In [1]:
```
import numpy as np
import pandas as pd
```

In [2]:
```
data = xls = pd.ExcelFile("messy.xlsx")
```

In [3]:
```
messy = pd.read_excel(data, 'test')
```

In [4]:
```
messy
```

Out[4]:

	CUst ID	JOIN% DATE	Unnamed: 2	mobiles	FLL NAM
0	1.0	08/07/19	NaN	84333605993	mIChAel mIcHALek
1	2.0	09/10/18	NaN	973444062	Andrew Jimenez
2	3.0	20190630	NaN	338262954	Ann Gow
3	4.0	2019-09-06 00:00:1567702800	NaN	84966068026	James Chen
4	5.0	20170812	NaN	84767065885	Dollie Martinez
...	...	...	...	...	...
95	96.0	20190816	NaN	84334555439	Ron Grollimund
96	97.0	2017-08-22 00:00:1503334800	NaN	84972812359	Daniel Bentley
97	98.0	2018-02-17 00:00:1518800400	NaN	84865784802	Eugene Brown
98	99.0	20190612	NaN	841221343238	elMER milLER
99	100.0	20190822	NaN	1264626143	Frances Holland

100 rows × 5 columns

Clean the names of columns to lowercase separated by "_", remove any empty column if necessary

In [5]:
```
messy = messy.dropna(axis = 1)
```

In [6]:
```
messy
```

Out[6]:

	CUst ID	JOIN% DATE	mobiles	FLL NAM
0	1.0	08/07/19	84333605993	mIChAel mIcHALek
1	2.0	09/10/18	973444062	Andrew Jimenez
2	3.0	20190630	338262954	Ann Gow
3	4.0	2019-09-06 00:00:1567702800	84966068026	James Chen
4	5.0	20170812	84767065885	Dollie Martinez
...	...	...	...	...
95	96.0	20190816	84334555439	Ron Grollimund
96	97.0	2017-08-22 00:00:1503334800	84972812359	Daniel Bentley
97	98.0	2018-02-17 00:00:1518800400	84865784802	Eugene Brown
98	99.0	20190612	841221343238	elMER milLER
99	100.0	20190822	1264626143	Frances Holland

100 rows × 4 columns

In [7]:
```
messy.columns = ["cust_id", "join_date", "mobiles","full_name"]
```

In [8]:
```
messy
```

Out[8]:

	cust_id	join_date	mobiles	full_name
0	1.0	08/07/19	84333605993	mIChAel mIcHALek
1	2.0	09/10/18	973444062	Andrew Jimenez
2	3.0	20190630	338262954	Ann Gow
3	4.0	2019-09-06 00:00:1567702800	84966068026	James Chen
4	5.0	20170812	84767065885	Dollie Martinez
...	...	...	...	...
95	96.0	20190816	84334555439	Ron Grollimund
96	97.0	2017-08-22 00:00:1503334800	84972812359	Daniel Bentley
97	98.0	2018-02-17 00:00:1518800400	84865784802	Eugene Brown
98	99.0	20190612	841221343238	elMER milLER
99	100.0	20190822	1264626143	Frances Holland

100 rows × 4 columns

Change the date column to the same format ‘YYYY-MM-DD’

In [9]:

```
def simple_date (x):
    if len(x) > 20:
        date = x.split()[0]

    else:
        date = x

    return date
```

In [10]:

```
messy['simple_date'] = messy['join_date'].apply(simple_date)
```

In [11]:

```
messy
```

Out[11]:

	cust_id	join_date	mobiles	full_name	simple_date
0	1.0	08/07/19	84333605993	mIChAel mIcHALek	08/07/19
1	2.0	09/10/18	973444062	Andrew Jimenez	09/10/18
2	3.0	20190630	338262954	Ann Gow	20190630
3	4.0	2019-09-06 00:00:1567702800	84966068026	James Chen	2019-09-06
4	5.0	20170812	84767065885	Dollie Martinez	20170812
...	...	...	...	...	...
95	96.0	20190816	84334555439	Ron Grollimund	20190816
96	97.0	2017-08-22 00:00:1503334800	84972812359	Daniel Bentley	2017-08-22
97	98.0	2018-02-17 00:00:1518800400	84865784802	Eugene Brown	2018-02-17
98	99.0	20190612	841221343238	elMER milLER	20190612
99	100.0	20190822	1264626143	Frances Holland	20190822

100 rows × 5 columns

In [12]:

```
messy['join_date'] = pd.to_datetime(messy['simple_date'], errors='coerce')
```

In [13]:

```
messy
```

Out[13]:

	cust_id	join_date	mobiles	full_name	simple_date
0	1.0	2019-08-07	84333605993	mIChAel mIcHALek	08/07/19
1	2.0	2018-09-10	973444062	Andrew Jimenez	09/10/18
2	3.0	2019-06-30	338262954	Ann Gow	20190630
3	4.0	2019-09-06	84966068026	James Chen	2019-09-06
4	5.0	2017-08-12	84767065885	Dollie Martinez	20170812
...	...	...	...	...	...
95	96.0	2019-08-16	84334555439	Ron Grollimund	20190816
96	97.0	2017-08-22	84972812359	Daniel Bentley	2017-08-22
97	98.0	2018-02-17	84865784802	Eugene Brown	2018-02-17
98	99.0	2019-06-12	841221343238	elMER milLER	20190612
99	100.0	2019-08-22	1264626143	Frances Holland	20190822

100 rows × 5 columns

In [14]:

```
messy = messy.drop('simple_date', axis = 1)
```

In [15]:

```
messy
```

Out[15]:

	cust_id	join_date	mobiles	full_name
0	1.0	2019-08-07	84333605993	mIChAel mIcHALek
1	2.0	2018-09-10	973444062	Andrew Jimenez
2	3.0	2019-06-30	338262954	Ann Gow
3	4.0	2019-09-06	84966068026	James Chen
4	5.0	2017-08-12	84767065885	Dollie Martinez
...	...	...	...	...
95	96.0	2019-08-16	84334555439	Ron Grollimund
96	97.0	2017-08-22	84972812359	Daniel Bentley
97	98.0	2018-02-17	84865784802	Eugene Brown
98	99.0	2019-06-12	841221343238	elMER milLER
99	100.0	2019-08-22	1264626143	Frances Holland

100 rows × 4 columns

Change the name column to the title case

In [16]:

```
messy ['full_name'] = messy['full_name'].str.title()
```

In [17]:

```
messy
```

Out[17]:

	cust_id	join_date	mobiles	full_name
0	1.0	2019-08-07	84333605993	Michael Michalek
1	2.0	2018-09-10	973444062	Andrew Jimenez
2	3.0	2019-06-30	338262954	Ann Gow
3	4.0	2019-09-06	84966068026	James Chen
4	5.0	2017-08-12	84767065885	Dollie Martinez
...	...	...	...	...
95	96.0	2019-08-16	84334555439	Ron Grollimund
96	97.0	2017-08-22	84972812359	Daniel Bentley
97	98.0	2018-02-17	84865784802	Eugene Brown
98	99.0	2019-06-12	841221343238	Elmer Miller
99	100.0	2019-08-22	1264626143	Frances Holland

100 rows × 4 columns

Make a new “email” column with form: {last_name}.{first_name}.{id}@yourcompany.com

In [24]:

```
messy['first_name'] = messy['full_name'].apply(lambda x: x.split()[0])
```

In [25]:

```
messy['last_name'] = messy['full_name'].apply(lambda x: x.split()[1])
```

In [26]:

```
messy['email'] = messy['last_name'] + '.' +\
                 messy['first_name'] + '.' +\
                 messy['cust_id'].astype(int).astype(str) +\
                 '@yourcompany.com'
```

In [27]:

```
messy
```

Out[27]:

	cust_id	join_date	mobiles	full_name	email	first_name	la
0	1.0	2019-08-07	84333605993	Michael Michalek	Michalek.Michael.1@yourcompany.com	Michael	
1	2.0	2018-09-10	973444062	Andrew Jimenez	Jimenez.Andrew.2@yourcompany.com	Andrew	
2	3.0	2019-06-30	338262954	Ann Gow	Gow.Ann.3@yourcompany.com	Ann	
3	4.0	2019-09-06	84966068026	James Chen	Chen.James.4@yourcompany.com	James	
4	5.0	2017-08-12	84767065885	Dollie Martinez	Martinez.Dollie.5@yourcompany.com	Dollie	
...	...	...	...	...	...	...	
95	96.0	2019-08-16	84334555439	Ron Grollimund	Grollimund.Ron.96@yourcompany.com	Ron	G
96	97.0	2017-08-22	84972812359	Daniel Bentley	Bentley.Daniel.97@yourcompany.com	Daniel	
97	98.0	2018-02-17	84865784802	Eugene Brown	Brown.Eugene.98@yourcompany.com	Eugene	
98	99.0	2019-06-12	841221343238	Elmer Miller	Miller.Elmer.99@yourcompany.com	Elmer	
99	100.0	2019-08-22	1264626143	Frances Holland	Holland.Frances.100@yourcompany.com	Frances	

100 rows × 7 columns

In [28]:

```
messy = messy.drop(['first_name','last_name'], axis = 1)
```

In [29]:

```
messy
```

Out[29]:

	cust_id	join_date	mobiles	full_name	email
0	1.0	2019-08-07	84333605993	Michael Michalek	Michalek.Michael.1@yourcompany.com
1	2.0	2018-09-10	973444062	Andrew Jimenez	Jimenez.Andrew.2@yourcompany.com
2	3.0	2019-06-30	338262954	Ann Gow	Gow.Ann.3@yourcompany.com
3	4.0	2019-09-06	84966068026	James Chen	Chen.James.4@yourcompany.com
4	5.0	2017-08-12	84767065885	Dollie Martinez	Martinez.Dollie.5@yourcompany.com
...	...	...	...	...	...
95	96.0	2019-08-16	84334555439	Ron Grollimund	Grollimund.Ron.96@yourcompany.com
96	97.0	2017-08-22	84972812359	Daniel Bentley	Bentley.Daniel.97@yourcompany.com
97	98.0	2018-02-17	84865784802	Eugene Brown	Brown.Eugene.98@yourcompany.com
98	99.0	2019-06-12	841221343238	Elmer Miller	Miller.Elmer.99@yourcompany.com
99	100.0	2019-08-22	1264626143	Frances Holland	Holland.Frances.100@yourcompany.com

100 rows × 5 columns

Change the phone number column to the format “84......”

In [30]:

```
def correct_mobile_number (y):

    prefix = '84'
    mobile_number = ''

    if len(y) == 9 or len(y) == 10:
        mobile_number = prefix + y

    else:
        mobile_number = y

    return mobile_number
```

In [31]:

```
messy['mobiles'] = messy['mobiles'].astype(str)\
                                    .apply(correct_mobile_number)
```

In [32]:

```
messy
```

Out[32]:

	cust_id	join_date	mobiles	full_name	email
0	1.0	2019-08-07	84333605993	Michael Michalek	Michalek.Michael.1@yourcompany.com
1	2.0	2018-09-10	84973444062	Andrew Jimenez	Jimenez.Andrew.2@yourcompany.com
2	3.0	2019-06-30	84338262954	Ann Gow	Gow.Ann.3@yourcompany.com
3	4.0	2019-09-06	84966068026	James Chen	Chen.James.4@yourcompany.com
4	5.0	2017-08-12	84767065885	Dollie Martinez	Martinez.Dollie.5@yourcompany.com
...	...	...	...	...	...
95	96.0	2019-08-16	84334555439	Ron Grollimund	Grollimund.Ron.96@yourcompany.com
96	97.0	2017-08-22	84972812359	Daniel Bentley	Bentley.Daniel.97@yourcompany.com
97	98.0	2018-02-17	84865784802	Eugene Brown	Brown.Eugene.98@yourcompany.com
98	99.0	2019-06-12	841221343238	Elmer Miller	Miller.Elmer.99@yourcompany.com
99	100.0	2019-08-22	841264626143	Frances Holland	Holland.Frances.100@yourcompany.com

100 rows × 5 columns

Find any duplicated ID and remove those who join later.

In [33]:

```
messy = messy.sort_values(by=['join_date', 'cust_id'])
```

In [34]:

```
messy.loc[messy.duplicated(subset = 'cust_id') == True, :]
```

Out[34]:

	cust_id	join_date	mobiles	full_name	email
13	14.0	2017-12-10	84356173988	Paul Ruper	Ruper.Paul.14@yourcompany.com
5	21.0	2018-06-02	84339769174	Roger Callender	Callender.Roger.21@yourcompany.com
68	69.0	2018-09-18	841289086592	Sharon Harris	Harris.Sharon.69@yourcompany.com
8	9.0	2018-12-01	84782904001	James Demers	Demers.James.9@yourcompany.com
40	41.0	2019-01-01	841267025689	Elizabeth Herrington	Herrington.Elizabeth.41@yourcompany.com
23	24.0	2019-04-04	84766048406	Juanita Nutter	Nutter.Juanita.24@yourcompany.com
72	86.0	2019-06-11	84865639641	Yvette Hayden	Hayden.Yvette.86@yourcompany.com
32	33.0	2019-07-23	84974404339	Lawrence Cummings	Cummings.Lawrence.33@yourcompany.com
24	97.0	2019-09-02	84978909747	Stephanie Short	Short.Stephanie.97@yourcompany.com
65	66.0	2019-12-09	84869366677	Florence Luallen	Luallen.Florence.66@yourcompany.com

In [35]:

```
messy = messy.loc[messy.duplicated(subset = 'cust_id') == False, :]
```

In [36]:

```
messy
```

Out[36]:

	cust_id	join_date	mobiles	full_name	email
35	36.0	2017-02-10	84339661824	Gregory Miele	Miele.Gregory.36@yourcompany.com
37	38.0	2017-08-02	84344179399	Melvin Pigg	Pigg.Melvin.38@yourcompany.com
25	26.0	2017-08-07	84966042581	Charles Garnand	Garnand.Charles.26@yourcompany.com
4	5.0	2017-08-12	84767065885	Dollie Martinez	Martinez.Dollie.5@yourcompany.com
96	97.0	2017-08-22	84972812359	Daniel Bentley	Bentley.Daniel.97@yourcompany.com
...	...	...	...	...	...
3	4.0	2019-09-06	84966068026	James Chen	Chen.James.4@yourcompany.com
12	13.0	2019-09-15	84971262200	Kim Biddle	Biddle.Kim.13@yourcompany.com
19	20.0	2019-09-23	84352351854	Barbara Daugherty	Daugherty.Barbara.20@yourcompany.com
10	11.0	2019-10-07	84973180839	Otis Arnold	Arnold.Otis.11@yourcompany.com
91	92.0	2019-11-05	841267874536	Patricia Coon	Coon.Patricia.92@yourcompany.com

90 rows × 5 columns

Filter those who join since 2019 and export to a csv file, delimited by “|”, file name “emp_{report_date}.csv” with report_date = today.

In [37]:

```
report = messy.loc[messy['join_date'] >= '2019-01-01' , :]
```

In [38]:

```
from datetime import datetime
report_date = str(datetime.today()).split()[0]

report.to_csv(f"emp_{report_date}.csv", index=False)
```