

```
In [ ]: 1 #####
        2 ## Pandas Cheat Sheet compiled by Paschal Chukwuemeka Amah.##
        3 #####
```

```
In [ ]: 1 #Create an empty dataframe with three columns
        2 df = pd.DataFrame(columns = ["x", "y", "z"])
```

```
In [ ]: 1 #Quick summary of a dataframe: number of columns and rows
        2 df.shape
```

```
In [ ]: 1 #Import a comma seperated file
        2 df = pd.read_csv(filename.csv)
        3
        4 #Write a csv file
        5 pd.to_csv(filename.csv)
```

```
In [ ]: 1 #Import a space delimited file
        2 df = pd.read_csv("filename.txt", delim_whitespace=True)
```

```
In [ ]: 1 #Import an excel file
        2 xls = pd.ExcelFile("file.extension")
        3
        4 #Peep the excel sheets
        5 xls.sheet_names
        6
        7 #Load a sheet
        8 df = xls.parse("sheetName")
```

```
In [ ]: 1 #Write to excel
        2 from pandas import ExcelWriter
        3
        4 #Create a write object #Sample is the name of file written
        5 writer = ExcelWriter('sample.xlsx')
        6
        7 #Write to a sheet 'sheet1'
        8 df.to_excel(writer, 'Sheet1')
        9 writer.save()
```

```
In [ ]: 1 #Import a json file - especially one sourced from a social media like Twitter
        2 df = pd.read_jso("filename.json", lines = True, encoding = "utf-8")
```

```
In [ ]: 1 #####
        2 ## Clean up moves ##
        3 #####
```

```
In [ ]: 1 #Remove non-characters from column
        2 df['colName'] = df['colName'].lambda x: x.replace('[^a-zA-Z]', '')
```

```

In [ ]: 1 #Case conversion
        2
        3 #Convert whole dataframe to TitleCase
        4 df = df.apply(lambda x: x.astype(str).str.title())
        5
        6 #To lower case
        7 df = df.apply(lambda x: x.astype(str).str.lower())
        8
        9 #To UPPERCASE
       10 df = df.apply(lambda x: x.astype(str).str.upper())

In [ ]: 1 #Drop duplicates leaving first occurrence
        2 df.drop_duplicates(keep = 'first')

In [ ]: 1 #Replace parts of column content for all rows. Here, all items but the first
        2 df.loc[:, "colName"].map(lambda x: x.replace(x[4:], ''))

In [ ]: 1 #Replace parts of column content for all rows. Here, first 9 items are replaced
        2 df.loc[:, "colName"].map(lambda x: x.replace(x[:9], ''))

In [ ]: 1 #Replace parts of a column content for all rows. Here, all items but the first
        2 df.loc[:, "colName"].map(lambda x: x.replace(x[4:], '')+"abc")

In [ ]: 1 #Remove whitespaces in a column for all rows
        2 df.loc[:, "colName"].map(lambda x: "".join(x.split()))

In [ ]: 1 #Merge two columns as str into a new column
        2 df["NewColName"] = df["colName_x"].map(str) + df["colName_y"].map(str)

In [ ]: 1 #Pick only the first 11 columns for all rows
        2 df.iloc[:, :11]

In [ ]: 1 #Make a column of numbers from 0 to a particular range after importing numpy
        2 df["NewCol"] = np.arange(0, len(df))

In [ ]: 1 #Sort a dataframe according to a column in descending order and reset the index
        2 df.sort_values("colName", ascending = True).reset_index(drop=True)

In [ ]: 1 #Return parts of df that satisfy a condition in a column
        2 df[df["colName"] == criteria] # eg1 df[df["colName"] == 'lagos'] eg2 df[df["colName"] == 'lagos']

In [ ]: 1 #Drop part of a dataframe tail - 8 here
        2 df.drop(df.tail(8).index)

In [ ]: 1 #Drop part of a dataframe head - 8 here
        2 df.drop(df.head(8).index)

In [ ]: 1 #For a column, replace x ie row with y for all rows, if str in row/line, else
        2 df["colName"].map(lambda x: replace(x, y) if 'str' in x else x)

```

```
In [ ]: 1 #Group dataframe by a column and count
        2 df.groupby("colName").count()
        3
        4 #Group dataframe by 2 columns and count
        5 df.groupby(["colName1", "colName2"]).count()
```

```
In [ ]: 1 #Group dataframe by 2 columns, count and sort in descending order
        2 df.groupby("colName").count().sort_values("Other", ascending = False)
        3
        4 #Group dataframe by 2 columns, count and sort in descending order
        5 df.groupby(["colName1", "colName2"]).count().sort_values("Other", ascending =
```

```
In [ ]: 1 #Convert database column to List
        2 list1 = df.colName.tolist()
```