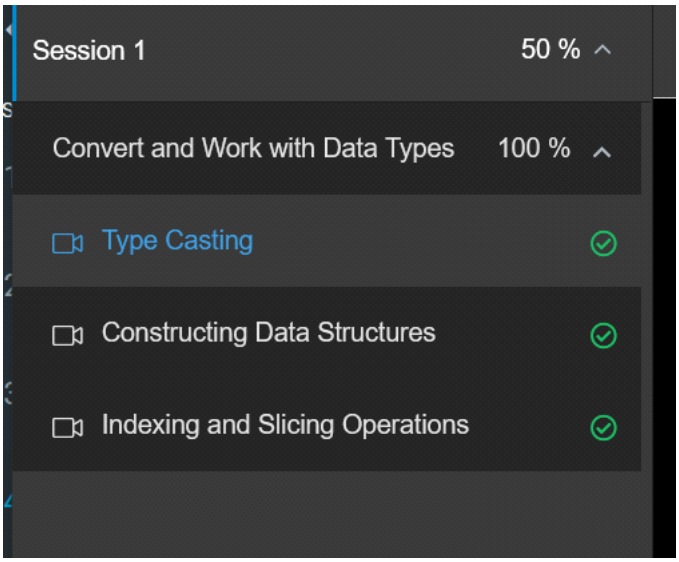
08 April 2021 18:30



08 April 2021 18:34

1 Type Casting

str

```
>>> price = 3.95
>>> widgets = 5

>>> print("The price of the widgets is ", price)
The price of the widgets is  3.95

>>> print("We have " + widgets + "in stock")
Traceback (most recent call last):
  File "<pyshell#3>", line 1, in <module>
    print("We have " + widgets + "in stock")
TypeError: can only concatenate str (not "int") to str

>>> print("We have " + str(widgets) + " widgets in stock")
We have 5 widgets in stock

>>> print(f"The price ({price}) as an integer is {int(price)}")
The price (3.95) as an integer is 3

>>> print(f"The number of widgets {widgets} as a float is {float(widgets)}")
The number of widgets 5 as a float is 5.0
```

Note this !

08 April 2021 18:48

(2) Constructing data structures

Fancy way of saying "building lists"

Arrays in other languages

[ ]

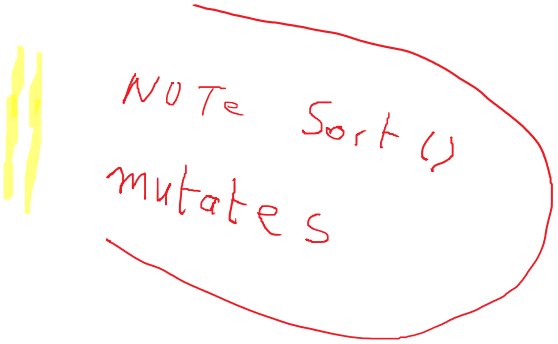
Can mix data types within list, but Not recommended

```
>>> regions =["north", "south", "east", "west"]
>>> sales=[30000, 20000, 40000, 350000]

>>> employees=["Alice", "Vera", "Flo", "Mel"]

>>> for employee in employees:
        print(f"{employee} is an employee!")


Alice is an employee!
Vera is an employee!
Flo is an employee!
Mel is an employee!

>>> employees.append("Thomas")
>>> employees.remove("Flo")
>>> for e in employees:
        print(e)

Alice
Vera
Mel
Thomas
```

.append()

.remove()

```
>>> employees.sort()
>>> for e in employees:
        print(e)

Alice
Mel
Thomas
Vera
```



08 April 2021 19:45

Indexing and Slicing

1) Python zero-based

```
>>> regions =["North", "South", "East", "West"]
>>> sales=[30000, 20000,40000,35000]
>>> employees=["Alice", "Vera", "Flo", "Mel"]

>>> print("Region", regions[0], " Sales: ", sales[0])
Region North  Sales:  30000

>>> print(f"The last region is {regions[-1]}")
The last region is West

>>> employees[2]="Thomas"
>>> for e in employees:
        print(e)


Alice
Vera
Thomas
Mel
```

|| slicing