In [4]:

```
import pandas as pd
```

In [5]:

```
data = pd.read_csv(″D:\KSR\Python\ExcelFile\EmployeeDataToClean.csv″,encoding = ’ISO-8859-1
```

In [6]:

```
data.shape
```

Out[6]:

```
(50, 14)
```

In [7]:

```
data.head(3)
```

Out[7]:

	EmployeeID	NationalIDNumber	LoginID
0	1	14417807	adventure-works\guy1
1	2	253022876	adventure-works\kevin0
2	3	509647174	NaN

In [8]:

```
#is this 15 columns really needed?   #object --> need to share clean data to Data Analyst f
```

In [9]:

```
#7418408634 ----> mask ---> 741xxxxx34
```

In [10]:

```
list(data.columns)
```

Out[10]:

```
[’EmployeeID’,
 ’NationalIDNumber’,
 ’LoginID’,
 ’Title’,
 ’PhoneNumber’,
 ’BirthDate’,
 ’MaritalStatus’,
 ’Gender’,
 ’HireDate’,
 ’Dept’,
 'Salary',
 'Job Grade',
 'CurrentFlag',
 'rowguid']
```

In [11]:

```
data.isna().sum()
```

Out[11]:

```
EmployeeID           0
NationalIDNumber     0
LoginID             37
Title                0
PhoneNumber          7
BirthDate            0
MaritalStatus        0
Gender               0
HireDate             0
Dept                 0
Salary               6
Job Grade            0
CurrentFlag          0
rowguid              0
dtype: int64
```

In [12]:

```
(data.isna().sum() / data.shape[0]) *100
```

Out[12]:

```
EmployeeID           0.0
NationalIDNumber     0.0
LoginID             74.0
Title                0.0
PhoneNumber         14.0
BirthDate            0.0
MaritalStatus        0.0
Gender               0.0
HireDate             0.0
Dept                 0.0
Salary              12.0
Job Grade            0.0
CurrentFlag          0.0
rowguid              0.0
dtype: float64
```

In [13]:

```
data[’PhoneNumber’] = data[’PhoneNumber’].fillna(0)
```

In [14]:

```
data[’PhoneNumber’] = data[’PhoneNumber’].astype(’int64’)   #this is called typecasting
```

In [15]:

```
data.dtypes
```

Out[15]:

```
EmployeeID             int64
NationalIDNumber       int64
LoginID               object
Title                 object
PhoneNumber            int64
BirthDate             object
MaritalStatus         object
Gender                object
HireDate              object
Dept                  object
Salary               float64
Job Grade             object
CurrentFlag            int64
rowguid               object
dtype: object
```

In [16]:

```
data.head(5)
```

Out[16]:

	EmployeeID	NationalIDNumber	LoginID
0	1	14417807	adventure-works\guy1
1	2	253022876	adventure-works\kevin0
2	3	509647174	NaN
3	4	112457891	NaN
4	5	480168528	NaN

In [17]:

```
#7418408634 ----> mask ---> 741xxxxx34
```

In [18]:

```
data[’PhoneNumber’] = data[’PhoneNumber’].astype(’str’)
```

```
In  [19]:

a = '7418408634’

In  [20]:

a[-2:]

Out[20]:

'34’

In  [21]:

a[0:3]

Out[21]:

'741’

In  [22]:

a[0:3]+'XXXXX’+a[-2:]

Out[22]:

'741XXXXX34’

In  [23]:

def phno_masking(a):
    if a == '0’:
        return ″XXXXXXXXXX″
    else:
        result = a[0:3]+'XXXXX’+a[-2:]
        return result

In  [24]:

phno_masking('0’)

Out[24]:

'XXXXXXXXXX’

In  [25]:

phno_masking('7654325467’)
```

Out[25]:

```
’765XXXXX67’
```

In [26]:

```
data[’PhoneNumber’] = data[’PhoneNumber’].apply(phno_masking)
```

In [27]:

```
data.head(5)
```

Out[27]:

	EmployeeID	NationalIDNumber	LoginID
0	1	14417807	adventure-works\guy1
1	2	253022876	adventure-works\kevin0
2	3	509647174	NaN
3	4	112457891	NaN
4	5	480168528	NaN

In [28]:

```
def marital_status(a):
   if a == ’M’:
           return ″Married″
   else:
       return ″Single″
```

In [29]:

```
def gender(a):
    if a == ’M’:
        return ’Male’
    else:
        return ’Female’
```

In [30]:

```
data[’MaritalStatus’] = data[’MaritalStatus’].apply(marital_status)
data[’Gender’] = data[’Gender’].apply(gender)
```

In [31]:

```
data.head(5)
```

Out[31]:

	EmployeeID	NationalIDNumber	LoginID
0	1	14417807	adventure-works\guy1
1	2	253022876	adventure-works\kevin0

2	3	509647174	NaN
3	4	112457891	NaN
4	5	480168528	NaN

In [32]:

```
data.shape
```

Out[32]:

```
(50, 14)
```

In [33]:

```
data.shape[0]
```

Out[33]:

```
                  50
```

In [34]:

```
data['EmployeeID'].count() #total emp count
```

Out[34]:

```
np.int64(50)
```

In [35]:

```
data['EmployeeID'].nunique()  #total emp distinct count
```

Out[35]:

```
                  50
```

In [36]:

```
data['NationalIDNumber'].nunique()  #this column is not useful bcz we already employeeid as
```

Out[36]:

```
                  50
```

In [37]:

```
#data cleaning steps
#1.Remove nationalIDNumber  #this column is not useful bcz we already employeeid as unique
#2. Remove LoginID  #because this has 74% missing values
#3.Split title as fistname and lastname (u can drop th TITLE column)
```

```
#4.fillna for sales by average sales
#5.Remove rowguid(as this is same randam id, also this is not useful,bcz we already employe
#6.remove currentflag(everyone have same value)
```

In [38]:

```
data[[’Title’]].head(10)
```

Out[38]:

	Title
0	Gustavo Achong
1	Catherine Abel
2	Kim Abercrombie
3	Humberto Acevedo
4	Pilar Ackerman
5	Frances Adams
6	Margaret Smith
7	Carla Adams
8	Jay Adams
9	Ronald Adina

In [39]:

```
a = ’Ragavi Pandi’
```

In [40]:

```
type(a)
```

Out[40]:

```
str
```

In [41]:

```
a.split()
```

Out[41]:

```
[’Ragavi’, ’Pandi’]
```

In [42]:

```
FN =a.split()[0]
LN =a.split()[1]
```

In [43]:

```
FN
```

Out[43]:

```
'Ragavi’
```

In [44]:

```
LN
```

Out[44]:

```
'Pandi’
```

In [45]:

```
data[['Fistname','Lastname']] = data['Title'].str.split(' ',n=1,expand= True)
```

In [46]:

```
data.head(5)
```

Out[46]:

	EmployeeID	NationalIDNumber	LoginID
0	1	14417807	adventure-works\guy1
1	2	253022876	adventure-works\kevin0
2	3	509647174	NaN
3	4	112457891	NaN
4	5	480168528	NaN

In [47]:

```
data.isna().sum() / data.shape[0]
```

Out[47]:

```
EmployeeID          0.00
NationalIDNumber    0.00
LoginID             0.74
Title               0.00
PhoneNumber         0.00
BirthDate           0.00
MaritalStatus       0.00
Gender              0.00
HireDate            0.00
Dept                0.00
Salary              0.12
Job Grade           0.00
CurrentFlag         0.00
rowguid             0.00
Fistname            0.00
Lastname            0.00
```

```
dtype: float64
```

In [48]:

```
data[['Salary']].mean()
```

Out[48]:

```
Salary    2563.727273
dtype: float64
```

In [49]:

```
data[['Salary']].median()
```

Out[49]:

```
Salary    2417.0
dtype: float64
```

In [50]:

```
data[['Salary']].min()
```

Out[50]:

```
Salary    548.0
dtype: float64
```

In [51]:

```
data[['Salary']].max()
```

Out[51]:

```
Salary    4547.0
dtype: float64
```

In [52]:

```
data.groupby(['Dept'])[['Salary']].mean()
```

Out[52]:

	Salary
Dept	
Finance	2715.454545
Human Resource	2598.333333
Logistics	3156.5
Production	2379.615385

Sales	2005.6
sales	2480.666667

In [53]:

```
list(data['Dept'].unique())
```

Out[53]:

```
['Sales', 'Finance', 'Logistics', 'Human Resource', 'sales', 'Production']
```

In [54]:

```
data['Dept'] = data['Dept'].str.capitalize()
```

In [55]:

```
data[data['Salary'].isna()]
```

Out[55]:

	EmployeeID	NationalIDNumber	LoginID
5	6	24756624	NaN
6	7	309738752	NaN
41	42	441044382	NaN
42	43	718299860	NaN
43	44	685233686	NaN
44	45	295971920	NaN

In [56]:

```
data['Salary'] = data['Salary'].fillna(data[['Salary']].mean())
```

In [57]:

```
data['Job Grade'].unique()
```

Out[57]:

```
array(['Admin', 'Management', 'Operations'], dtype=object)
```

In [58]:

```
data['rowguid'].nunique()
```

Out[58]:

```
                 50
```

In [59]:

```
data[['CurrentFlag']].head(5)
```

Out[59]:

	CurrentFlag
0	-1
1	-1
2	-1
3	-1
4	-1

In [60]:

```
data['CurrentFlag'].unique()
```

Out[60]:

```
array([-1])
```

In [61]:

```
#drop columns
```

In [62]:

```
data.shape
```

Out[62]:

```
(50, 16)
```

In [63]:

```
data = data.drop(['CurrentFlag','Title','rowguid','LoginID','NationalIDNumber'], axis = 1)
```

In [64]:

```
data.head(5)
```

Out[64]:

	EmployeeID	PhoneNumber	BirthDate
0	1	925XXXXX51	21-02-1986 00:00
1	2	923XXXXX60	12-03-1991 00:00
2	3	941XXXXX59	21-09-1978 00:00
3	4	880XXXXX25	01-11-1978 00:00
4	5	XXXXXXXXXX	07-06-1963 00:00

In [65]:

```
data.shape
```

Out[65]:

```
(50, 11)
```

In [66]:

```
data['Salary'] = data['Salary'].fillna(data['Salary'].mean())
```

In [67]:

```
data.isna().sum()
```

Out[67]:

```
EmployeeID       0
PhoneNumber      0
BirthDate        0
MaritalStatus    0
Gender           0
HireDate         0
Dept             0
Salary           0
Job Grade        0
Fistname         0
Lastname         0
dtype: int64
```

In [68]:

```
data.columns
```

Out[68]:

```
Index(['EmployeeID', 'PhoneNumber', 'BirthDate', 'MaritalStatus', 'Gender',
       'HireDate', 'Dept', 'Salary', 'Job Grade', 'Fistname', 'Lastname'],
      dtype='object')
```

In [69]:

```
data = data[[ 'EmployeeID','Fistname', 'Lastname', 'PhoneNumber', 'BirthDate', 'MaritalStat
       'HireDate', 'Dept', 'Salary', 'Job Grade']]
```

In [70]:

```
data.head(5)
```

Out[70]:

	EmployeeID	Fistname	Lastname
0	1	Gustavo	Achong
1	2	Catherine	Abel
2	3	Kim	Abercrombie

3	4	Humberto	Acevedo
4	5	Pilar	Ackerman

In [ ]:

')

Title	PhoneNumber	BirthDate	MaritalStatus	Gender
Gustavo Achong	9.26E+09	21-02-1986 00:00	M	M
Catherine Abel	9.24E+09	12-03-1991 00:00	S	M
Kim Abercrombie	9.42E+09	21-09-1978 00:00	M	M

'or reporting

Title	PhoneNumber	BirthDate	MaritalStatus	Gender
Gustavo Achong	9257522351	21-02-1986 00:00	M	M
Catherine Abel	9235868360	12-03-1991 00:00	S	M
Kim Abercrombie	9416421559	21-09-1978 00:00	M	M
Humberto Acevedo	8800042425	01-11-1978 00:00	S	M
Pilar Ackerman	0	07-06-1963 00:00	M	M

Title	PhoneNumber	BirthDate	MaritalStatus	Gender
Gustavo Achong	925XXXXX51	21-02-1986 00:00	M	M
Catherine Abel	923XXXXX60	12-03-1991 00:00	S	M
Kim Abercrombie	941XXXXX59	21-09-1978 00:00	M	M
Humberto Acevedo	880XXXXX25	01-11-1978 00:00	S	M
Pilar Ackerman	XXXXXXXXXX	07-06-1963 00:00	M	M

Title	PhoneNumber	BirthDate	MaritalStatus	Gender
Gustavo Achong	925XXXXX51	21-02-1986 00:00	Married	Male
Catherine Abel	923XXXXX60	12-03-1991 00:00	Single	Male

Kim Abercrombie	941XXXXX59	21-09-1978 00:00	Married	Male
Humberto Acevedo	880XXXXX25	01-11-1978 00:00	Single	Male
Pilar Ackerman	XXXXXXXXXX	07-06-1963 00:00	Married	Male

unique identifier

identifier

eid as unique identifier)

Title	PhoneNumber	BirthDate	MaritalStatus	Gender
Gustavo Achong	925XXXXX51	21-02-1986 00:00	Married	Male
Catherine Abel	923XXXXX60	12-03-1991 00:00	Single	Male
Kim Abercrombie	941XXXXX59	21-09-1978 00:00	Married	Male
Humberto Acevedo	880XXXXX25	01-11-1978 00:00	Single	Male
Pilar Ackerman	XXXXXXXXXX	07-06-1963 00:00	Married	Male

Title	PhoneNumber	BirthDate	MaritalStatus	Gender
Frances Adams	XXXXXXXXXX	26-01-1979 00:00	Single	Male
Margaret Smith	XXXXXXXXXX	25-11-1959 00:00	Single	Female
Chris Ashton	907XXXXX61	22-10-1979 00:00	Single	Female
Teresa Atkinson	835XXXXX69	04-10-1976 00:00	Married	Male
John Ault	900XXXXX52	27-07-1972 00:00	Single	Male
Robert Avalos	921XXXXX79	04-05-1993 00:00	Single	Male

MaritalStatus	Gender	HireDate	Dept	Salary
Married	Male	02-02-2013 00:00	Sales	2295
Single	Male	31-08-2013 00:00	Sales	962
Married	Male	16-06-2014 00:00	Finance	4006
Single	Male	10-07-2014 00:00	Logistics	4547
Married	Male	16-07-2014 00:00	Human resource	1932

```
us’, ’Gender’,
```

PhoneNumber	BirthDate	MaritalStatus	Gender	HireDate
925XXXXX51	21-02-1986 00:00	Married	Male	02-02-2013 00:00
923XXXXX60	12-03-1991 00:00	Single	Male	31-08-2013 00:00
941XXXXX59	21-09-1978 00:00	Married	Male	16-06-2014 00:00

880XXXXX25	01-11-1978 00:00	Single	Male	10-07-2014 00:00
XXXXXXXXXX	07-06-1963 00:00	Married	Male	16-07-2014 00:00

HireDate	Dept	Salary	Job Grade	CurrentFlag
02-02-2013 00:00	Sales	2295	Admin	-1
31-08-2013 00:00	Sales	962	Management	-1
16-06-2014 00:00	Finance	4006	Admin	-1

HireDate	Dept	Salary	Job Grade	CurrentFlag
02-02-2013 00:00	Sales	2295	Admin	-1
31-08-2013 00:00	Sales	962	Management	-1
16-06-2014 00:00	Finance	4006	Admin	-1
10-07-2014 00:00	Logistics	4547	Admin	-1
16-07-2014 00:00	Human Resource	1932	Admin	-1

HireDate	Dept	Salary	Job Grade	CurrentFlag
02-02-2013 00:00	Sales	2295	Admin	-1
31-08-2013 00:00	Sales	962	Management	-1
16-06-2014 00:00	Finance	4006	Admin	-1
10-07-2014 00:00	Logistics	4547	Admin	-1
16-07-2014 00:00	Human Resource	1932	Admin	-1

HireDate	Dept	Salary	Job Grade	CurrentFlag
02-02-2013 00:00	Sales	2295	Admin	-1
31-08-2013 00:00	Sales	962	Management	-1

16-06-2014 00:00	Finance	4006	Admin	-1
10-07-2014 00:00	Logistics	4547	Admin	-1
16-07-2014 00:00	Human Resource	1932	Admin	-1

HireDate	Dept	Salary	Job Grade	CurrentFlag
02-02-2013 00:00	Sales	2295	Admin	-1
31-08-2013 00:00	Sales	962	Management	-1
16-06-2014 00:00	Finance	4006	Admin	-1
10-07-2014 00:00	Logistics	4547	Admin	-1
16-07-2014 00:00	Human Resource	1932	Admin	-1

HireDate	Dept	Salary	Job Grade	CurrentFlag
25-07-2014 00:00	Sales	NaN	Management	-1
31-07-2014 00:00	Sales	NaN	Operations	-1
17-07-2015 00:00	Logistics	NaN	Admin	-1
18-07-2018 00:00	Logistics	NaN	Admin	-1
18-07-2015 00:00	Sales	NaN	Operations	-1
18-07-2017 00:00	Production	NaN	Admin	-1

Job Grade	Fistname	Lastname
Admin	Gustavo	Achong
Management	Catherine	Abel
Admin	Kim	Abercrombie
Admin	Humberto	Acevedo
Admin	Pilar	Ackerman

Dept	Salary	Job Grade
Sales	2295	Admin
Sales	962	Management
Finance	4006	Admin

Logistics	4547	Admin
Human resource	1932	Admin

rowguid
{AAE1D04A-C237-4974-B4D5-935247737718}
{1B480240-95C0-410F-A717-EB29943C8886}
{9BBBFB2C-EFBB-4217-9AB7-F97689328841}

rowguid
{AAE1D04A-C237-4974-B4D5-935247737718}
{1B480240-95C0-410F-A717-EB29943C8886}
{9BBBFB2C-EFBB-4217-9AB7-F97689328841}
{59747955-87B8-443F-8ED4-F8AD3AFDF3A9}
{1D955171-E773-4FAD-8382-40FD898D5D4D}

rowguid
{AAE1D04A-C237-4974-B4D5-935247737718}
{1B480240-95C0-410F-A717-EB29943C8886}
{9BBBFB2C-EFBB-4217-9AB7-F97689328841}
{59747955-87B8-443F-8ED4-F8AD3AFDF3A9}
{1D955171-E773-4FAD-8382-40FD898D5D4D}

rowguid
{AAE1D04A-C237-4974-B4D5-935247737718}
{1B480240-95C0-410F-A717-EB29943C8886}

{9BBBFB2C-EFBB-4217-9AB7-F97689328841}
{59747955-87B8-443F-8ED4-F8AD3AFDF3A9}
{1D955171-E773-4FAD-8382-40FD898D5D4D}

rowguid	Fistname	Lastname
{AAE1D04A-C237-4974-B4D5-935247737718}	Gustavo	Achong
{1B480240-95C0-410F-A717-EB29943C8886}	Catherine	Abel
{9BBBFB2C-EFBB-4217-9AB7-F97689328841}	Kim	Abercrombie
{59747955-87B8-443F-8ED4-F8AD3AFDF3A9}	Humberto	Acevedo
{1D955171-E773-4FAD-8382-40FD898D5D4D}	Pilar	Ackerman

rowguid	Fistname	Lastname
{E87029AA-2CBA-4C03-B948-D83AF0313E28}	Frances	Adams
{2CC71B96-F421-485E-9832-8723337749BB}	Margaret	Smith
{794A0B1F-C46A-401C-984D-008996FC7092}	Chris	Ashton
{6B10192F-D570-47C4-82C9-3D979B1EFDC1}	Teresa	Atkinson
{13909262-4136-492F-BCA3-0B0E3773B03E}	John	Ault
{45358AE8-0B0E-4C11-90BB-DAC3EC0D5C82}	Robert	Avalos