9.50 / 10

Group_08_exersice_04

January 11, 2021

1 Exercice 1:

3/3 pts

write a function that takes a filename and then all the urls in the file "urls.txt" line by line. Remove the `http://www.` parts of each url and write the urls without `http://www.`part in the file "domains.txt". The function returns nothing.

Examples:

`http://www.rakuten.co.jp` should be `rakuten.co.jp`

`http://www.craigslist.org` should be `craigslist.org`

`http://www.amazon.de` should be `amazon.de`

[3]:
```
# your code here

def remove_url(file_1,file_2):

    a=[]

    with open(file_1,"r") as file:
        for line in file:
            a.append(line[11:])


    with open(file_2, "w") as file:
        for url in a:
            file.write(url)
```

[4]:
```
remove_url("urls.txt","domains.txt")
```

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
<ipython-input-4-e7d99c124f40> in <module>
----> 1 change_file("urls.txt","domains.txt")

<ipython-input-3-d869958b84b2> in change_file(file_1, file_2)
      5     a=[]
      6
----> 7     with open(file_1,"r") as file:
      8         for line in file:
      9             a.append(line[11:])

FileNotFoundError: [Errno 2] No such file or directory: 'urls.txt'
```

2 Exercice 2: (Hint: Use lists inside a list to write a two dimensional array. Also use a for loop inside a for loop could be helpful)

3.25/3.5 pts

Without using exernal library, create a function which print a matrix `n x n`with `1`on the diagonal, otherwise `0`. The function should take an argument `n`and then print the matrix.

Example: . for `n = 3`

```
1  0  0
0  1  0
0  0  1
```

. for `n = 4`

```
1  0  0  0
0  1  0  0
0  0  1  0
0  0  0  1
```

etc…

[1]:

```
# your code here
def matrix(n):
    matrix=[]
    for i in range(n):
        element=[]
        for j in range(n):
            if i==j:
                element.append(1)
            else:
                element.append(0)
        matrix.append(element)

    for line in matrix:
        print(line)

matrix(5)
```

```
[1, 0, 0, 0, 0]
[0, 1, 0, 0, 0]
```

```
[0, 0, 1, 0, 0]
[0, 0, 0, 1, 0]
[0, 0, 0, 0, 1]
```

3 Exercice 3:

3.25/3.50 pts

Without using exernal library, compute the sum of two matrix. The sum of two matrix can be done as shown in the WIKI-PAGE . The function should be able to sum both `n x n` matrix and `n x m` matrix. Write a function witch takes two matrix and return the (sum) matrix.

You can use the function from exercice 2, to print the result :)

[10]:

```python
# your code here
def sum_matrix(a:list,b:list):
    result=a
    for i in range (len(a)):
        for j in range(len(a[0])):
            result[i][j]=a[i][j] + b[i][j]

    for r in result:
        print(r)

def direct_sum(c,d):
    result=[]
    for i in range(len(c)+len(d)):
        line=[]
        for j in range(len(c[0])+len(d[0])):
            if i<len(c):
                if j<len(c[0]):
                    line.append(c[i][j])
                else:
                    line.append(0)
            else:
                if j<len(c[0]):
                    line.append(0)
                else:
                    line.append(d[i-len(c)][j-len(c[0])])
        result.append(line)

    for r in result:
        print(r)
```

[11]:

```python
matrix_1 = [[1,3],[1,0],[1,2]]
matrix_2 = [[0,0],[7,5],[2,1]]
sum_matrix(matrix_1,matrix_2)
```

```
[1, 3]
[8, 5]
```

```
[3, 3]
```

[14]:

```
# test your functions

matrix_1 = [[1,3],[1,0],[1,2]]
matrix_2 = [[0,0],[7,5],[2,1]]

matrix_3 = [[1,3,2],[2,3,1]]
matrix_4 = [[1,6],[0,1]]

#sum_entrywise = sum_matrix(matrix_1, matrix_2)
#print(format_print(sum_entrywise))

sum_direct_sum = direct_sum(matrix_3, matrix_4)
#print(format_print(sum_direct_sum))
```

```
[1, 3, 2, 0, 0]
[2, 3, 1, 0, 0]
[0, 0, 0, 1, 6]
[0, 0, 0, 0, 1]
```

[ ]:

```
```