade D, I'm making it as one like 30.
So this is the style of 10 operators.
Okay, so in last session we have seen
last three sessions S and
after the different ways to write a conditional statement.
This is also one more way
and real time bank example scenario we have in in this
session we have discussed about scenario statements.
In the next statements already, we have used in operator
and not in an in operator,
but with help of inland,
not in one more scenario, one more session.
We'll, in the next session we'll see about each statements
with the in and not in operators.

So we have a name for them for the in knotted,
what these operators are called membership operators.
Okay, with the membership operators we see nests,
the statements.
Thank you very much. Let's meet in the next session.
Bye-bye.

Video: IF Statement WITH IN & NOT IN

About statement with practical.
So in this session we are going
to talk about membership operators
and let's include these membership operators.
In the e statement. What are the two membership
operators in?
And Martin, okay, let's take,
I'm taking to the updates for here.
These are my friends list
money.
And will I think that these are the Facebook friends
or some, any other social media friends?
And I'm taking your friends,
your friends, like
who are the common friends in this?
Above this list?
My friends count sport.
Your friends count is far, but there are common friends.
First I want to find out common friends.
For example, let's text. Susan is actually my friend.
Whether he's available to know friends list
or not, I want to check your friends.
So TRO is not available in your friends.
Same thing I'm checking with not Ravi.

Ravi is my friend and also he is your friend.
So that Ravi is our common friend.
This clue is enough to build common friends.
That's a common friend initially until time.
But I look through my list for Yu
or MF my friend in my friend.
So each trend is coming into, so here's one mistake list
my practice.
Each list, I'm checking in the your if EMF
in your
friends in the ation.
What is the value of? Let's make it as you have.
You face a short term
in ation.
What is available in our friends list? No.
If not, I'm not doing anything. Suppose second iteration.
Money, money also not with you.
Condition falls, extra ation
is available in your friends list.
Then I'm appending that
common up.
Up do, yeah. Now check the final output print.
So Ravi, we no other the common friends between you and me.
Similarly only my friends ever.
That means those are exclusively my friends, but not all.
Eventually I'm keeping empty list.
This time I'm using negation in the place of in operator.
I will use, not in operator, just check it
for my friends.
If Yf not in, if you have not your friends,
It is true.
I'm opening to only my
friends don't up and up.
Yes. Now check it.

Only my friends
check this here.
Virgin and money are my friends, but not your friends.
On the same style. I want to build
only your friends,
only your friends list.
I want to find
initially I'm the list.
Now try send the loop to your friends
or your friends.
Check that to your friend is not there in my friends list.
If you have not,
not in my friends, then add
to only your friends do up
and off, up and off.
What Jeff,
the list is ready.
Check the values of only your friends.
So Dvia, Karen LaMi, your friends.
Now I'm printing all the things at one place on spot
Common Friends that is in common.
Friends, your friends.
And I can only your friends.
My friends.
Only my friends.
So take this list between you and my conference.
You have your own exclusive friends.
That is the Blackman and my friends Astrogen and money.
Okay? So in this way you can take help
of membership operators, like, uh, in and
nor you can use them into give statements
and a statements, any conditional statements.
Okay? So this is what our discussion
in the next session onwards, we go with the for loops.

That means looping statements up to now like, uh,
controlling statements have sent from the next session on
what we, we'll start discussing about looping statements.
The looping statements are like loop while in Python.
We have two loops. Let's meet in the next session.
Thank you very much.