```
#mount and import hospital_er csv file
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
import pandas as pd
```

```
df = pd.read_csv('/content/drive/MyDrive/ER data/Copy of Hospital ER_Data.csv')
```

```
df.head()
```

	Patient Id	Patient Admission Date	Patient First Inital	Patient Last Name	Patient Gender	Patient Age	Patient Race	Department Referral	Patient Admission Flag	Patient Satisfaction Score	Patient Waittime	Patients CM
0	145-39-5406	20-03-2024 08:47	H	Glasspool	M	69	White	NaN	False	10.0	39	0
1	316-34-3057	15-06-2024 11:29	X	Methuen	M	4	Native American/Alaska Native	NaN	True	NaN	27	0
2	897-46-3852	20-06-2024 09:13	P	Schubuser	F	56	African American	General Practice	True	9.0	55	0
	358-31-	04-02-					Native	General				

```
#columns with numerical data
df.describe()
```

	Patient Age	Patient Satisfaction Score	Patient Waittime	Patients CM
count	9216.000000	2517.000000	9216.000000	9216.000000
mean	39.855143	4.992054	35.259874	0.052083
std	22.755125	3.138043	14.735323	0.222207
min	1.000000	0.000000	10.000000	0.000000
25%	20.000000	2.000000	23.000000	0.000000
50%	39.000000	5.000000	35.000000	0.000000
75%	60.000000	8.000000	48.000000	0.000000
max	79.000000	10.000000	60.000000	1.000000

```
df['Patients CM'].value_counts()
```

Patients CM	count
0	8736
1	480

dtype: int64

```
# looking for missing values
df.isnull().sum()
```

	0
Patient Id	0
Patient Admission Date	0
Patient First Inital	0
Patient Last Name	0
Patient Gender	0
Patient Age	0
Patient Race	0
Department Referral	5400
Patient Admission Flag	0
Patient Satisfaction Score	6699
Patient Waittime	0
Patients CM	0

dtype: int64

```
#inspecting columns ["Department Referral"] and ["Patient Satisfaction Score"]
df.loc[:,["Department Referral","Patient Satisfaction Score"]]
# ["Patient Satisfaction Score"] 0 for score and representative of missing data, use for service improvement
# ["Department Referral"] not 0 but representative of no referral needed
```

	Department Referral	Patient Satisfaction Score
0	NaN	10.0
1	NaN	NaN
2	General Practice	9.0
3	General Practice	8.0
4	Orthopedics	NaN
...	...	...
9211	General Practice	NaN
9212	NaN	NaN
9213	NaN	NaN
9214	General Practice	1.0
9215	General Practice	NaN

9216 rows × 2 columns

```
df.head()
```

	Patient Id	Patient Admission Date	Patient First Inital	Patient Last Name	Patient Gender	Patient Age	Patient Race	Department Referral	Patient Admission Flag	Patient Satisfaction Score	Patient Waittime	Patients CM
0	145-39-5406	20-03-2024 08:47	H	Glasspool	M	69	White	NaN	False	10.0	39	0
1	316-34-3057	15-06-2024 11:29	X	Methuen	M	4	Native American/Alaska Native	NaN	True	NaN	27	0
2	897-46-3852	20-06-2024 09:13	P	Schubuser	F	56	African American	General Practice	True	9.0	55	0
3	358-31-	04-02-2024	U	Titcombe	F	24	Native American/Alaska	General	True	8.0	31	0

```
# concatenation Patient Full Name
df["Patient Full Name"] = df["Patient First Inital"] + " " + df["Patient Last Name"]
df.head()
```

	Patient Id	Patient Admission Date	Patient First Inital	Patient Last Name	Patient Gender	Patient Age	Patient Race	Department Referral	Patient Admission Flag	Patient Satisfaction Score	Patient Waittime	Patients CM
0	145-39-5406	20-03-2024 08:47	H	Glasspool	M	69	White	NaN	False	10.0	39	0
1	316-34-3057	15-06-2024 11:29	X	Methuen	M	4	Native American/Alaska Native	NaN	True	NaN	27	0
2	897-46-3852	20-06-2024 09:13	P	Schubuser	F	56	African American	General Practice	True	9.0	55	0
	358-31-	04-02-					Native	General				

```
df["Patient Gender"].value_counts()
# rename columns for clarity
```

Patient Gender	count
M	4705
F	4487
NC	24

dtype: int64

```
df["Patient Gender"] = df["Patient Gender"].replace("M","Male")
df["Patient Gender"] = df["Patient Gender"].replace("F","Female")
df["Patient Gender"] = df["Patient Gender"].replace("NC","Not Conformed")
df["Patient Gender"].value_counts()
```

Patient Gender	count
Male	4705
Female	4487
Not Conformed	24

dtype: int64

```
df.to_csv("cleaned_hospital_er.csv", index=False)
```