**What are negative indexes and why are they used?**

Negative indexes are a feature in some programming languages (notably Python) that allow you to access elements from the end of a sequence (like a list, string, or tuple) rather than from the beginning.

***How They Work:***

In Python, indexing starts at 0. So, for a list like this:

my\_list = ['a', 'b', 'c', 'd']

* my\_list[0] returns 'a'
* my\_list[1] returns 'b'

With **negative indexing**, you count from the end:

* my\_list[-1] returns 'd' (the last element)
* my\_list[-2] returns 'c'
* my\_list[-4] returns 'a'

***Why Negative Indexes Are Used:***

1. **Convenience**: They provide a quick and readable way to access elements at the end of a sequence without needing to calculate the length.

last\_item = my\_list[-1]

1. **Slicing from the end**: Negative indexes are especially useful when slicing.

last\_two = my\_list[-2:] # ['c', 'd']

1. **Avoiding len() calls**: You don’t need to write.

my\_list[len(my\_list) - 1] to get the last item.

***Summary:***

Negative indexes:

* Start at -1 for the last element
* Increase toward -len(sequence) for the first element
* Make code more concise and easier to read when dealing with the tail of a sequence

They're a powerful feature that simplifies many common operations on sequences.