



CoGrammar

Conditional Statements



**SKILLS
FOR LIFE**

SKILLS BOOTCAMPS



Department
for Education

Lecture Objectives

1. **Define operators and explain their usage in programming.**
2. **Distinguish Between different types of operators.**
3. **Demonstrate and apply basic operator usage in your programs.**

Software Engineering Lecture Housekeeping

- The use of disrespectful language is prohibited in the questions, this is a supportive, learning environment for all - please engage accordingly.
(FBV: Mutual Respect.)
- No question is daft or silly - **ask them!**
- There are **Q&A sessions** midway and at the end of the session, should you wish to ask any follow-up questions. Moderators are going to be answering questions as the session progresses as well.
- If you have any questions outside of this lecture, or that are not answered during this lecture, please do submit these for upcoming Open Classes.
You can submit these questions here: [Open Class Questions](#)

Software Engineering Lecture Housekeeping cont.

- For all **non-academic questions**, please submit a query:
www.hyperiondev.com/support
- Report a **safeguarding** incident:
www.hyperiondev.com/safeguardreporting
- We would love your **feedback** on lectures: [Feedback on Lectures](#)



Poll:

Assessment



CoGrammar

Recap on Python and Variables

Booleans

- ★ Booleans can only be stored as one of two things: `True` or `False`.
- ★ Mainly used for conditional checks.
- ★ Booleans should be declared in Python with capitals. Using lowercase for booleans will return an error in Python.

Comparison Operators

OPERATOR	OPERATION	EXAMPLE
<code>==</code> Equal to	True if x has the same value as y	<code>x == y # True</code>
<code>!=</code> Not equal to	True if x does NOT have the same value as y	<code>x != y # False</code>
<code>>=</code> greater than or equal to	True if x is greater than or equal to y	<code>x >= y # True</code>
<code><=</code> Less than or equal to	True if x is less than or equal to y	<code>x <= y # True</code>

Logical Operators

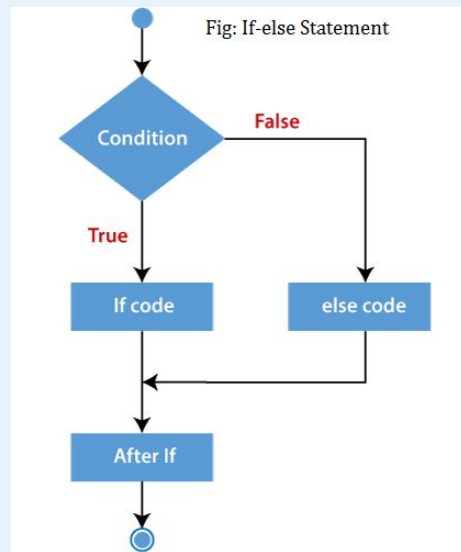
OPERATOR	OPERATION	EXAMPLE
and	True if both x AND y are true (logical conjunction)	If x and y : print(z)
or	True if either x OR y are true (logical disjunction)	If x or y print(z)
not	True if the opposite of x is true (logical negation)	If not x print(y)

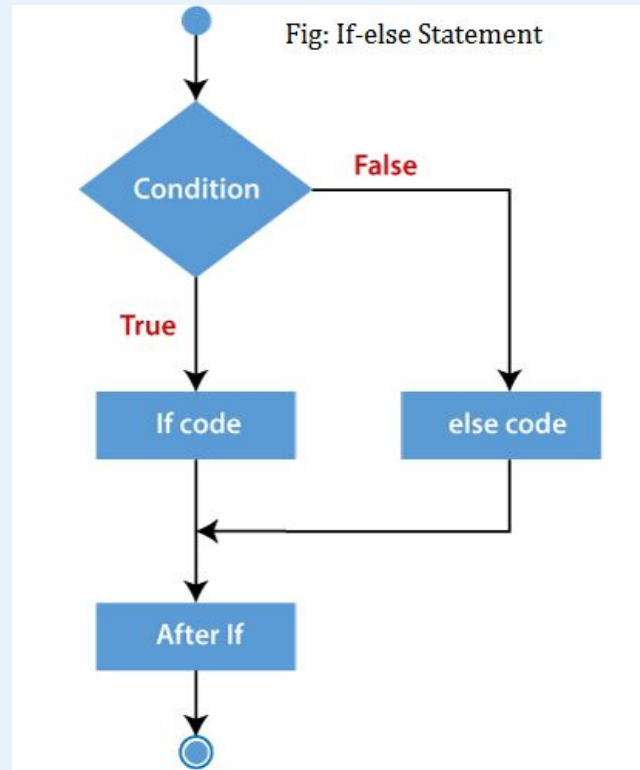
Elif Statements

- ★ What if there is a situation where we could have multiple statements that are True?
- ★ This is where elif comes into play: Else if → elif
- ★ Elif statements are mainly used to handle the case when multiple True statements are present.
- ★ Note that you can have multiple elif statements in an if-else block.

Control Structures

- ★ Control structures are code that will analyse variables and then choose a direction to follow based on the input provided.
- ★ Think of it as a form of branching: depending on the provided input, your program will have one of x branches to follow.
- ★ e.g. "If I finish my work early, I will go to bed. Else, I will have to work through the night".







Question:



**How are comparison operators
different from logical operators?**



Few Things To Note

- ★ There is **no limit** to the number of elif statements one could have in an if-else block.
- ★ Only one **final** else statement is allowed.
- ★ Each condition is checked in order.
- ★ If one condition is **True**, that branch executes, and the statement ends.
- ★ Even if there are multiple True conditions, only the first **True** branch will execute.

and Operator

- ★ Returns as True when both conditions specified are met.
- ★ Example:

```
if grade > 50 and grade > 75:  
    print("conjunction")  
    print("Both Conditions have been met :)")
```

or Operator

- ★ Returns True if either of the specified conditions are met.
- ★ Example:

```
elif grade > 50 or grade > 75:  
    print("disjunction")  
    print("At least one of these conditions have been met.")
```


not Operator

- ★ Changes the condition from True to False and vice versa.
- ★ Example:

```
if not grade > 50:  
    print("Negation")
```

Best Practices





Challenge:



What will this combination of operators return?





Poll:

Assessment



Wrapping Up

Control Structures

Programming Constructs that manage the flow of execution in a programme

If-else statements

Allows the execution of a block of code based on specified condition

Progression Criteria

✓ **Criterion 1: Initial Requirements**

- Complete 15 hours of Guided Learning Hours and the first four tasks within two weeks.

✓ **Criterion 2: Mid-Course Progress**

- Software Engineering: Finish 14 tasks by week 8.
- Data Science: Finish 13 tasks by week 8.

✓ **Criterion 3: Post-Course Progress**

- Complete all mandatory tasks by 24th March 2024.
- Record an Invitation to Interview within 4 weeks of course completion, or by 30th March 2024.
- Achieve 112 GLH by 24th March 2024.

✓ **Criterion 4: Employability**

- Record a Final Job Outcome within 12 weeks of graduation, or by 23rd September 2024.

CoGrammar

Questions around String and Variables



CoGrammar

Thank you for joining

