

00:18 Issues with arrays
that linked list solves
05:54 Doubly linked list
06:37 Big O analysis (array
vs linked list)
08:02 Python
implementation
26:00 Exercise



```
stock_prices = [298,305,320,301,292]
```

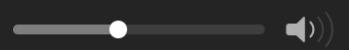
stock_prices

0x00500	298
0x00504	305
0x00508	320
0x0050A	301
0x0050F	292

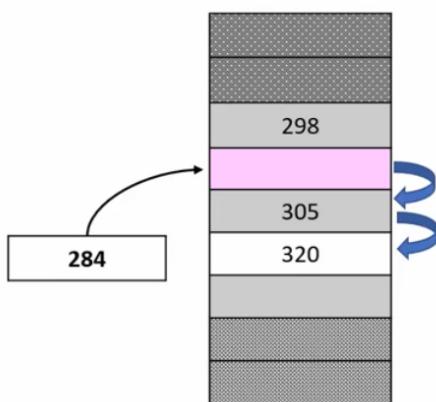
```
stock_prices.insert(1, 284)
```

298
284
305
320
301
292

Array insertion = O(n)



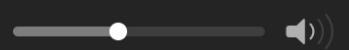
```
stock_prices = []  
  
stock_prices.append(298)  
stock_prices.append(305)  
stock_prices.append(320)  
  
stock_prices.insert(1,284)
```



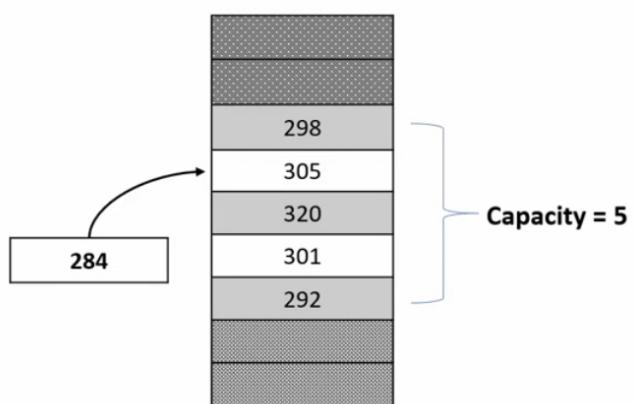
01:55

-26:21





```
stock_prices = []  
  
stock_prices.append(298)  
stock_prices.append(305)  
stock_prices.append(320)  
stock_prices.append(301)  
stock_prices.append(292)  
  
stock_prices.insert(1,284)
```



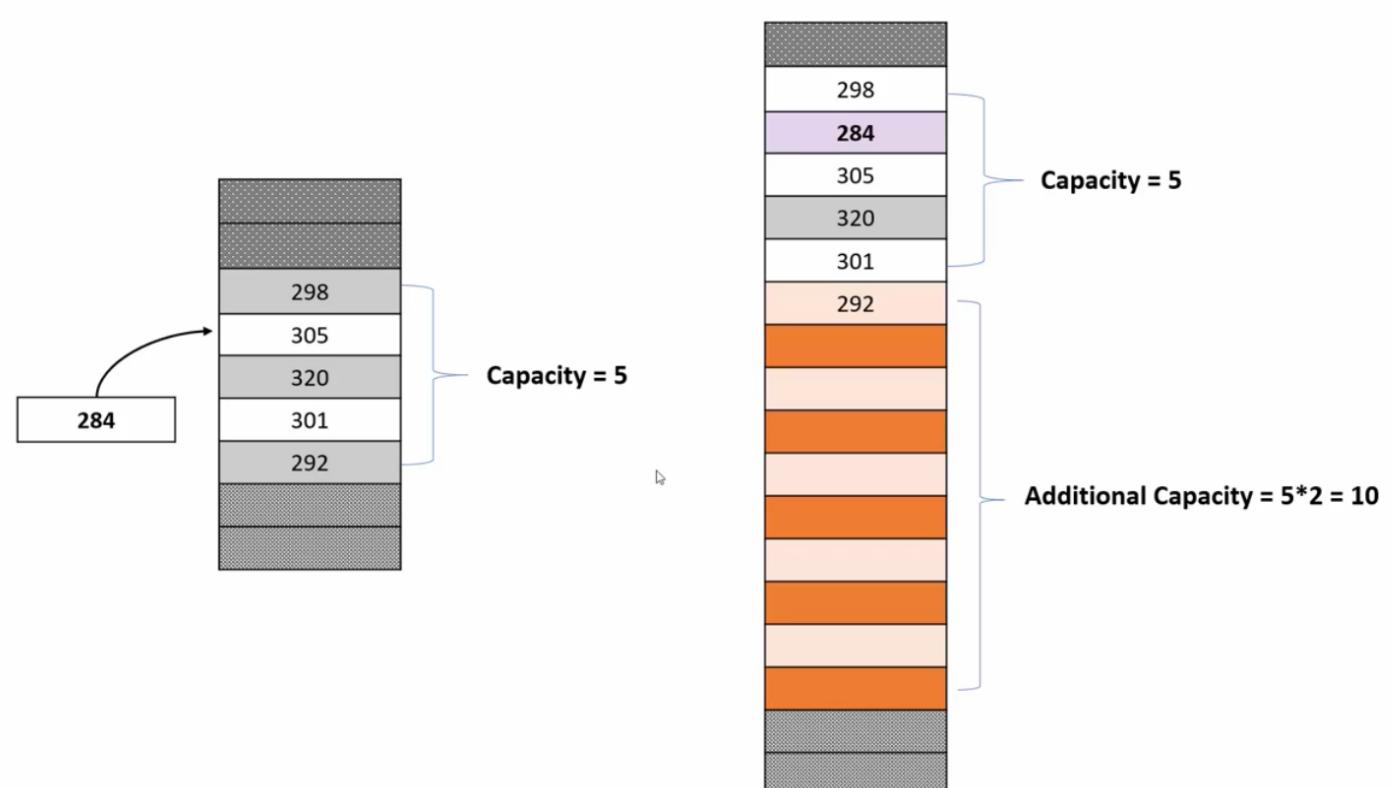
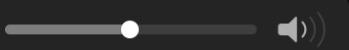
Assumes that your array is full,
but you wanna insert some value into it.



02:06

-26:10





03:17

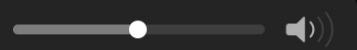


- 24:59

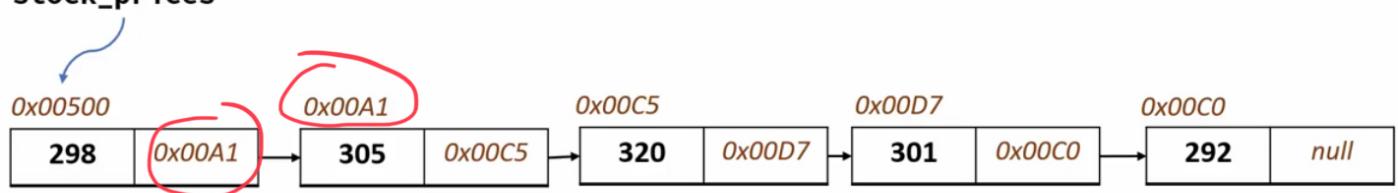


11:46 Sat Sep 11

✈️ WiFi VPN ⚡ 65% 🔋



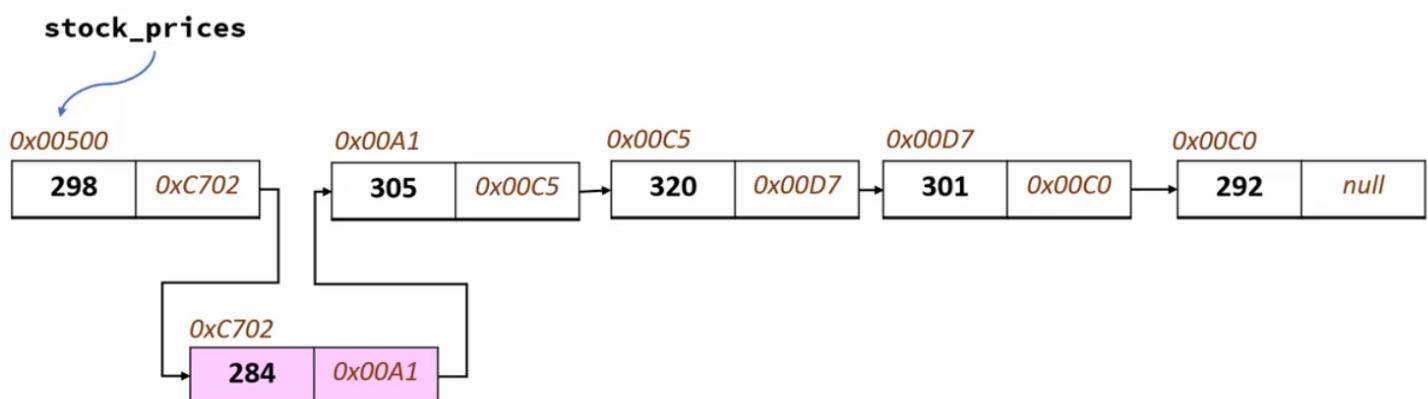
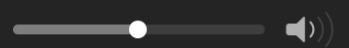
stock_prices



04:11

- 24:05

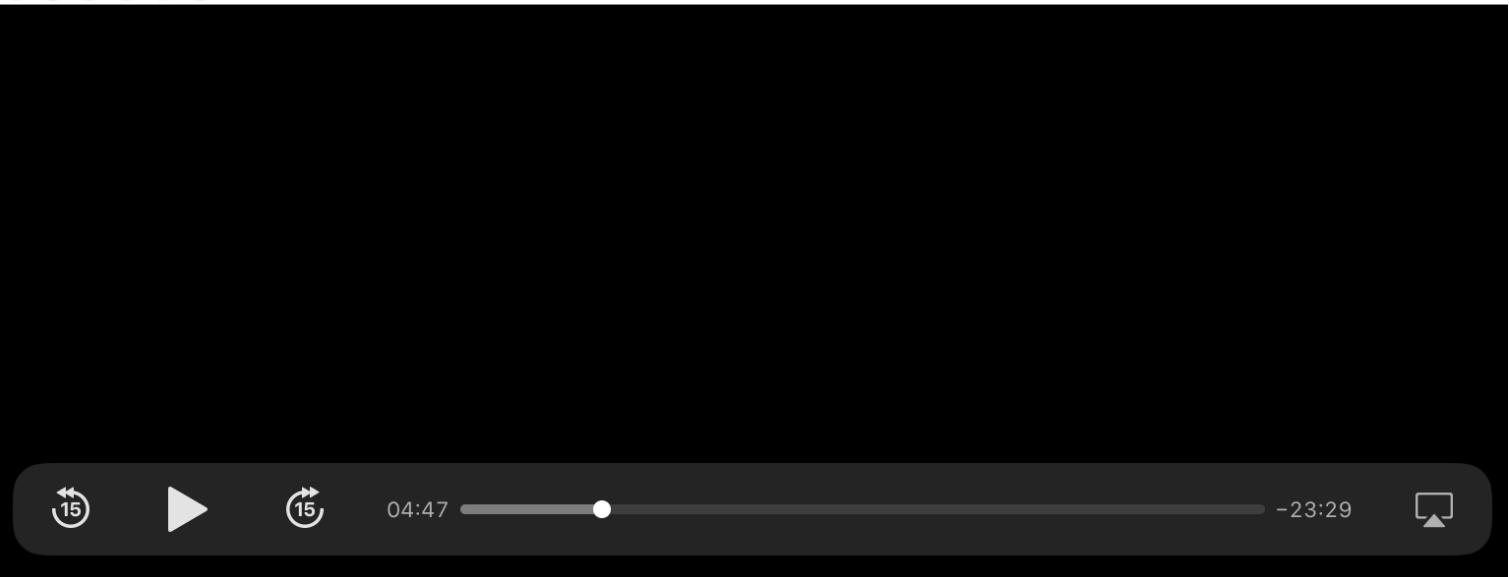


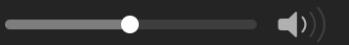


Insert Element at beginning = O(1)

Delete Element at beginning = O(1)

Insert/Delete Element at the end = O(n)





Linked list has **two main benefits** over an array,

1. You don't need to pre-allocate space
2. Insertion is easier



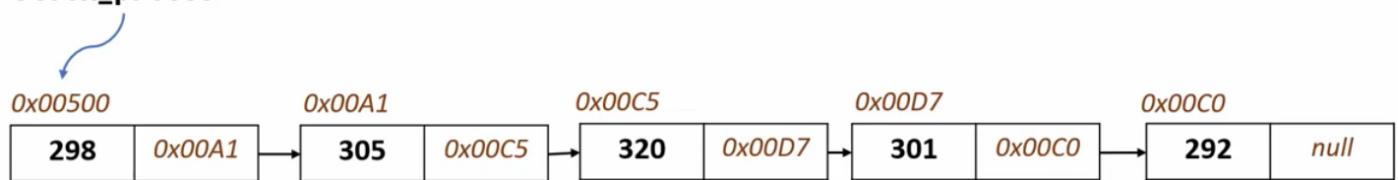
05:20

- 22:56



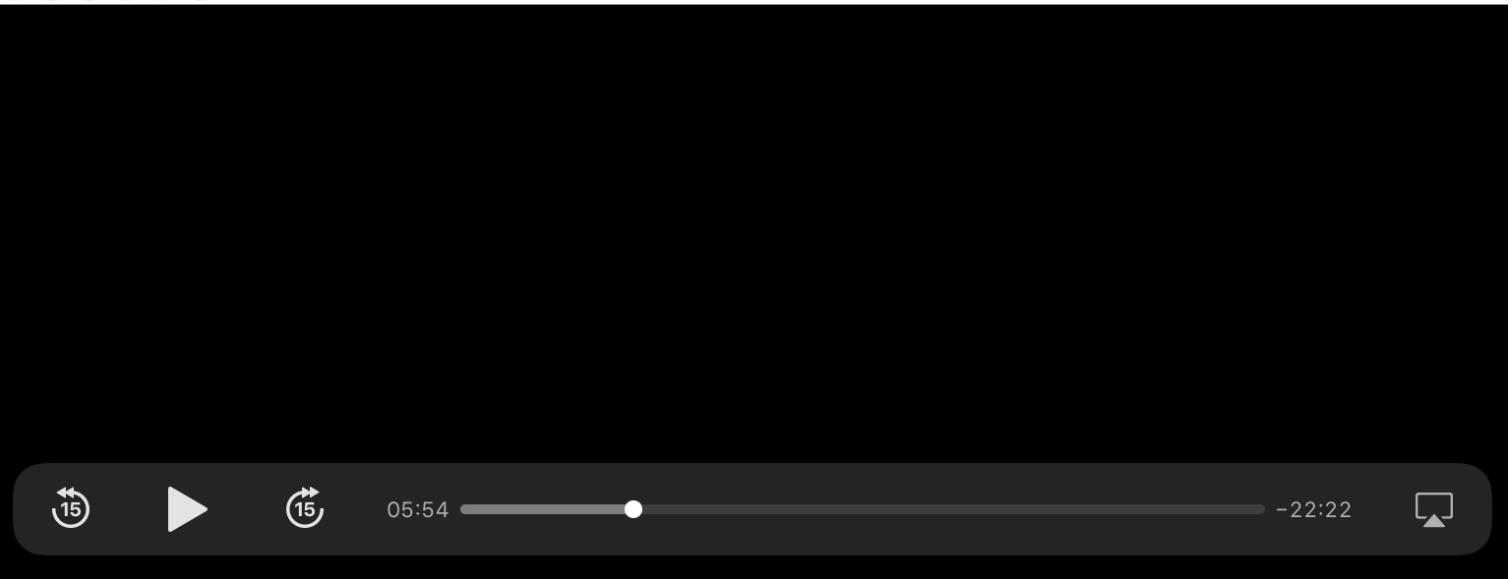


stock_prices



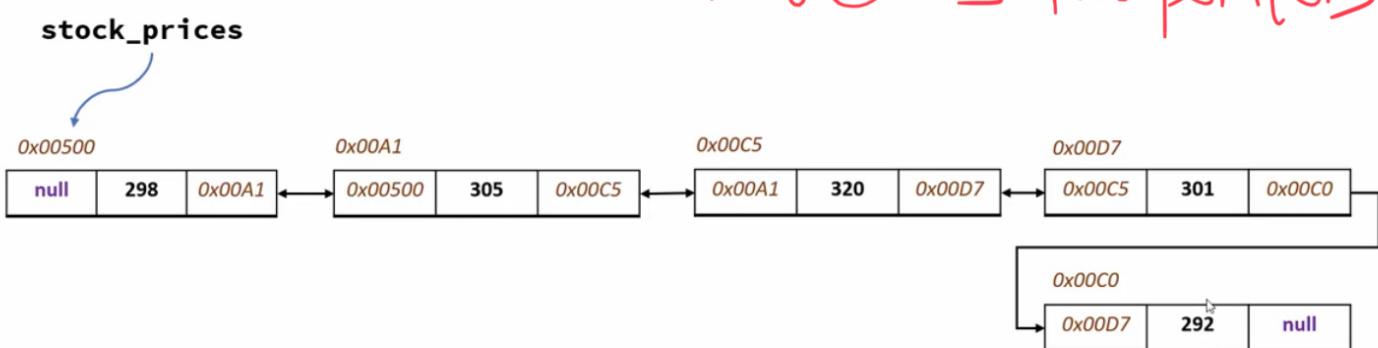
Linked List Traversal = O(n)

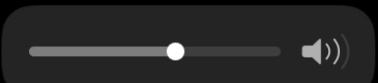
Accessing Element By value = O(n)



Double Linked List

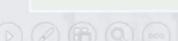
The node has two pointers





you need to find
your value
one by one

	Array	Linked List
Indexing	$O(1)$	$O(n)$
Insert/Delete Element At Start	$O(n)$ you need to copy other values and swap them	$O(1)$ / I don't need to copy
Insert/Delete Element At End	$O(1)$ - amortized find, insert/delete, don't need to copy & swap	$O(n)$ need to find the location $O(n)$ one by one
Insert Element in Middle	$O(n)$ need copy & swap other values	need to find the location one by one



achieve those methods of linkedlist:

(1) print (name)

(2) get_length

(3). insert_at_beginning

(7) insert - varnames

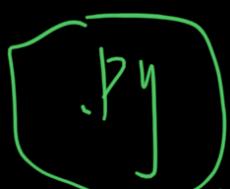
(4). insert_at_end

(insert in list

(5) insert_at

of varnames)

(a specific location)



can be

(6) remove_at

found on github

(a specific location)

