Data Wrangling

In [43]:
```
## Import pandas library
import pandas as pd
```

In [2]:
```
## Read CSV
data = pd.read_csv("dirtydata - dirtydata.csv")
data
```

Out[2]:

	Duration	Date	Pulse	Maxpulse	Calories
0	60	2020/12/01'	110	130	409.1
1	60	2020/12/02'	117	145	479.0
2	60	2020/12/03'	103	135	340.0
3	45	2020/12/04'	109	175	282.4
4	45	2020/12/05'	117	148	406.0
5	60	2020/12/06'	102	127	-300.0
6	60	2020/12/07'	110	136	374.0
7	450	2020/12/08'	104	134	253.3
8	30	2020/12/09'	109	133	195.1
9	60	2020/12/10'	98	124	269.0
10	60	2020/12/11'	103	147	329.3
11	60	2020/12/12'	100	120	250.7
12	60	2020/12/12'	100	120	250.7
13	60	2020/12/13'	106	128	345.3
14	60	2020/12/14'	104	132	379.3
15	60	2020/12/15'	98	123	275.0
16	60	2020/12/16'	98	120	215.2
17	60	2020/12/17'	100	120	300.0
18	45	2020/12/18'	90	112	NaN
19	60	2020/12/19'	103	123	323.0
20	45	2020/12/20'	97	125	243.0
21	60	2020/12/21'	108	131	364.2
22	45	NaN	100	119	282.0
23	60	2020/12/23'	130	101	300.0
24	45	2020/12/24'	105	132	246.0
25	60	2020/12/25'	102	126	334.5
26	60	20201226	100	120	250.0
27	60	2020/12/27'	92	118	241.0
28	60	2020/12/28'	103	132	NaN
29	60	2020/12/29'	100	132	-280.0
30	60	2020/12/30'	102	129	380.3

31	60	2020/12/31'	92	115	243.0

In [3]:
```
## Return 1st 5 elements
data.head()
```

Out[3]:

	Duration	Date	Pulse	Maxpulse	Calories
0	60	2020/12/01'	110	130	409.1
1	60	2020/12/02'	117	145	479.0
2	60	2020/12/03'	103	135	340.0
3	45	2020/12/04'	109	175	282.4
4	45	2020/12/05'	117	148	406.0

In [4]:
```
data.shape
```

Out[4]:
```
(32, 5)
```

In [5]:
```
## Data statistics
data.describe()
```

Out[5]:

	Duration	Pulse	Maxpulse	Calories
count	32.000000	32.000000	32.000000	30.000000
mean	68.437500	103.500000	128.500000	266.013333
std	70.039591	7.832933	12.998759	164.876415
min	30.000000	90.000000	101.000000	-300.000000
25%	60.000000	100.000000	120.000000	247.000000
50%	60.000000	102.500000	127.500000	282.200000
75%	60.000000	106.500000	132.250000	343.975000
max	450.000000	130.000000	175.000000	479.000000

In [6]:
```
## check no. of null values in each column
data.isnull().sum()
```

Out[6]:
```
Duration    0
Date        1
Pulse       0
Maxpulse    0
Calories    2
dtype: int64
```

In [7]:
```
data.count()
#data.count(axis = 'rows')
```

Out[7]:
```
Duration    32
Date        31
Pulse       32
Maxpulse    32
Calories    30
dtype: int64
```

In [8]:
```
data.dtypes
```

Out[8]:
```
Duration      int64
Date         object
Pulse         int64
```

```
Maxpulse      int64
Calories    float64
dtype: object
```

In [9]:
```
## Replace values with it's absolute
data['Calories'] = data['Calories'].abs()
```

In [10]:
```
## Mean of the column
x = data.Calories.mean()
x
```

Out[10]:
```
304.68
```

In [11]:
```
## Absolute
data['Calories'] = data['Calories'].abs()
```

In [12]:
```
## Replace null values with mean
data['Calories'].fillna(x,inplace = True)
```

In [13]:
```
data.isnull().sum()
```

Out[13]:
```
Duration    0
Date        1
Pulse       0
Maxpulse    0
Calories    0
dtype: int64
```

In [15]:
```
data.dropna(subset = ["Date"],inplace=True)
```

In [16]:
```
data.isnull().sum()
```

Out[16]:
```
Duration    0
Date        0
Pulse       0
Maxpulse    0
Calories    0
dtype: int64
```

In [17]:
```
data['Calories'] = data['Calories'].astype(int)
```

In [18]:
```
data['Date'] = pd.to_datetime(data['Date'])
```

In [19]:
```
data.loc[7,'Duration']
```

Out[19]:
```
450
```

In [20]:
```
## TO find minimun
data['Duration'].min()
```

Out[20]:
```
30
```

In [21]:
```
## TO find maximum
data['Duration'].max()
```

Out[21]:
```
450
```

In [31]:
```
# for i in data.Duration:
#     if i > 60 and i < 45:
#         print(i)
#         data.loc[i,'Duration']=45
```

In [39]:
```
# data.loc[(data['Duration'] > 60 and data['Duration'] < 45).item(), 'Duration'] = 45
```

In [40]:
```
data.loc[7,'Duration'] = 45
```

In [41]:
```
data.duplicated()
```

Out[41]:
```
0     False
1     False
2     False
3     False
4     False
5     False
6     False
7     False
8     False
9     False
10    False
11    False
12     True
13    False
14    False
15    False
16    False
17    False
18    False
19    False
20    False
21    False
23    False
24    False
25    False
26    False
27    False
28    False
29    False
30    False
31    False
dtype: bool
```

In [42]:
```
## drop duplicated rows
data.drop_duplicates(inplace = True)
```

In [84]:
```
data.to_csv('Dirty_data_preprop')
```

NBA Data

In [47]:
```
data_nba = pd.read_csv("nba.csv")
```

In [48]:
```
data_nba
```

Out[48]:

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0	PG	25	2-Jun	180	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99	SF	25	6-Jun	235	Marquette	6796117.0
2	John Holland	Boston Celtics	30	SG	27	5-Jun	205	Boston University	NaN
3	R.J. Hunter	Boston Celtics	28	SG	22	5-Jun	185	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8	PF	29	10-Jun	231	NaN	5000000.0
...	...	...	...	...	...	...	...	...	...

452	Trey Lyles	Utah Jazz	41	PF	20	10-Jun	234	Kentucky	2239800.0
453	Shelvin Mack	Utah Jazz	8	PG	26	3-Jun	203	Butler	2433333.0
454	Raul Neto	Utah Jazz	25	PG	24	1-Jun	179	NaN	900000.0
455	Tibor Pleiss	Utah Jazz	21	C	26	3-Jul	256	NaN	2900000.0
456	Jeff Withey	Utah Jazz	24	C	26	Jul-00	231	Kansas	947276.0

457 rows × 9 columns

In [51]:
```
data_nba.shape
```

Out[51]:
```
(457, 9)
```

In [57]:
```
data_nba['Position'].value_counts()
```

Out[57]:
```
SG    102
PF    100
PG     92
SF     85
C      78
Name: Position, dtype: int64
```

In [58]:
```
data_nba
```

Out[58]:

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0	PG	25	2-Jun	180	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99	SF	25	6-Jun	235	Marquette	6796117.0
2	John Holland	Boston Celtics	30	SG	27	5-Jun	205	Boston University	NaN
3	R.J. Hunter	Boston Celtics	28	SG	22	5-Jun	185	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8	PF	29	10-Jun	231	NaN	5000000.0
...	...	...	...	...	...	...	...	...	...
452	Trey Lyles	Utah Jazz	41	PF	20	10-Jun	234	Kentucky	2239800.0
453	Shelvin Mack	Utah Jazz	8	PG	26	3-Jun	203	Butler	2433333.0
454	Raul Neto	Utah Jazz	25	PG	24	1-Jun	179	NaN	900000.0
455	Tibor Pleiss	Utah Jazz	21	C	26	3-Jul	256	NaN	2900000.0
456	Jeff Withey	Utah Jazz	24	C	26	Jul-00	231	Kansas	947276.0

457 rows × 9 columns

In [59]:
```
## Converting to numerics
data_nba['Position'].replace(['SG','PF','PG','SF','C'],[0,1,2,3,4], inplace=True)
```

In [60]:
```
data_nba
```

Out[60]:

	Name	Team	Number	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0	2	25	2-Jun	180	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99	3	25	6-Jun	235	Marquette	6796117.0
2	John Holland	Boston Celtics	30	0	27	5-Jun	205	Boston University	NaN
3	R.J. Hunter	Boston Celtics	28	0	22	5-Jun	185	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8	1	29	10-Jun	231	NaN	5000000.0

...	...	...	...	...	...	...	...	...	...
452	Trey Lyles	Utah Jazz	41	1	20	10-Jun	234	Kentucky	2239800.0
453	Shelvin Mack	Utah Jazz	8	2	26	3-Jun	203	Butler	2433333.0
454	Raul Neto	Utah Jazz	25	2	24	1-Jun	179	NaN	900000.0
455	Tibor Pleiss	Utah Jazz	21	4	26	3-Jul	256	NaN	2900000.0
456	Jeff Withey	Utah Jazz	24	4	26	Jul-00	231	Kansas	947276.0

457 rows × 9 columns

In [83]:
```
data_nba.to_csv('Replace_function_preprocess')
```

Label Encoding

In [61]:
```
df2= pd.read_csv("nba.csv")
```

In [62]:
```
from sklearn import preprocessing
```

In [63]:
```
df2['Position'].unique()
```

Out[63]:
```
array(['PG', 'SF', 'SG', 'PF', 'C'], dtype=object)
```

In [65]:
```
label_encoder = preprocessing.LabelEncoder()
df2['Position'] = label_encoder.fit_transform(df2['Position'])
```

In [67]:
```
df2['Position'].unique()
```

Out[67]:
```
array([2, 3, 4, 1, 0])
```

In [68]:
```
df2.Age.min()
```

Out[68]:
```
19
```

In [69]:
```
df2.Age.max()
```

Out[69]:
```
40
```

Quantitative to Categorical

In [78]:
```
category = pd.cut(df2.Age, bins=[19,25,30,35,40],labels=['A','B','C','D'])
```

In [80]:
```
## insert in df2
df2.insert(3, 'Age_Group', category)
```

In [81]:
```
df2
```

Out[81]:

	Name	Team	Number	Age_Group	Position	Age	Height	Weight	College	Salary
0	Avery Bradley	Boston Celtics	0	A	2	25	2-Jun	180	Texas	7730337.0
1	Jae Crowder	Boston Celtics	99	A	3	25	6-Jun	235	Marquette	6796117.0
2	John Holland	Boston Celtics	30	B	4	27	5-Jun	205	Boston University	NaN

3	R.J. Hunter	Boston Celtics	28	A	4	22	5-Jun	185	Georgia State	1148640.0
4	Jonas Jerebko	Boston Celtics	8	B	1	29	10-Jun	231	NaN	5000000.0
...	...	...	...	...	...	...	...	...	...	...
452	Trey Lyles	Utah Jazz	41	A	1	20	10-Jun	234	Kentucky	2239800.0
453	Shelvin Mack	Utah Jazz	8	B	2	26	3-Jun	203	Butler	2433333.0
454	Raul Neto	Utah Jazz	25	A	2	24	1-Jun	179	NaN	900000.0
455	Tibor Pleiss	Utah Jazz	21	B	0	26	3-Jul	256	NaN	2900000.0
456	Jeff Withey	Utah Jazz	24	B	0	26	Jul-00	231	Kansas	947276.0

457 rows × 10 columns

In [82]:
```
df2.to_csv('Preprocessed_nba_csv')
```

In [ ]:
```
```