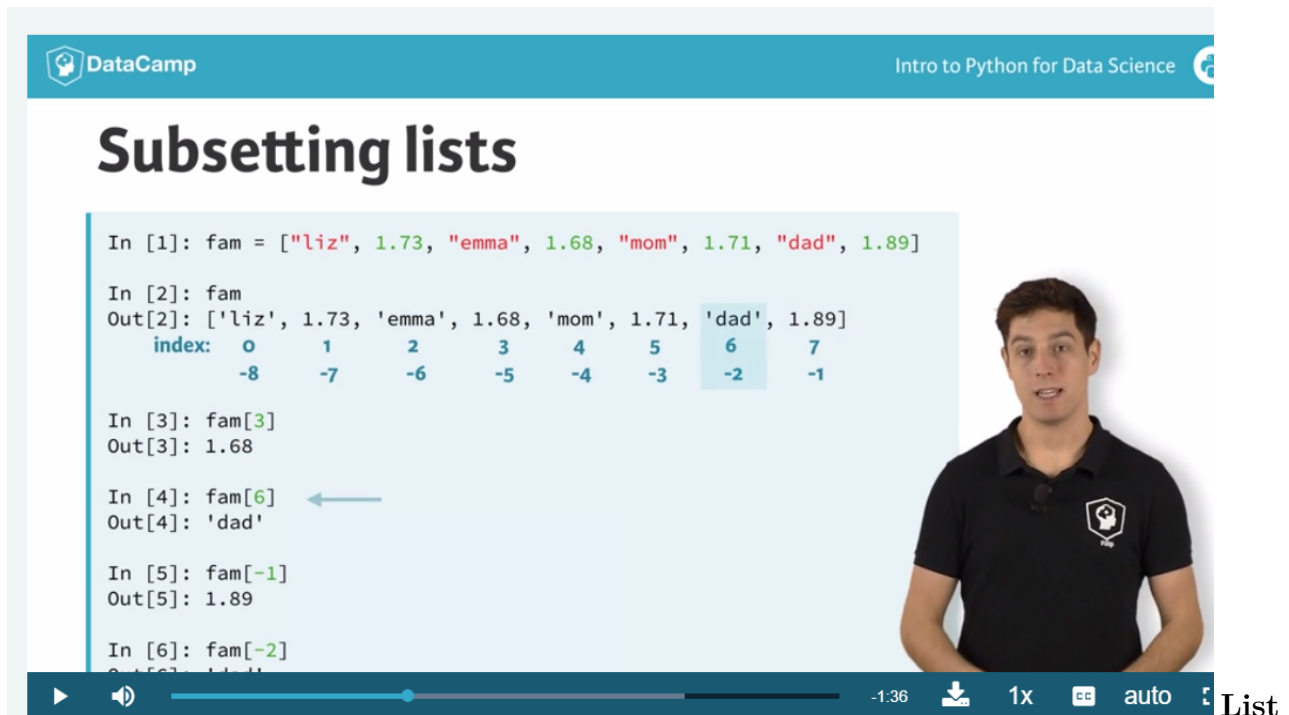


## Some Python code

### 1 Subsetting List



**DataCamp** Intro to Python for Data Science

## Subsetting lists

```
In [1]: fam = ["liz", 1.73, "emma", 1.68, "mom", 1.71, "dad", 1.89]

In [2]: fam
Out[2]: ['liz', 1.73, 'emma', 1.68, 'mom', 1.71, 'dad', 1.89]
index:  0      1      2      3      4      5      6      7
       -8     -7     -6     -5     -4     -3     -2     -1

In [3]: fam[3]
Out[3]: 1.68

In [4]: fam[6]
Out[4]: 'dad'

In [5]: fam[-1]
Out[5]: 1.89

In [6]: fam[-2]
Out[6]: 'dad'
```

**List**

#### Slicing :

```
list ['liz', 1.73, 'emma', 1.68, 'mom', 1.71, 'dad', 1.89]
list [3:5]
//Terminate:
[1.68, 'mom']
```

### 2 Add to list

```
# Create the areas list and make some changes
areas = ["hallway", 11.25, "kitchen", 18.0, "chill zone", 20.0,
        "bedroom", 10.75, "bathroom", 10.50]

# Add poolhouse data to areas, new list is areas_1
areas_1 = areas + ["poolhouse", 24.5]
print(areas_1)
```

```
# Add garage data to areas_1, new list is areas_2
areas_2 = areas_1 + ["garage", 15.45]
print(areas_2)
```

### 3 Delete list element

```
x = ["a", "b", "c", "d"]
del(x[1])
print(x)
```

Then we have: ['b','c','d']

### 4 Inner workings on lists

```
# Create list areas
areas = [11.25, 18.0, 20.0, 10.75, 9.50]
```

```
# Create areas_copy
areas_copy = areas
```

```
# Change areas_copy
areas_copy[0] = 5.0
```

```
# Print areas
print(areas)
```

### 5 Print type of Function

```
result = type(3.0)
# Assign type of function 3.0 for variable 'result'
print(result)
# Print 'result'
```

## 6 Length of type

```
result = 3.0 + 2.5
# Assign the value for result
print(len(result))
# Print the length of 'result'
```

## 7 Show Help in Python

```
# Show sth about the code you need
help(function,...)
```

## 8 Sorted

```
# sorted() take three arguments(iterable, key, reverse)
# If you don't specify anything in sorted() then key=None, reverse=True
# (decending order)
sorted(iterable, [key], [reverse=])
```

## 9 Upper Method

```
# Upper String [str.upper]
room = 'poolhouses'
room_up = room.upper()
# Print room_up
print(room_up)
```

## 10 Count Method

```
# Count the string str.count([substr or str], start=, end=len(str))
print(room.count('o'))
```

## 11 List Method

```
# Index Method
areas = [11.25, 18.0, 20.0, 10.75, 9.50]
```

```

# Print Index of 20.0
print(areas.index(12.0))
# Print Count of 14.5
print(areas.count(14.5))
# Add an element to the list it is called on
print(areas.append(23.00))
# Remove an element to the list that matches the input
print(areas.remove(10.75))
#Reverse the order of the elements in the list it is called on.
print(areas.reverse())

```

## 12 Numpy

```

# Create list baseball
baseball = [180, 215, 210, 210, 188, 176, 209, 200]

# Import the numpy package as np
import numpy as np

# Create a numpy array from baseball: np_baseball
np_baseball = np.array(baseball)

# Print out type of np_baseball
print(type(np_baseball))
# Print out dtype of np_baseball
print(type(np_baseball.dtype))

```

## 13 If-else in Python

```

if [expression]:
    statement(s)
else:
    statement(s)

```

### 14 Numpy Array

```
# To use NumPy array we import NumPy package
import numpy as np
# create a list
height = [12.45,78.90,34.56,56.78]
# Use np.array(...) to make NumPy array
np_height = np.array(height)
# We can add function in array
np_height_m = np_height * 0.05
print(np_height_m)
```