5.1_solutions

January 27, 2023

1 Solutions

1.0.1 handling files.

*__You will need the folder `data_mixed` in the same location as your notebook__

1. Make a function that gets a directory as argument (for instance `data_mixed`) and returns a dictionnary, where the keys are the names of the elements in the folder, and the values are either `True` or `False` if the element is a file or not.

[1]:
```
import os

def dict_from_path(directory):
    #full_path=os.path.join(os.getcwd(), directory)  # this is an option
    rel_path="./"+directory+'/'                      # this is anther option
    all_elements=os.listdir(rel_path)
    is_file= [os.path.isfile(rel_path+e) for e in all_elements]
    mydict=dict(zip(all_elements, is_file))

    return mydict


mydict= dict_from_path(directory='data_mixed')

print(mydict)
```

```
{'apple.json': True, 'CashFlow.xlsx': True, 'CashFlow_CF_Scenario1.csv': True,
'CashFlow_CF_Scenario2.csv': True, 'CashFlow_CF_Scenario3.csv': True,
'Data_Set_Penguins': False, 'GermanDataClean_OUT.csv': True,
'GermanDataClean_OUT2.csv': True, 'giltBondPrices.txt': True, 'myplot.png':
True, 'my_plot.jpeg': True, 'my_plot.png': True, 'new_germabdataclean.csv':
True, 'optionPortfolio.csv': True, 'optionPortfolio_2.csv': True,
'optionPortfolio_OUT.csv': True, 'StatisticalFieldTheory_NN.pdf': True,
'surfaceplot.pdf': True, 'surfaceplot.png': True, 'timeSeriesData.pkl': True,
'^DJI.csv': True, '^GSPC.csv': True, '^IXIC.csv': True}
```

[2]:
```
# method 2, using a dictionnary comprehension
#which is same as list comprehension but in a dis=ctionnary
```

```
import os

def dict_from_path(directory):
    rel_path="./"+directory+'/'
    all_elements=os.listdir(rel_path)
    mydict= {e : os.path.isfile(rel_path+e) for e in all_elements}

    return mydict

print(dict_from_path('data_mixed'))
```

```
{'apple.json': True, 'CashFlow.xlsx': True, 'CashFlow_CF_Scenario1.csv': True,
'CashFlow_CF_Scenario2.csv': True, 'CashFlow_CF_Scenario3.csv': True,
'Data_Set_Penguins': False, 'GermanDataClean_OUT.csv': True,
'GermanDataClean_OUT2.csv': True, 'giltBondPrices.txt': True, 'myplot.png':
True, 'my_plot.jpeg': True, 'my_plot.png': True, 'new_germabdataclean.csv':
True, 'optionPortfolio.csv': True, 'optionPortfolio_2.csv': True,
'optionPortfolio_OUT.csv': True, 'StatisticalFieldTheory_NN.pdf': True,
'surfaceplot.pdf': True, 'surfaceplot.png': True, 'timeSeriesData.pkl': True,
'^DJI.csv': True, '^GSPC.csv': True, '^IXIC.csv': True}
```

[3]:
```
# not recommendded but also posible
directory='data_mixed'
dict_from_path = lambda path : {e : os.path.isfile("./"+directory+'/'+e) for e␣
 ↪in os.listdir("./"+directory+'/' )}
mydict=dict_from_path('data_mixed')

print(mydict)
```

```
{'apple.json': True, 'CashFlow.xlsx': True, 'CashFlow_CF_Scenario1.csv': True,
'CashFlow_CF_Scenario2.csv': True, 'CashFlow_CF_Scenario3.csv': True,
'Data_Set_Penguins': False, 'GermanDataClean_OUT.csv': True,
'GermanDataClean_OUT2.csv': True, 'giltBondPrices.txt': True, 'myplot.png':
True, 'my_plot.jpeg': True, 'my_plot.png': True, 'new_germabdataclean.csv':
True, 'optionPortfolio.csv': True, 'optionPortfolio_2.csv': True,
'optionPortfolio_OUT.csv': True, 'StatisticalFieldTheory_NN.pdf': True,
'surfaceplot.pdf': True, 'surfaceplot.png': True, 'timeSeriesData.pkl': True,
'^DJI.csv': True, '^GSPC.csv': True, '^IXIC.csv': True}
```

[4]:
```
# Not even naming the function

mydict=(lambda path : {e : os.path.isfile("./"+directory+'/'+e) for e in os.
 ↪listdir("./"+directory+'/' )})('data_mixed')

print(mydict)
```

```
{'apple.json': True, 'CashFlow.xlsx': True, 'CashFlow_CF_Scenario1.csv': True,
'CashFlow_CF_Scenario2.csv': True, 'CashFlow_CF_Scenario3.csv': True,
'Data_Set_Penguins': False, 'GermanDataClean_OUT.csv': True,
```

```
'GermanDataClean_OUT2.csv': True, 'giltBondPrices.txt': True, 'myplot.png':
True, 'my_plot.jpeg': True, 'my_plot.png': True, 'new_germabdataclean.csv':
True, 'optionPortfolio.csv': True, 'optionPortfolio_2.csv': True,
'optionPortfolio_OUT.csv': True, 'StatisticalFieldTheory_NN.pdf': True,
'surfaceplot.pdf': True, 'surfaceplot.png': True, 'timeSeriesData.pkl': True,
'^DJI.csv': True, '^GSPC.csv': True, '^IXIC.csv': True}
```

2. Create a file with two lines of text and close it

[5]:
```
# I will put the file in the same direcgory as this jupyte notebook
my_file='test_file.txt'
f= open(my_file, "w")
f.write("So close, no matter how far \n")
f.write("Couldn't be much more from the heart \n")
f.close()
```

[6]:
```
# method 2, better
my_file='test_file.txt'
with open(my_file, "w") as f:
    f.write("So close, no matter how far \n")
    f.write("Couldn't be much more from the heart \n")
```

3. Open the file that you just created and append 2 other lines, then close it

[7]:
```
my_file='test_file.txt'
with open(my_file, "a") as f:  # we use 'a' dfor append
    f.write("Forever trusting who we are \n")
    f.write("And nothing else matters. \n")
```

4. Open the file that you just created and modified, and print all the lined except the third one on screen

[8]:
```
with open(my_file, "r") as f:
    for i,line in enumerate(f):
        if i == 2:  # remember, third line means index = 2
            continue
        print(i, line)
```

```
0 So close, no matter how far

1 Couldn't be much more from the heart

3 And nothing else matters.
```

5. The dataset `Data_Set_Penguins` contains three subfolders, each of one containing a number of images. Make a list containing the names of all the files in the three subfolders.

[9]:
```
# a quick visual inspection of the data
path='./data_mixed/Data_Set_Penguins'
os.listdir(path)
```

[9]:
```
['P01', 'P02', 'P03']
```

[10]:
```
os.listdir('./data_mixed/Data_Set_Penguins/P01')
```

[10]:
```
['Adam_000.JPG',
 'Adam_001.JPG',
 'Adam_002.JPG',
 'Adam_003.JPG',
 'Adam_004.JPG']
```

[11]:
```
# now we go

path_0='./data_mixed/Data_Set_Penguins'
subfolders=[path_0+'/'+sf for sf in os.listdir(path)]
subfolders
```

[11]:
```
['./data_mixed/Data_Set_Penguins/P01',
 './data_mixed/Data_Set_Penguins/P02',
 './data_mixed/Data_Set_Penguins/P03']
```

[12]:
```
#method 1. Using loops (one for subfolders, and another one for files in each␣
 ↪subfolder)
# then append each file to a list

path_0='./data_mixed/Data_Set_Penguins'
subfolders=[path_0+'/'+sf for sf in os.listdir(path)]

all_files=[]
for sf in subfolders:
    files_here=os.listdir(sf)
    for f in files_here:
        all_files.append(f)


print(all_files)
```

```
['Adam_000.JPG', 'Adam_001.JPG', 'Adam_002.JPG', 'Adam_003.JPG', 'Adam_004.JPG',
'Albert_000.JPG', 'Albert_001.JPG', 'Albert_002.JPG', 'Albert_003.JPG',
'Anoki_000.JPG', 'Anoki_001.JPG', 'Anoki_002.JPG']
```

[13]:
```
#method 2. subtitute a for loop for a list comprehension, and .append() for .
 ↪extent()

path_0='./data_mixed/Data_Set_Penguins'
```

```
subfolders=[path_0+'/'+sf for sf in os.listdir(path)]

all_files=[]
for sf in subfolders:
    list_files_here=[f for f in os.listdir(sf)]  #this list comp. replaces the␣
 ↪inner for loop
    all_files.extend(list_files_here)

print(all_files)
```

```
['Adam_000.JPG', 'Adam_001.JPG', 'Adam_002.JPG', 'Adam_003.JPG', 'Adam_004.JPG',
'Albert_000.JPG', 'Albert_001.JPG', 'Albert_002.JPG', 'Albert_003.JPG',
'Anoki_000.JPG', 'Anoki_001.JPG', 'Anoki_002.JPG']
```

[14]:
```
# method 3.  all with list comprehensions [ element for 'outer loop' for 'inner␣
 ↪loop' ]
## This is too complicated and therefore not recommended!!!
#personnaly, I think method 2 is a better compromise between concise and␣
 ↪readable

path_0='./data_mixed/Data_Set_Penguins'
subfolders=[path_0+'/'+sf for sf in os.listdir(path)]

[f for sf in subfolders for f in os.listdir(sf) ]
```

[14]:
```
['Adam_000.JPG',
 'Adam_001.JPG',
 'Adam_002.JPG',
 'Adam_003.JPG',
 'Adam_004.JPG',
 'Albert_000.JPG',
 'Albert_001.JPG',
 'Albert_002.JPG',
 'Albert_003.JPG',
 'Anoki_000.JPG',
 'Anoki_001.JPG',
 'Anoki_002.JPG']
```

[ ]:
```

```