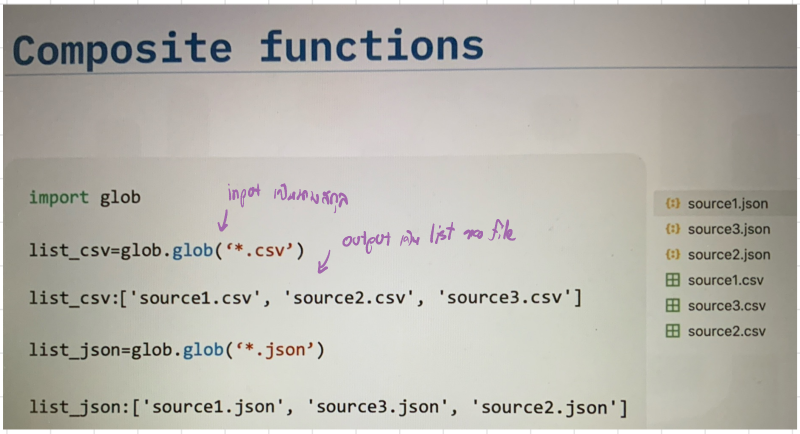
Extract:

glob

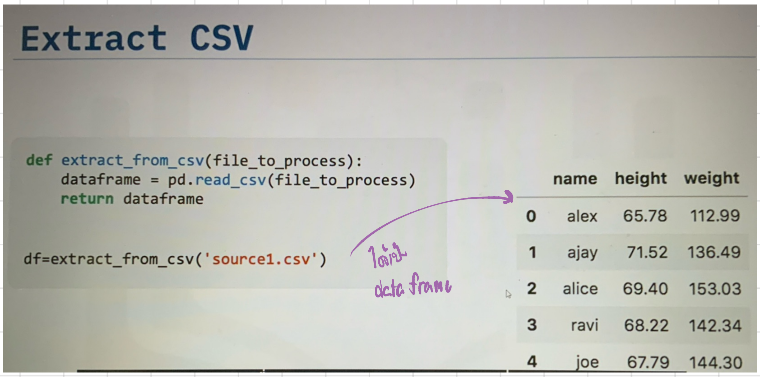


Composite functions

```
import glob

list_csv=glob.glob('*.csv')

list_csv:['source1.csv', 'source2.csv', 'source3.csv']

list_json=glob.glob('*.json')

list_json:['source1.json', 'source3.json', 'source2.json']
```

input เป็นนามสกุล

output เป็น list ของ file

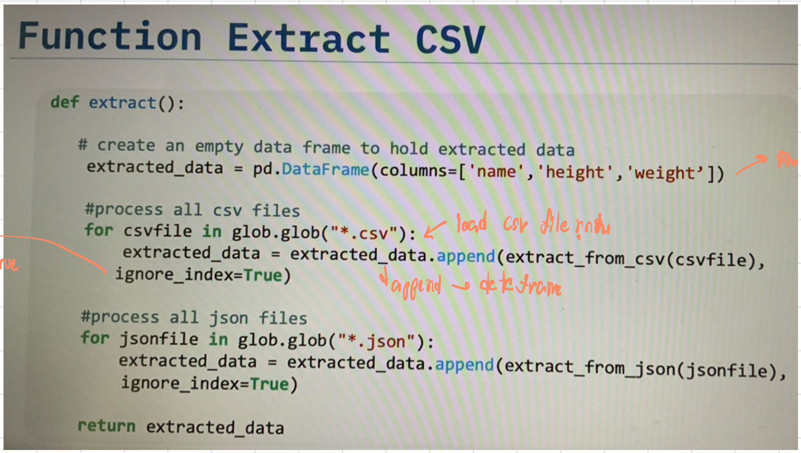


Extract CSV

```
def extract_from_csv(file_to_process):
    dataframe = pd.read_csv(file_to_process)
    return dataframe

df=extract_from_csv('source1.csv')
```

ได้เป็น data frame

	name	height	weight
0	alex	65.78	112.99
1	ajay	71.52	136.49
2	alice	69.40	153.03
3	ravi	68.22	142.34
4	joe	67.79	144.30



Function Extract CSV

```
def extract():

    # create an empty data frame to hold extracted data
    extracted_data = pd.DataFrame(columns=['name','height','weight'])

    #process all csv files
    for csvfile in glob.glob("*.csv"):
        extracted_data = extracted_data.append(extract_from_csv(csvfile),
        ignore_index=True)

    #process all json files
    for jsonfile in glob.glob("*.json"):
        extracted_data = extracted_data.append(extract_from_json(jsonfile),
        ignore_index=True)

    return extracted_data
```

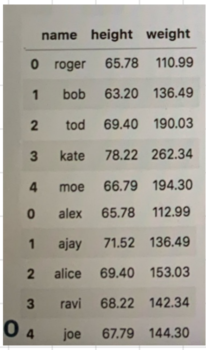
ทำ line นี้สร้าง Header

ให้ dataframe

load csv file ทุกตัว

append → data frame

เป็น True

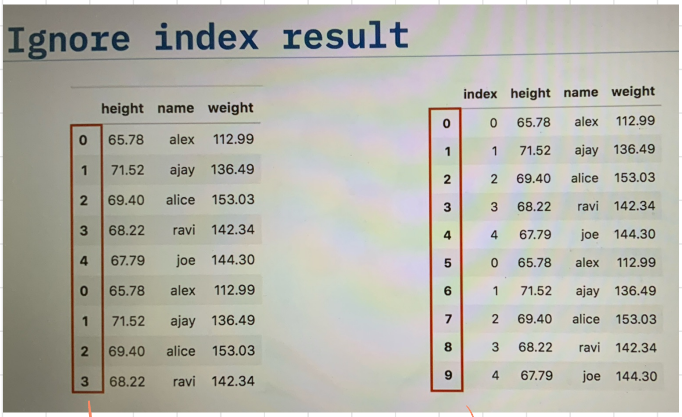


	name	height	weight
0	roger	65.78	110.99
1	bob	63.20	136.49
2	tod	69.40	190.03
3	kate	78.22	262.34
4	moe	66.79	194.30
0	alex	65.78	112.99
1	ajay	71.52	136.49
2	alice	69.40	153.03
3	ravi	68.22	142.34
4	joe	67.79	144.30

ทำให้

csv . file

ต่อกันแบบนี้โดยไม่มี Header



Ignore index result

ถ้าเราไม่ set

Ignore_index = True

	height	name	weight
0	65.78	alex	112.99
1	71.52	ajay	136.49
2	69.40	alice	153.03
3	68.22	ravi	142.34
4	67.79	joe	144.30
0	65.78	alex	112.99
1	71.52	ajay	136.49
2	69.40	alice	153.03
3	68.22	ravi	142.34

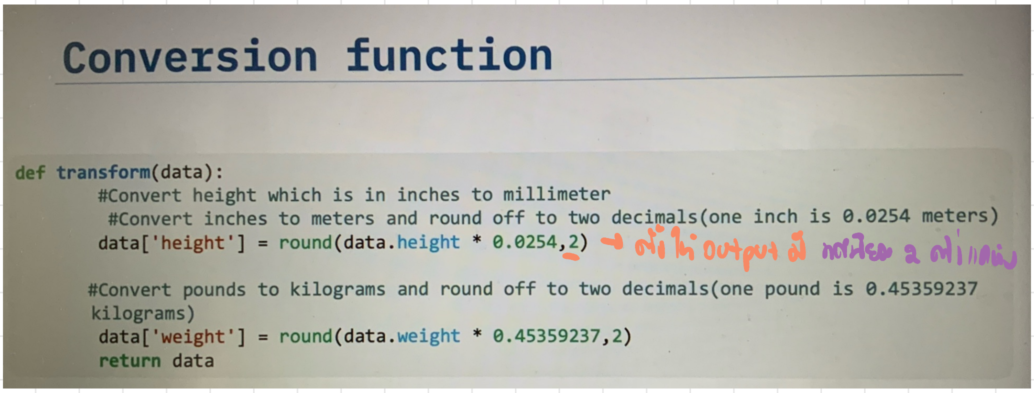
	index	height	name	weight
0	0	65.78	alex	112.99
1	1	71.52	ajay	136.49
2	2	69.40	alice	153.03
3	3	68.22	ravi	142.34
4	4	67.79	joe	144.30
5	0	65.78	alex	112.99
6	1	71.52	ajay	136.49
7	2	69.40	alice	153.03
8	3	68.22	ravi	142.34
9	4	67.79	joe	144.30

ถ้าเป็น False index จะเหมือนกับ

Original file ทำให้ระบุไม่ได้

index = True ทำให้ index จะเรียงไปตาม row ตามลำดับ

Transform:



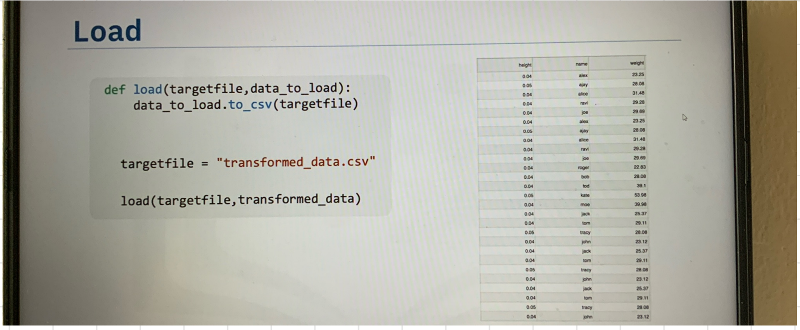
Conversion function

```
def transform(data):
        #Convert height which is in inches to millimeter
         #Convert inches to meters and round off to two decimals(one inch is 0.0254 meters)
        data['height'] = round(data.height * 0.0254,2)

       #Convert pounds to kilograms and round off to two decimals(one pound is 0.45359237
       kilograms)
        data['weight'] = round(data.weight * 0.45359237,2)
        return data
```

→ ตั้งให้ output มี ทศนิยม 2 ตำแหน่ง

Load:

- save Dataframe เป็น .csv



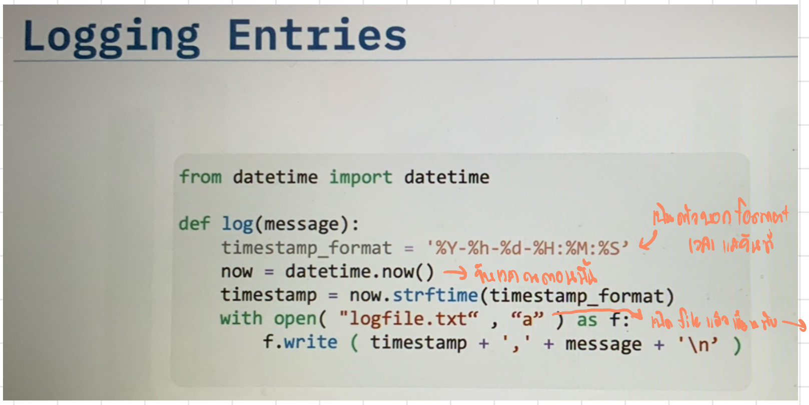
Load

```
def load(targetfile,data_to_load):
    data_to_load.to_csv(targetfile)


 targetfile = "transformed_data.csv"

 load(targetfile,transformed_data)
```

height	name	weight
0.04	alex	23.25
0.05	ajay	28.08
0.04	alice	31.48
0.04	ravi	29.28
0.04	joe	29.69
0.04	alex	23.25
0.05	ajay	28.08
0.04	alice	31.48
0.04	ravi	29.28
0.04	joe	29.69
0.04	roger	22.83
0.04	bob	28.08
0.04	tod	39.1
0.05	kate	53.98
0.04	moe	39.98
0.04	jack	25.37
0.04	tom	29.11
0.05	tracy	28.08
0.04	john	23.12
0.04	jack	25.37
0.04	tom	29.11
0.05	tracy	28.08
0.04	john	23.12
0.04	jack	25.37
0.04	tom	29.11
0.05	tracy	28.08
0.04	john	23.12

Logging:

- ETL เสร็จ ดูว่ามีการทำตอนไหน



Logging Entries

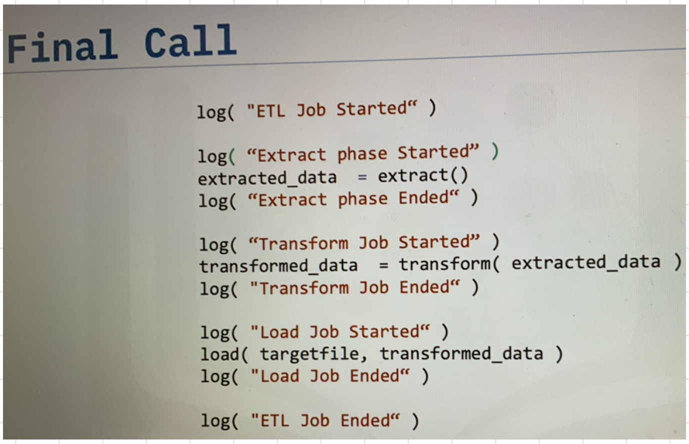
```
from datetime import datetime

def log(message):
    timestamp_format = '%Y-%h-%d-%H:%M:%S'
    now = datetime.now()
    timestamp = now.strftime(timestamp_format)
    with open( "logfile.txt“ , “a” ) as f:
        f.write ( timestamp + ',' + message + '\n’ )
```

เป็นตัวบอก format เวลา และวันที่

→ จับเวลา ณ ตอนนั้น

→ เปิด file แล้วเขียนทับ → ทุกอย่างที่ถูกเขียนจะ append ไปยัง file

เรียกใช้ ETL :



Final Call

```
log( "ETL Job Started“ )

log( “Extract phase Started” )
extracted_data  = extract()
log( “Extract phase Ended“ )

log( “Transform Job Started” )
transformed_data  = transform( extracted_data )
log( "Transform Job Ended“ )

log( "Load Job Started“ )
load( targetfile, transformed_data )
log( "Load Job Ended“ )

log( "ETL Job Ended“ )
```