Reference : https://sparkbyexamples.com/pandas/pandas-map-function-explained/

1. Drops columns

df.drop("U\_Keys",axis=1, inplaPrce=True)

# Remove two columns name is 'C' and 'D'

df.drop(['C', 'D'], axis**=**1

2.Drops Duplicate

cutoff.drop\_duplicates(keep=False,inplace=True)

\*Remove duplicate \*

dd = dfRO.drop\_duplicates(subset='BKH - Booking Ref',keep=False)

=>>False (Drop all duplicate ) >>First (Except first occurance) >>Last(Except Last occurance)

Note : duplicate impacted due to lower and upper case sensivity.

# Remove all duplicate rows

df2 = df.drop\_duplicates(keep=False)

# Delete duplicate rows based on specific columns

df2 = df.drop\_duplicates(subset=["Courses", "Fee"], keep=False)

# Drop duplicate rows in place

df.drop\_duplicates(inplace=True)

# Using DataFrame.apply() and lambda function

df2 = df.apply(lambda x: x.astype(str).str.lower()).drop\_duplicates(subset=['Courses', 'Fee'], keep='first')

3.Replace blank cell with 0

df['Voyage Reference'].fillna(0,inplace=True)

4.map/merge \*\*https://datagy.io/vlookup-in-python-and-pandas-using-map-or-merge/

5.Rename to existing columns

df.rename(columns={'Voyage Reference':'Voyage'},inplace=True)

6.Concate 2 data series & add 2 columns.

df['Uniq'] = df['Voyage'].astype(str)+'|'+df['Stop Location'].astype(str)

7.Sort the pandas series

cutoff['Uniq'] = cutoff['Uniq'].sort\_values(axis=0,ascending=True, inplace=False)

dfRO = dfRO.sort\_values(by='VR - Rollover Date',ascending=True)

8.Vlookup function in python

df = pd.merge(df,cutoff[['Uniq','Cut-Off']],on='Uniq',how='left')

different type of merge.

https://jakevdp.github.io/PythonDataScienceHandbook/03.07-merge-and-join.html

9. Converting pandas.tslib.Timestamp to datetime python

d2 = rw['Cut-Off\_y'].to\_pydatetime() #with date & time

d2 = rw['Cut-Off\_y'].to\_pydatetime().date() # with date

10.Add multiple Blank columns

colmns1 = ['Fault','Split','Original Booking','Door']

df[colmns1] = ""

11. Right function pandas dataseries

df['Job Reference'].str[-1:]

12.Selecting multiple columns in a Pandas dataframe

nRoll = dfRO[['BKH - Booking Ref','VR - Fault',]]

13.Filter all rows that do not contain letters (alpha) in ´pandas´

df = df[df['Col A.'].str.contains('[A-Za-z]')]

14. Filter Alphabet in column excluding blank

df[(df['Split'].str.contains('[A-Za-z]',na=False))

**import** **re**

**>>>** s1.str.contains('PARROT', flags=re.IGNORECASE, regex=**True**)

15. Convert Data type of pandas series String to number

df['Job Reference'] = df['Job Reference'].astype(str)

final\_df['Deliverable Qty'] = pd.to\_numeric(final\_df['Deliverable Qty'])

16.CountIf function in pytho

df['COUNTIFS'] = df.groupby('user\_id').cumcount() + 1

17. Fill empty cells in column with value of other columns

hc['ID'].fillna(hc['First Name'] + hc['Last Name'], inplace=True)

OR

hc.loc[hc["ID"].isnull(),'ID'] = hc["First Name"] + hc["Last Name"]

18.reduce day from date

currDate = date.today()-timedelta(days=3)

19.Convert string to date

cnvtFormat = datetime.strptime(rw['arrDtlocAct'], '%Y-%m-%d %H:%M')

20. DataFrame: if value in a cell, copy value to cells below it

df.UNP[‘action\_amount’].fillna(method='backfill')

ptners1 = ptners1.fillna(method='ffill')

21. convert pandas series to string values,

strVal = ''.join(df)

22.Extract method with list of ..

<https://www.w3resource.com/pandas/series/series-str-extractall.php>

<https://www.shecancode.io/blog/filter-a-pandas-dataframe-by-a-partial-string-or-pattern-in-8-ways>

# 23. [Convert Pandas Series to DateTime in a DataFrame](https://stackoverflow.com/questions/28133018/convert-pandas-series-to-datetime-in-a-dataframe)

df["TimeReviewed"] = pd.to\_datetime(df["TimeReviewed"])

# 24. Python – Find the closest date from a List

<https://www.geeksforgeeks.org/python-find-the-closest-date-from-a-list/>

abs = date values.

res **=** min(test\_date\_list, key**=lambda** sub: abs(sub **-** test\_date))

for id,rw in DB\_report1.iterrows():

    try:

        outPt = ptners1.loc[ptners1['protName'].str.contains(rw['POINT\_NAME'], case=False,na=False) &  ptners1['vesselName'].str.contains(rw['VESSEL\_NAME'], case=False,na=False),'arrDtlocAct']

        outPt = pd.to\_datetime(outPt)

        result = outPt.to\_list()

        # lkVal  =datetime.strptime(rw['ETA\_DATE'], '%Y-%m-%d %H:%M:%S')

        lkVal = rw['ETA\_DATE'].to\_pydatetime()

        res = min(result, key=lambda sub: abs(sub - lkVal))

        DB\_report1.loc[id,'new'] = res

    except:

        continue

25.Conditional formatting in datafram

<https://www.youtube.com/watch?v=qUj7BVZmuiw>

<https://stackoverflow.com/questions/68056855/pandas-style-conditional-formatting-highlight-on-text>

https://queirozf.com/entries/pandas-dataframe-examples-styling-cells-and-conditional-formatting

def color\_negative\_red(value):

    # print(row)

    # value = row.loc['Date\_RangeGap']

    if value == '30':

        color = 'yellow'

    elif value == '15-29':

        color = 'orange'

    elif value == '1-14':

        color = 'green'

    else:

        color = 'white'

    # return ['background-color : %s' % color for r in row]

    return 'background-color : %s' % color

DB\_report1 = DB\_report1.style.applymap(color\_negative\_red,subset='Date\_RangeGap')

* Styler.applymap(func) for element-wise styles.
* Styler.apply(func, axis=0) for column-wise styles.
* Styler.apply(func, axis=1) for row-wise styles.
* Styler.apply(func, axis=None) for tablewise styles.

26. build the regex expression online

https://regex101.com/

# 27. How to Use Like Operator in Pandas DataFrame

df[df['class'].str.contains('sh|rd', regex=True, na=True)]

28. Convert string to integer in Python

# convert the num into string

converted\_num **=** int(num)

# convert the num into string

converted\_num **=** float(num)

29. Drop rows from the dataframe based on certain condition applied on a column

# First filter out those rows which

# does not contain any data

df **=** df.dropna(how **=** 'all')

# Filter all rows for which the player's

# age is greater than or equal to 25

df.drop(df[df['deleted\_Lanes']=='Del'].index,inplace=True)

30.Regex find sub-string from string.

# re.findall(r'(.\*)day','-1 day 5 hours 39 minutes 47 seconds')

31.Read from multiple sheets

import pandas as pd

df = pd.read\_excel('users.xlsx', sheet\_name = [0,1,2])

df = pd.read\_excel('users.xlsx', sheet\_name = ['User\_info','compound'])

df = pd.read\_excel('users.xlsx', sheet\_name = None) # read all sheets

## pd.ExcelFile()

With this approach, we create a **pd.ExcelFile** object to represent the Excel file. We do not need to specify which sheets to read when using this method. Note that the previous **read\_excel()** method returns a dataframe or a dictionary of dataframes; whereas **pd.ExcelFile(**) returns a reference object to the Excel file.

f = pd.ExcelFile('users.xlsx')

>>> f

<pandas.io.excel.\_base.ExcelFile object at 0x00000138DAE66670>

To get data from a sheet, we can use the **parse()** method, and provide the sheet name.

>>> df.parse(sheet\_name = 'User\_info')

## 32. LEFT, RIGHT and MID Functions

https://www.listendata.com/2019/06/python-string-functions.html

|  |  |  |
| --- | --- | --- |
| **Function** | **Description** | **MS EXCEL FUNCTION** |
| mystring[:N] | Extract N number of characters from start of string. | LEFT( ) |
| mystring[-N:] | Extract N number of characters from end of string | RIGHT( ) |
| mystring[X:Y] | Extract characters from middle of string, starting from X position and ends with Y | MID( ) |
| str.split(sep=' ') | Split Strings | - |
| str.replace(old\_substring, new\_substring) | Replace a part of text with different sub-string | REPLACE( ) |
| str.lower() | Convert characters to lowercase | LOWER( ) |
| str.upper() | Convert characters to uppercase | UPPER( ) |
| str.contains('pattern', case=False) | Check if pattern matches  (Pandas Function) | SQL LIKE Operator |
| str.extract(regular\_expression) | Return matched values (Pandas Function) | - |
| str.count('sub\_string') | Count occurence of pattern in string | - |
| str.find( ) | Return position of sub-string or pattern | FIND( ) |
| str.isalnum() | Check whether string consists of only alphanumeric characters | - |
| str.islower() | Check whether characters are all lower case | - |
| str.isupper() | Check whether characters are all upper case | - |
| str.isnumeric() | Check whether string consists of only numeric characters | - |
| str.isspace() | Check whether string consists of only whitespace characters | - |
| len( ) | Calculate length of string | LEN( ) |
| cat( ) | Concatenate Strings (Pandas Function) | CONCATENATE( ) |
| separator.join(str) | Concatenate Strings | CONCATENATE( ) |

df['StateInitial'] = df['state'].str[:2]+ df[‘city'].str[:3]

df\_data['Job\_Bkg\_POL'] = df\_data['Job Reference'].str[:3] + df\_data['Booking POL'].str[:2]

## 33. VLOOKUP in Python

total\_merge = df1.merge(df2, on='id', how='outer', indicator=True)

R1 = total\_merge[total\_merge['\_merge']=='both']

R2 = total\_merge[total\_merge['\_merge']=='left\_only']

R3 = total\_merge[total\_merge['\_merge']=='right\_only']

ValueError: Cannot use name of an existing column for indicator column python

Because you did one merge, you now have a column called \_merge in the your df3. And when you merge again, you cannot create yet another \_merge.

### 33. **Remove spaces Trim**

**Remove spaces in the BEGINNING and END of a string:**

sentence= sentence.strip()

df1['State'] **=** df1['State'].str.strip()

**Remove spaces in the BEGINNING of a string:**

sentence = sentence.lstrip()

df1['State'] **=** df1['State'].str.lstrip()

**Remove spaces in the END of a string:**

sentence= sentence.rstrip()  
df1['State'] **=** df1['State'].str.rstrip()

\*\*\*\*eliminate all the whitespace from a string, on both ends, and in between words.\*\*

>>> import re

>>> re.sub("\s+", # one or more repetition of whitespace

'', # replace with empty string (->remove)

''' hello

... apple

... ''')

### strip()

Python String strip() function will remove leading and trailing whitespaces

[34.](https://gist.github.com/eldala07/c2ef7176b3e43b6f45405d189b5242e1)**[Find, open and reply all to an outlook mail with python](https://gist.github.com/eldala07/c2ef7176b3e43b6f45405d189b5242e1)**

Find last mail containing a certain subject in the 'sent items'

# Open the mail with reply all function

for message in messages\_sent:

    # print('Message class: {}'.format(message.Class))

    # print('Message subject: {}'.format(message.Subject.encode("utf-8")))

    # Message class 43 -> mail

    if message.Class == 43:

        if message\_subject\_to\_find in message.Subject:

            subject\_found = message.Subject

            found = True

            message.ReplyAll().Display()

            break

35. Validate the email id :

import re

valid\_regex = r'\b[A-Za-z0-9.\_%+-]+@[A-Za-z0-9.-]+\.[A-Z|a-z]{2,}\b'

def check(email):

if(re.fullmatch(valid\_regex, email)):

print("This is a valid email")

else:

print("This is an invalid email")

email = "correctemail@gmail.com"

check(email)

36.TkInter design

from tkinter import \*

import tkinter.font as font

gui = Tk(className='Python Examples - Button')

gui.geometry("500x200")

myFont = font.Font(family='Helvetica', size=20, weight='bold')

button = Button(gui, text='My Button', bg='#0052cc', fg='#ffffff')

# apply font to the button label

button['font'] = myFont

# add button to gui window

button.pack()

gui.mainloop()

+++++++++++++++++++++++++++++++++++++++++==

myFont = font.Font(family='Helvetica', size=18, weight='bold')

lbl = tk.Label(gui, text='Email ID')

lbl['font'] = myFont

lbl.place(relx=0.0, rely=0.1, anchor='w')

entry = Entry(gui, width= 42)

entry.place(relx= 0.3, rely= 0.1, anchor= CENTER)

lbl = tk.Label(gui, text='Folder')

lbl['font'] = myFont

lbl.place(relx=0.5, rely=0.1, anchor='w')

entry2 = Entry(gui, width= 42)

entry2.place(relx= 0.8, rely= 0.1, anchor= CENTER)

button = Button(gui, text='GetMail', bg='#0052cc', fg='#ffffff',height= 2, width=10,command=extrM)

button['font'] = myFont

button.place(relx=0.5, rely=0.5, anchor=CENTER)

gui.mainloop()

++++++++++++++++++++++++++++++++++++++++++=

37. Send mail for another accounts outlook

    outlook = win32com.client.Dispatch("Outlook.Application")

    oacctuse = None

    for oac in outlook.Session.Accounts.\_dispobj\_:

        if oac.DisplayName == frm:

            oacctuse = oac

            break

38.Create folder and write notepad

    target\_fldr = output\_dir/str(Subj)

    target\_fldr.mkdir(parents=True,exist\_ok=True)

    Path(target\_fldr/"Email\_body.txt").write\_text(str(body))

39. Save & rename file the attachment in outlook

   attachment.SaveAsFile(os.path.join(outputDir,attachment.FileName))

                        f1 = os.path.join(outputDir,attachment.FileName)

                        f2 = os.path.join(outputDir,newName)

                        os.rename(f1,f2)

40. Multiple columns combine

df['FullName'] = df[['First\_Name', 'Last\_Name']].apply(lambda x: '\_'.join(x), axis=1)

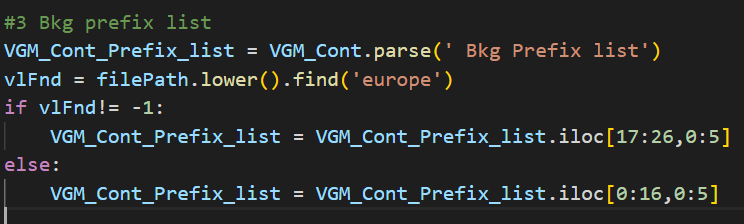
41. Replace NA with value

df\_4['VGM contact'] = df\_4['VGM contact'].fillna(";")

42.Drop duplicate the data based on multiple columns.

newdf = df1.drop\_duplicates(subset = ['order\_id', 'customer\_id'], keep = 'last').reset\_index(drop = True)

43. Find Funtion with row column range



44. askFile dialog

from tkinter import filedialog

def browseFoldr():

    fldr = filedialog.askdirectory()

    return fldr

45. Python Pandas Error tokenizing data – csv

df = pandas.read\_csv(filepath, sep='delimiter', header=None)

# 46. pandas.Series.str.split

**Series.str.split(*pat=None*, *n=- 1*, *expand=False*, *\**, *regex=None*)**

# 47. Concate dataframe with identical columns

dfCom = pd.concat([dfCom,dfN])

# 48. Pivot table : https://pandas.pydata.org/docs/user\_guide/reshaping.html

 = dfCom.pivot(index='Stock Name',columns='Date\_Based',values='RS')

# 49. [how to sort pandas dataframe from one column](https://stackoverflow.com/questions/37787698/how-to-sort-pandas-dataframe-from-one-column)

final\_df = df.sort\_values(by=['2'], ascending=False)

50. Convert string to date

pd.to\_datetime(final\_df.loc[id,'Date'])

51.Sort date in dataframe

        final\_df['Date'] = pd.to\_datetime(final\_df['Date'])

        final\_df.sort\_values(by='Date', ascending=False,inplace=True)

52. clean or drop dataframe data

final\_df.drop(final\_df.index[:],inplace=True)

**53.** ValueError: Index contains duplicate entries, cannot reshape

**df.pivot\_table(index='team', columns='position', values='points', aggfunc='sum')**

# 54. [Convert column of date objects in Pandas DataFrame to strings](https://stackoverflow.com/questions/19738169/convert-column-of-date-objects-in-pandas-dataframe-to-strings)

dfComNSE['Date'] = dfComNSE['Date'].dt.strftime('%d-%b-%Y')

55. Highlight the cell values

def color\_negative\_red(val):

    color = 'red' if val < 0 else 'green'

    return 'color: %s' % color

pvTbl = pvTbl.style.applymap(color\_negative\_red)

56. Cell color

<https://www.blog.pythonlibrary.org/2021/08/11/styling-excel-cells-with-openpyxl-and-python/>

57. Fille the sheet cells

def background\_colors(path):

    wb = load\_workbook('Result.xlsx')

    sheet = wb.active

    rw\_range = sheet.max\_row

    col\_range = sheet.max\_column

    for cl in range(2,(col\_range+1)):

        for rw in range(4,(rw\_range+1)):

            if sheet.cell(rw,cl).value <0:

                sheet.cell(rw,cl).fill = PatternFill(start\_color='FF0000', end\_color='FF0000', fill\_type="solid")

            else:

                sheet.cell(rw,cl).fill = PatternFill(start\_color='00B050', end\_color='00B050', fill\_type="solid")

    wb.save(path)

background\_colors("Result.xlsx")

58. google translate

https://medium.com/analytics-vidhya/exploring-google-cloud-translate-api-for-machine-translation-in-python-ef60c123fc37

59. read multiple sheets in excel

bkID = pd.ExcelFile('Bookler\_Emails\_List.xlsx')

bkID.parse('SheetName')

60. Rename with Reset index

nTbl.reset\_index().rename(columns={'Job Reference':'Booking','Container Number':'Container','Booking POL':'POL','Voyage':'Voyage Ref','Cut Off Local Date':'VGM Cut-off','Remaining time before cut off':'Day remaining to send VGM'},inplace=True)

61. Outlook Html body content replace with html table

mailT['Body11'].fillna("",inplace=True)

mail.HTMLBody = "{0}".format(mailT.to\_html(header=False,index=False,justify='left',border='0'))

mail.HTMLBody = mail.HTMLBody.replace("{0}",nTbl.to\_html())

mail.Display()

62. Remove file from folder

os.remove('c:\\Users\\ssc.achauhan\\Desktop\\Projects\\VGM\_\\Excel\\chaser2.csv')

63. String contains multiple substring

exlChanelN['POL\_N'].str.contains('JP|TW|SG')

64. Read XLSB/Binary File in Pandas Python

import pandas as pd

dfcluster = pd.read\_excel('c:/xml/baseline/distribucion.xlsb', sheet\_name='Cluster', index\_col=0, engine='pyxlsb')

65.Read PDF and merge

**from** PyPDF2 **import** PdfFileMerger

#Create and instance of PdfFileMerger() class

merger = PdfFileMerger()

#Create a list with file names

pdf\_files = ['pdf\_files/sample\_page1.pdf', 'pdf\_files/sample\_page2.pdf']

#Iterate over the list of file names

**for** pdf\_file **in** pdf\_files:

#Append PDF files

merger.append(pdf\_file)

#Write out the merged PDF

merger.write("merged\_2\_pages.pdf")

merger.close()

66. Pandas Series.str.startswith()

bool\_series **=** data["College"].str.startswith(search, na **=** False)

67. Extract email from string

import pandas as pd

import re as re

pd.set\_option('display.max\_columns', 10)

df = pd.DataFrame({

'name\_email': ['Alberto Franco af@gmail.com','Gino Mcneill gm@yahoo.com','Ryan Parkes rp@abc.io', 'Eesha Hinton', 'Gino Mcneill gm@github.com']

})

print("Original DataFrame:")

print(df)

def find\_email(text):

email = re.findall(r'[\w\.-]+@[\w\.-]+',str(text))

return ",".join(email)

df['email']=df['name\_email'].apply(lambda x: find\_email(x))

print("\Extracting email from dataframe columns:")

print(df)

68. Select columns basis on index

# Using iloc[] to select column by Index

df2 = df.iloc[:,[1,3,4]] # Select columns by Index

df2 = df.iloc[:,1:4] # Select between indexes 1 and 4 (2,3,4)

df2 = df.iloc[:,2:] # Select From 3rd to end

df2 = df.iloc[:,:2] # Select First Two Columns

Copy

69. select rows basis on index

result **=** df.iloc[2]

# to select multiple rows

result **=** df.iloc[[2, 3, 5]]

70. Get index of Dataframe

dd = df.index.to\_list()

71. Get 09 as month

"{:02d}".format(tday.month-1)

72. Get name of month

import calendar

for month\_idx in range(1, 13):

print (calendar.month\_name[month\_idx])

print (calendar.month\_abbr[month\_idx])

print ("")

73. Read excel file with specific columns

"usecols" should help, use range of columns (as per excel worksheet, A,B...etc.) below are the examples

**1. Selected Columns**

df = pd.read\_excel(file\_location,sheet\_name='Sheet1', usecols="A,C,F")

**2. Range of Columns and selected column**

df = pd.read\_excel(file\_location,sheet\_name='Sheet1', usecols="A:F,H")

**3. Multiple Ranges**

df = pd.read\_excel(file\_location,sheet\_name='Sheet1', usecols="A:F,H,J:N")

**4. Range of columns**

df = pd.read\_excel(file\_location,sheet\_name='Sheet1', usecols="A:N")

74. consider row as column header

.

dk = df.drop(index=df.index[:skpRw], axis=0)

header\_row = dk.iloc[0]

df4 = pd.DataFrame(dk.values[1:], columns=header\_row)

75. Send outlook mail

import win32com.client as client

from pathlib import Path

id = 'ssc.achauhan@cma-cgm.com'

outlook = client.Dispatch("Outlook.Application")

ol\_msg = outlook.CreateItem(0)

ol\_msg.To = id

ol\_msg.Subject = 'New indicators : - EQM Tracking -' + shName

ol\_msg.Body = 'Hi,\n\nPlease find the attached file as requested.\n\nThank you'

attachment1 = shName + '.xlsx'

src\_file = Path.cwd() / attachment1

ol\_msg.Attachments.Add(str(src\_file))

# ol\_msg.display()

outlook.Send()

76. How to delete rows from a pandas DataFrame based on a conditional expression [duplicate].

lUnt = df\_11[df\_11['PTS Code']==df\_11['Final POD']].index

df\_11.drop(lUnt,inplace=True)

df.drop(df[df.score < 50].index, inplace=True)

77. Exe file Generate with package erro

importlib.metadata.PackageNotFoundError: No package metadata was found for pikepdf python python

\*\* pyinstaller --onefile --copy-metadata pyproj "example.py" \*\*

78. Get cell value of dataframe without index

print(df.to\_string(index=False))

79. Tkinter messagebox

**from** tkinter **import** **\***

**from** tkinter **import** messagebox

root **=** Tk()

root.geometry("300x200")

w **=** Label(root, text **=**'GeeksForGeeks', font **=** "50")

w.pack()

messagebox.showinfo("showinfo", "Information")

messagebox.showwarning("showwarning", "Warning")

messagebox.showerror("showerror", "Error")

messagebox.askquestion("askquestion", "Are you sure?")

messagebox.askokcancel("askokcancel", "Want to continue?")

messagebox.askyesno("askyesno", "Find the value?")

messagebox.askretrycancel("askretrycancel", "Try again?")

root.mainloop()

80. FileDialog

import tkinter.filedialog  
   
tkinter.filedialog.asksaveasfilename()  
tkinter.filedialog.asksaveasfile()  
tkinter.filedialog.askopenfilename()  
tkinter.filedialog.askopenfile()  
tkinter.filedialog.askdirectory()  
tkinter.filedialog.askopenfilenames()  
tkinter.filedialog.askopenfiles()

# 81. [How to get the length of a cell value or count of cell value in pandas dataframe](https://stackoverflow.com/questions/37335598/how-to-get-the-length-of-a-cell-value-in-pandas-dataframe)

df['EventItem'].apply(len)

%timeit df['Event'].str.count("/") + 1

100 loops, best of 3: 3.18 ms per loop

%timeit df['Event'].str.split("/").str.len()

100 loops, best of 3: 4.28 ms per loop

%timeit df['Event'].str.split("/").apply(len)

# 82. [Split cell into multiple rows in pandas dataframe](https://stackoverflow.com/questions/50731229/split-cell-into-multiple-rows-in-pandas-dataframe)

https://stackoverflow.com/questions/50731229/split-cell-into-multiple-rows-in-pandas-dataframe

k = (df.apply(lambda x: x.str.split(',').explode()).reset\_index())

83. Change column heading and replace alt enter with space

Hrp.columns = Hrp.columns.str.strip().str.lower().str.replace('\n', '\_').str.replace('(', '').str.replace(')', '')

84. Change header with first row in Pandas Dataframe

#set the location of the first row with columns

data.columns = data.iloc[0]

## 85. extract a date from a datetime Dataframe

* df['Dates'] = pd. to\_datetime(df['date']). dt. date.
* df['Time'] = pd. to\_datetime(df['date']). dt. time.

86. TypeError: Cannot perform 'rand\_' with a dtyped [object] array and scalar of type [bool]

Ans. keep the code in group( during logical operator) i.e. in bracket

87.  specific cell is empty in a pandas DataFrame

**#check if value in first row of column 'A' is empty**

**print(pd.isnull(df.loc[0, 'A']))**

88.Create empty Dataframe

pd.DataFrame({'email':sf.index, 'list':sf.values})

89. outlook mail with attachment

import os

import win32com.client as win32

# construct Outlook application instance

olApp = win32.Dispatch('Outlook.Application')

olNS = olApp.GetNameSpace('MAPI')

# construct the email item object

mailItem = olApp.CreateItem(0)

mailItem.Subject = 'Dummy Email'

mailItem.BodyFormat = 1

mailItem.Body = "Hello World"

mailItem.To = '<Recipient Email>'

mailItem.Attachments.Add(os.path.join(os.getcwd(), 'bitcoin.png'))

mailItem.Attachments.Add(os.path.join(os.getcwd(), 'csv file.png'))

mailItem.Display()

mailItem.Save()

mailItem.Send()

90. read PDF scan data into text

<https://betterprogramming.pub/how-to-convert-pdfs-into-searchable-key-words-with-python-85aab86c544>

pip install PyPDF2

pip install textract

pip install nltk

import PyPDF2   
import textractfrom nltk.tokenize import word\_tokenize  
from nltk.corpus import stopwords

91. Check if String Contains Substring in Python

str="Hello, World!"

print("World" in str)

===============

str="Hello, World!"

if str.find("World")!=-1:

print("Found the string")

else:

print("Not found!!!")

================

re.search(pattern, string, flags[optional])

92. Split columns

new **=** data["Name"].str.split(" ", n **=** 1, expand **=** True)

# making separate first name column from new data frame

data["First Name"]**=** new[0]

# making separate last name column from new data frame

data["Last Name"]**=** new[1]

93. Convert string to DATETime

**from** dateutil **import** parser

DT **=** parser.parse("Jun 23 2022 07:31PM")

94. Outlook folder items iteration

https://www.youtube.com/watch?v=iP1Ko9sFkzs

95 sorting to dataframe column

sorted\_df = df.sort\_values(by=['Column\_name'], ascending=True)

96. convert input string data into list

lst1 = [int(item) for item in input("Enter the list items : ").split()]

### 97. **Appending dataframe in a for loop**

for i in range(4,11): df=df.append({'Table of 9':i\*9,'Table of 10':i\*10},ignore\_index=True)

98. select range of rows table

pd.read\_excel('resultat-elections-2012.xls', sheet\_name = 'France entière T1T2', skiprows = 2, nrows= 5, usecols = 'A:H')

pd.read\_excel('resultat-elections-2012.xls', index\_col = None, skiprows= 2, nrows= 5, sheet\_name='France entière T1T2', usecols=range(0,8))

<https://sparkbyexamples.com/pandas/pandas-select-rows-by-index/#:~:text=To%20select%20the%20rows%2C%20the,it%20to%20select%20alternate%20rows>.

99. Count Occurrences of Element in Python List

sample\_list = ["a", "ab", "a", "abc", "ab", "ab"]

print(sample\_list.count("a"))

100. Python – Ways to remove duplicates from list

### **Using set()**

unqLabl = list(set(dupLable))

101. Deletion of rows from dataframe

https://sparkbyexamples.com/pandas/pandas-drop-first-n-rows-from-dataframe/#:~:text=Using%20iloc%5B%5D%20to%20Drop%20First%20N%20Rows%20of%20DataFrame,rows%20you%20wanted%20to%20delete.

102. Convert Row to Column Header

<https://sparkbyexamples.com/pandas/pandas-convert-row-to-column-header-in-dataframe/#:~:text=columns()%20to%20Convert%20Row,by%20extracting%20the%20first%20row>.

# Below are quick example

# Assign row as column headers

df.columns = df.iloc[0]

# Using DataFrame.rename()

df2 = df.rename(columns=df.iloc[1])

# Convert row to header and remove the row

df2 = df.rename(columns=df.iloc[0]).loc[1:]

# Using DataFrame.rename() to convert row to column header

df.rename(columns=df.iloc[1], inplace = True)

# Using DataFrame.values[]

header\_row = df.iloc[0]

df2 = pd.DataFrame(df.values[1:], columns=header\_row)

103 . assignment operator

dSht += sht + "|"

104. Find Value in Dataframe anywhere

#How to select all rows of the DataFrame that contain the values 25, 9 #in any of the columns:

df[df.isin([25, 9]).any(axis=1)]

104. Dataframe **Reset index in place**

df.reset\_index(inplace=True)

105. split line with new line

kk.splitlines(True)

106. Selenium

from selenium import webdriver

from selenium.webdriver.chrome.options import Options

from selenium.webdriver.common.keys import Keys

from selenium.common.exceptions import TimeoutException

from selenium.webdriver.support.ui import WebDriverWait

from selenium.webdriver.support import expected\_conditions as EC

from selenium.webdriver.common.by import By

import re

import os

import pandas as pd

import time

from bs4 import BeautifulSoup

# from webdriver\_manager.chrome import ChromeDriverManager

url = 'https://network.infornexus.com/login'

options = Options()

options.add\_argument("--window-size=1920,1200")

# options.add\_argument("--headless")  # temp

os.environ['WDM\_SSL\_VERIFY']='0'    #Disable the SSL

# driver = webdriver.Chrome(ChromeDriverManager().install(),options = options)

driver = webdriver.Chrome(options=options)

driver.get(url)

time.sleep(5)

lgn = driver.find\_element(By.XPATH,'//\*[@id="login"]')

lgn.send\_keys('ssc.pricing\_fmcg@cma-cgm.com')

time.sleep(1)

lgn = driver.find\_element(By.XPATH,'//\*[@id="password"]')

lgn.send\_keys('Fmcg!2018')

time.sleep(1)

driver.find\_element(By.XPATH,'//\*[@id="loginButton"]').click()

time.sleep(5)

**Get chrom driver**

options = Options()

options.headless = True

driver = webdriver.Chrome(executable\_path=r"./chromedriver", options=options)

#Click hyperlink using value

driver.find\_element(By.LINK\_TEXT,'19-1365').click()

driver.find\_element\_by\_partial\_link\_text("value of partial ink text")

107 Open excel file

os.system('out.xlsx')

#### 108. **Drop down options which also support multi select options.**

1. select\_by\_index()
2. select\_by\_value()
3. select\_by\_visible\_text()
4. options property
5. is\_multiple property
6. all\_selected\_options property

109. Hyperlink web

driver.find\_elements(By.LINK\_TEXT, "Text of Link")

#### 110. find\_elements(By.CSS\_SELECTOR)

driver.find\_elements(By.CSS\_SELECTOR, "CSS Selectors")

#### 111. find\_elements(By.CLASS\_NAME)

driver.find\_elements(By.CLASS\_NAME,"class\_of\_element")

#### 112. find\_elements(By.TAG\_NAME)

driver.find\_elements(By.TAG\_NAME, "Tag name")

#### 113. find\_elements(By.PARTIAL\_LINK\_TEXT)

With this strategy, all elements with the partial link text value matching the location will be returned. If no element has a matching partial link text attribute, a NoSuchElementException will be raised.

**Syntax –**

driver.find\_elements(By.PARTIAL\_LINK\_TEXT, "Text of Link")

114. Wait for element in web using selenium

#Set a wait, for elements to load into the DOM

wait10 = WebDriverWait(driver, 10)

wait20 = WebDriverWait(driver, 20)

wait30 = WebDriverWait(driver, 30)

elem = wait10.until(EC.element\_to\_be\_clickable((By.ID, "userID")))

elem.send\_keys(userName)

115. Get table from website

# using of BeautifulSoup in project

tbl\_rw = 1+ len(driver.find\_elements(By.XPATH,'/html/body/table[2]/tbody/tr/td[2]/table/tbody/tr[6]/td/table/tbody/tr'))

tbl\_clms = len(driver.find\_elements(By.XPATH,'/html/body/table[2]/tbody/tr/td[2]/table/tbody/tr[6]/td/table/tbody/tr'))

dfN = pd.DataFrame()

for r in range(0, tbl\_rw+1):

    try:

        combVal = ''

        for p in range(1, 9):

            # obtaining the text from each column of the table

            value = driver.find\_element(By.XPATH,'/html/body/table[2]/tbody/tr/td[2]/table/tbody/tr[6]/td/table/tbody/tr['+str(r)+"]/td["+str(p)+']').text

            combVal += "|" + value

        # lst.append(combVal)

        df = pd.DataFrame({'Tender\_Det':[combVal]})

        dfN = pd.concat([df,dfN])

    except Exception:

        pass

116. VBA Userform in Window center

With Me

.Top = (Application.Height / 2) - (.Height / 2)

.Left = (Application.Width / 2) - (.Width / 2)

End With

117. VBA Userform combobox with Range

For Each cLoc In sht.Range("B:B")

If Len(cLoc) <> 0 Then

With Me.ComboBox2

.AddItem cLoc.Value

End With

End If

Next cLoc

118. Convert sheet into pandas DATAFRAME

from openpyxl import load\_workbook

2import pandas as pd

3

4wb = load\_workbook('sample.xlsx')

5ws = wb['sample']

6

7data = ws.values

8# Get the first line in file as a header line

9columns = next(data)[0:]

10# Create a DataFrame based on the second and subsequent lines of data

11df = pd.DataFrame(data, columns=columns)

119. Rename and Reorder the columnr

#Rename the column

dfN.rename(columns={'PROCESS (location)':'SUB PROCESS'},inplace=True)

columnList = ["Week","FTEs","PROCESS","SUB PROCESS","Actual volume","Expected Average Volumes / week","Volumes Health","TAT Achieved","TAT as per SLA","TAT Health","Accuracy Achieved","Accuracy as per SLA","Quality Health"]

df2 = dfN[columnList]

## 120. Check if a list is empty in Python

l1 = ["Hire", "the", "top", "1%", "freelancers"]

l2 = []

**if** bool(l2):

print("list is empty")

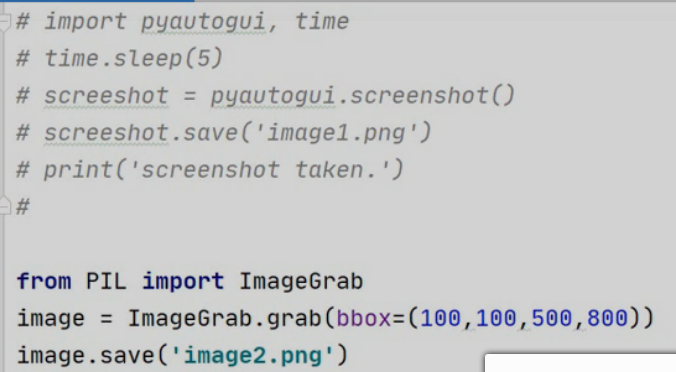
**else**:

print("list is not empty")

121. Image capture and reading GUI

<https://www.google.com/search?q=PyScreezeException%3A+The+Pillow+package+is+required+to+use+this+function.+pyautogui&rlz=1C1CHBD_enIN1014IN1014&oq=PyScreezeException%3A+The+Pillow+package+is+required+to+use+this+function.+pyautogui&aqs=chrome..69i57j69i58.9956j0j7&sourceid=chrome&ie=UTF-8#fpstate=ive&vld=cid:86957f59,vid:SeM6jd2ECTE>

<https://github.com/asweigart/pyautogui> \*\*\*\*\*



##### **How can I loop PyAutoGUI's locateCenterOnScreen until the image is found**

**def** detect\_image(path, duration**=**0):

**while** True:

        image\_location **=** pyautogui.locateCenterOnScreen(path)

**if** image\_location:

            pyautogui.click(image\_location[0], image\_location[1], duration**=**duration)

**break**

if pyautogui not working on window application.

<https://learncodebygaming.com/blog/pyautogui-not-working-use-directinput>

pip install pydirectinput

\*\*\*\*\*https://github.com/asweigart/pyautogui\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*RPA\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

122. UserWarning: : Data Validation extension is not supported and will be removed

import warnings

warnings.simplefilter(action='ignore', category=UserWarning)

123. recursion is detected during loading of "cv2" binary extensions

import cv2

opencv-python==4.5.5.64

Link :

<https://automatetheboringstuff.com/chapter18/#:~:text=To%20do%20this%2C%20set%20the,seconds%20after%20performing%20its%20action>.

>>> **import pyautogui**

>>> **pyautogui.size()**

(1920, 1080)

>>> **width, height = pyautogui.size()**

>>> **import pyautogui**

>>> **pyautogui.PAUSE = 1**

>>> **pyautogui.FAILSAFE = True**

## Moving the Mouse

>>> **import pyautogui**

>>> **for i in range(10):**

**pyautogui.moveTo(100, 100, duration=0.25)**

**pyautogui.moveTo(200, 100, duration=0.25)**

**pyautogui.moveTo(200, 200, duration=0.25)**

**pyautogui.moveTo(100, 200, duration=0.25)**

## Getting the Mouse Position

>>> **pyautogui.position()**

(311, 622)

>>> **pyautogui.position()**

(377, 481)

>>> **pyautogui.position()**

(1536, 637)

## Clicking the Mouse

>>> **import pyautogui**

>>> **pyautogui.click(10, 5)**

## Dragging the Mouse

import pyautogui, time

❶ time.sleep(5)

❷ pyautogui.click() # click to put drawing program in focus

distance = 200

while distance > 0:

❸ pyautogui.dragRel(distance, 0, duration=0.2) # move right

❹ distance = distance - 5

❺ pyautogui.dragRel(0, distance, duration=0.2) # move down

❻ pyautogui.dragRel(-distance, 0, duration=0.2) # move left

distance = distance - 5

pyautogui.dragRel(0, -distance, duration=0.2) # move up

## Scrolling the Mouse

>>> **import time, pyautogui**

>>> **time.sleep(5); pyautogui.scroll(100)**

## Getting a Screenshot

>>> **import pyautogui**

>>> **im = pyautogui.screenshot()**

>>> **im.getpixel((0, 0))**

(176, 176, 175)

>>> **im.getpixel((50, 200))**

(130, 135, 144)

| **Keyboard key string** | **Meaning** |
| --- | --- |
| 'a', 'b', 'c', 'A', 'B', 'C', '1', '2', '3', '!', '@', '#', and so on | The keys for single characters |
| 'enter' (or 'return' or '\n') | The ENTER key |
| 'esc' | The ESC key |
| 'shiftleft', 'shiftright' | The left and right SHIFT keys |
| 'altleft', 'altright' | The left and right ALT keys |
| 'ctrlleft', 'ctrlright' | The left and right CTRL keys |
| 'tab' (or '\t') | The TAB key |
| 'backspace', 'delete' | The BACKSPACE and DELETE keys |
| 'pageup', 'pagedown' | The PAGE UP and PAGE DOWN keys |
| 'home', 'end' | The HOME and END keys |
| 'up', 'down', 'left', 'right' | The up, down, left, and right arrow keys |
| 'f1', 'f2', 'f3', and so on | The F1 to F12 keys |
| 'volumemute', 'volumedown', 'volumeup' | The mute, volume down, and volume up keys (some keyboards do not have these keys, but your operating system will still be able to understand these simulated keypresses) |
| 'pause' | The PAUSE key |
| 'capslock', 'numlock', 'scrolllock' | The CAPS LOCK, NUM LOCK, and SCROLL LOCK keys |
| 'insert' | The INS or INSERT key |
| 'printscreen' | The PRTSC or PRINT SCREEN key |
| 'winleft', 'winright' | The left and right WIN keys (on Windows) |
| 'command' | The Command () key (on OS X) 'option' The OPTION key (on OS X) |

>>> import pyautogui

>>> screenWidth, screenHeight = pyautogui.size() # Returns two integers, the width and height of the screen. (The primary monitor, in multi-monitor setups.)

>>> currentMouseX, currentMouseY = pyautogui.position() # Returns two integers, the x and y of the mouse cursor's current position.

>>> pyautogui.moveTo(100, 150) # Move the mouse to the x, y coordinates 100, 150.

>>> pyautogui.click() # Click the mouse at its current location.

>>> pyautogui.click(200, 220) # Click the mouse at the x, y coordinates 200, 220.

>>> pyautogui.move(None, 10) # Move mouse 10 pixels down, that is, move the mouse relative to its current position.

>>> pyautogui.doubleClick() # Double click the mouse at the

>>> pyautogui.moveTo(500, 500, duration=2, tween=pyautogui.easeInOutQuad) # Use tweening/easing function to move mouse over 2 seconds.

>>> pyautogui.write('Hello world!', interval=0.25) # Type with quarter-second pause in between each key.

>>> pyautogui.press('esc') # Simulate pressing the Escape key.

>>> pyautogui.keyDown('shift')

>>> pyautogui.write(['left', 'left', 'left', 'left', 'left', 'left'])

>>> pyautogui.keyUp('shift')

>>> pyautogui.hotkey('ctrl', 'c')

## Display Message Boxes

>>> import pyautogui

>>> pyautogui.alert('This is an alert box.')

'OK'

>>> pyautogui.confirm('Shall I proceed?')

'Cancel'

>>> pyautogui.confirm('Enter option.', buttons=['A', 'B', 'C'])

'B'

>>> pyautogui.prompt('What is your name?')

'Al'

>>> pyautogui.password('Enter password (text will be hidden)')

'swordfish'

## Screenshot Functions

>>> import pyautogui

>>> im1 = pyautogui.screenshot()

>>> im1.save('my\_screenshot.png')

>>> im2 = pyautogui.screenshot('my\_screenshot2.png')

You can also locate where an image is on the screen:

>>> import pyautogui

>>> button7location = pyautogui.locateOnScreen('button.png') # returns (left, top, width, height) of matching region

>>> button7location

(1416, 562, 50, 41)

>>> buttonx, buttony = pyautogui.center(button7location)

>>> buttonx, buttony

(1441, 582)

>>> pyautogui.click(buttonx, buttony) # clicks the center of where the button was found

settings = pyautogui.locateCenterOnScreen("settings.PNG")   
pyautogui.moveTo(settings)  
pyautogui.click()  
time.sleep(2)

**123. Press Down keys**

pydirectinput.press('down')

**124.** PyAutoGui also lets you press a button multiple times:

pyautogui.press('tab', presses=5) # press TAB five times in a row

pyautogui.press('A', presses=1000) # press A a thousand times in a row

125. press and release button

import keyboard

keyboard.press\_and\_release('anykey')

126. alt-tab or other tasks that require more than one key to be pressed at the same time:.

from pynput.keyboard import Key, Controller

    # Holds down the alt key

    keyboard = Controller()

    time.sleep(3)

    keyboard.press(Key.alt\_l)

    pydirectinput.press("s")

    pydirectinput.press("c")

    keyboard.release(Key.alt\_l)

import pyautogui

# Holds down the alt key

pyautogui.keyDown("alt")

# Presses the tab key once

pyautogui.press("tab")

# Lets go of the alt key

pyautogui.keyUp("alt")

127. Enter the data in field (PyAutoGui also lets you press a button multiple times) 😊

pip3 install keyboard

import keyboard

keyboard.write('A',delay=0)

pyautogui.press('tab', presses=5) # press TAB five times in a row

pyautogui.press('A', presses=1000) # press A a thousand times in a row

**128. Download Tesseract ( IMAGE Text)**

[**https://www.google.com/search?rlz=1C1CHBD\_enIN1014IN1014&lei=TQXuY6CbPIOYseMPn\_KWoAI&q=extract%20specific%20text%20from%20image%20python&ved=2ahUKEwjg9cTD6pn9AhUDTGwGHR-5BSQQsKwBKAB6BAhWEAE&biw=1280&bih=520&dpr=1.5#fpstate=ive&vld=cid:0717cc9b,vid:YK9a8E45X\_Y**](https://www.google.com/search?rlz=1C1CHBD_enIN1014IN1014&lei=TQXuY6CbPIOYseMPn_KWoAI&q=extract%20specific%20text%20from%20image%20python&ved=2ahUKEwjg9cTD6pn9AhUDTGwGHR-5BSQQsKwBKAB6BAhWEAE&biw=1280&bih=520&dpr=1.5#fpstate=ive&vld=cid:0717cc9b,vid:YK9a8E45X_Y)

[**https://stackoverflow.com/questions/49101270/move-to-searched-text-on-active-screen-with-pyautogui**](https://stackoverflow.com/questions/49101270/move-to-searched-text-on-active-screen-with-pyautogui)

[**https://github.com/UB-Mannheim/tesseract/wiki**](https://github.com/UB-Mannheim/tesseract/wiki)

**C:\Users\ssc.achauhan\AppData\Local\Programs\Tesseract-OCR**

**import pyautogui**

**import pytesseract**

**import cv2**

**import numpy as np**

**text\_to\_find = 'DEHAMDBUR'**

**# find your installation (Happened to me)**

**pytesseract.pytesseract.tesseract\_cmd = r'C:\Users\ssc.achauhan\AppData\Local\Programs\Tesseract-OCR\tesseract.exe'**

**time.sleep(4)**

**# Take a screenshot of the main screen**

**screenshot = pyautogui.screenshot()**

**# Convert the screenshot to grayscale**

**img = cv2.cvtColor(np.array(screenshot), cv2.COLOR\_RGB2GRAY)**

**time.sleep(4)**

**data = pytesseract.image\_to\_data(img, lang='eng', output\_type='data.frame')**

**if len(data.loc[data['text']=='DEHAMDBUR',['left','top']]) >0:**

**x,y = data.loc[data['text']=='DEHAMDBUR',['left','top']].values[0]**

**else:**

**print('Not Found')**

**print('Done')**

**129 Pytesseract not working sometimes on perfectly clear Images**

[**https://www.google.com/search?q=Pytesseract+not+working+sometimes+on+perfectly+clear+Images&rlz=1C1CHBD\_enIN1014IN1014&biw=1280&bih=577&tbm=vid&ei=sCnvY-K\_NZad4-EPwc2FqAg&ved=0ahUKEwji15yvgZz9AhWWzjgGHcFmAYUQ4dUDCA0&uact=5&oq=Pytesseract+not+working+sometimes+on+perfectly+clear+Images&gs\_lcp=Cg1nd3Mtd2l6LXZpZGVvEAMyBQgAEKIEMgUIABCiBDIFCAAQogQyBQgAEKIEMgUIABCiBFAAWABg7QdoAHAAeACAAcsBiAHLAZIBAzItMZgBAKABAqABAcABAQ&sclient=gws-wiz-video#fpstate=ive&vld=cid:87b31f69,vid:9nUNPrvCFAE**](https://www.google.com/search?q=Pytesseract+not+working+sometimes+on+perfectly+clear+Images&rlz=1C1CHBD_enIN1014IN1014&biw=1280&bih=577&tbm=vid&ei=sCnvY-K_NZad4-EPwc2FqAg&ved=0ahUKEwji15yvgZz9AhWWzjgGHcFmAYUQ4dUDCA0&uact=5&oq=Pytesseract+not+working+sometimes+on+perfectly+clear+Images&gs_lcp=Cg1nd3Mtd2l6LXZpZGVvEAMyBQgAEKIEMgUIABCiBDIFCAAQogQyBQgAEKIEMgUIABCiBFAAWABg7QdoAHAAeACAAcsBiAHLAZIBAzItMZgBAKABAqABAcABAQ&sclient=gws-wiz-video#fpstate=ive&vld=cid:87b31f69,vid:9nUNPrvCFAE)

**REFERENCESS::** [**https://morioh.com/p/6cf1028b3d03**](https://morioh.com/p/6cf1028b3d03) **( python project)**

[**https://stackoverflow.com/questions/44619077/pytesseract-ocr-multiple-config-options**](https://stackoverflow.com/questions/44619077/pytesseract-ocr-multiple-config-options)