In [1]:

```
import pandas as pd
```

In [2]:

```
dcp = pd.read_csv("DCP CompuMed spreadsheet.csv")
```

In [3]:

```
dcp.head(5)
```

Out[3]:

	Exam ID	Patient ID	Patient Last, First	Age	DOB	Received Time	Study Time	Completed Time	Ov
0	389081	DCP001	EV, EV	38	10/11/1980	09/07/18 09:31	08/18/18 02:19	09/07/18 16:27	Norr rhythm\r\nWithi
1	389107	DCP002	RF, RF	52	3/24/1966	09/07/18 09:31	08/18/18 15:43	09/07/18 17:24	Norr rhythm\r\nSlov V1-3\
2	389106	DCP002	RF, RF	52	3/24/1966	09/07/18 09:31	08/18/18 10:42	09/07/18 17:00	Norr rhythm\r\nSlov progı
3	389088	DCP003	JE, JE	46	1/5/1972	09/07/18 09:31	08/18/18 10:46	09/07/18 16:49	Norr rhythm\r\nBorde v
4	389109	DCP004	RK, RK	36	12/2/1981	09/07/18 09:31	08/18/18 11:01	09/07/18 16:29	Norr rhythm\r\nIntrave

In [4]:

```
dcp.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 29 entries, 0 to 28
Data columns (total 13 columns):
Exam ID                      29 non-null int64
Patient ID                   29 non-null object
Patient Last, First          29 non-null object
Age                          29 non-null int64
DOB                          29 non-null object
Received Time                29 non-null object
Study Time                   29 non-null object
Completed Time               29 non-null object
Over Read                    29 non-null object
Physician<br />Last Name     29 non-null object
Physician<br />First Name    29 non-null object
Gender                       29 non-null object
Notes                        0 non-null float64
dtypes: float64(1), int64(2), object(10)
memory usage: 3.0+ KB
```

Need Modify Issues

1. Cloumn Names need to modify
    - Replace spaces with underscores
    - Remove special characters
    - Make all labels to lowercase
    - Shorten any long column names
2. Convert columns to numeric if need
3. Change data to DateTime if data is date

Continue Adding

1. Column Names need to modify

In [5]:

```
dcp.columns
```

Out[5]:

```
Index(['Exam ID', 'Patient ID', 'Patient Last, First', 'Age', 'DOB',
       'Received Time', 'Study Time', 'Completed Time', 'Over Read',
       'Physician<br />Last Name', 'Physician<br />First Name', 'Gender',
       'Notes'],
      dtype='object')
```

In [6]:

```
def clean_col(col):
    col = col.strip() # remove whitespaces
    col = col.replace(" ", "_") # replace spaces to _
    col = col.replace("<", "")
    col = col.replace("br","")
    col = col.replace("/","")
    col = col.replace(">","")
    col = col.replace(",","")
    col = col.lower() # lowercase all names
    return col

new_column = []
for x in dcp.columns:
    clean = clean_col(x)
    new_column.append(clean)

dcp.columns = new_column
dcp.columns
```

Out[6]:

```
Index(['exam_id', 'patient_id', 'patient_last_first', 'age', 'dob',
       'received_time', 'study_time', 'completed_time', 'over_read',
       'physician_last_name', 'physician_first_name', 'gender', 'notes'],
      dtype='object')
```

In [7]:

```
dcp["patient_id"].unique()
```

Out[7]:

```
array(['DCP001', 'DCP002', 'DCP003', 'DCP004', 'DCP005', 'DCP006',
       'DCP007', 'DCP008', 'DCP010', 'DCP011', 'DCP012', 'DCP013',
       'DCP014', 'DCP015', 'DCP016', 'DCP017', 'DCP018', 'DCP019',
       'DCP020', 'DCP021', 'DCP022', 'DCP023', 'DCP024', 'DCP025',
       'DCP026', 'DCP027', 'DCP029', 'DCP030'], dtype=object)
```

In [8]:

```
dcp["patient_id"] = dcp["patient_id"].str.replace("DCP", "")
dcp["patient_id"].unique()
```

Out[8]:

```
array(['001', '002', '003', '004', '005', '006', '007', '008', '010',
       '011', '012', '013', '014', '015', '016', '017', '018', '019',
       '020', '021', '022', '023', '024', '025', '026', '027', '029',
       '030'], dtype=object)
```

2. Convert Column to numeric

In [9]:

```
dcp["patient_id"] = dcp["patient_id"].astype(int)
dcp.dtypes
```

Out[9]:

```
exam_id                 int64
patient_id              int32
patient_last_first     object
age                     int64
dob                    object
received_time          object
study_time             object
completed_time         object
over_read              object
physician_last_name    object
physician_first_name   object
gender                 object
notes                 float64
dtype: object
```

Shorten long column name and data

In [10]:

```
dcp["patient_last_first"].unique()
```

Out[10]:

```
array(['EV, EV', 'RF, RF', 'JE, JE', 'RK, RK', 'JT, JT', 'JA, JA',
       'BG, BG', 'AN, AN', 'JB, JB', 'DR, DR', 'TL, TL', 'NL, NL',
       'JV, JV', 'VM, VM', 'PZ, PZ', 'HB, HB', 'MB, MB', 'BY, BY',
       'VP, VP', 'CP, CP', 'CA, CA', 'JEH, JEH', 'SS, SS', 'LV, LV',
       'AM, AM', 'DF, DF', 'NC, NC'], dtype=object)
```

In [11]:

```
dcp["patient_last_first"] = dcp["patient_last_first"].str.split().str[0]
dcp["patient_last_first"].unique()
```

Out[11]:

```
array(['EV,', 'RF,', 'JE,', 'RK,', 'JT,', 'JA,', 'BG,', 'AN,', 'JB,',
       'DR,', 'TL,', 'NL,', 'JV,', 'VM,', 'PZ,', 'HB,', 'MB,', 'BY,',
       'VP,', 'CP,', 'CA,', 'JEH,', 'SS,', 'LV,', 'AM,', 'DF,', 'NC,'],
      dtype=object)
```

In [12]:

```
dcp["patient_last_first"] = dcp["patient_last_first"].str.replace(",", "")
dcp["patient_last_first"].unique()
```

Out[12]:

```
array(['EV', 'RF', 'JE', 'RK', 'JT', 'JA', 'BG', 'AN', 'JB', 'DR', 'TL',
       'NL', 'JV', 'VM', 'PZ', 'HB', 'MB', 'BY', 'VP', 'CP', 'CA', 'JEH',
       'SS', 'LV', 'AM', 'DF', 'NC'], dtype=object)
```

In [13]:

```
dcp.rename({"patient_last_first": "patient_name"}, axis =1,
          inplace = True)
dcp.dtypes
```

Out[13]:

```
exam_id                 int64
patient_id              int32
patient_name           object
age                     int64
dob                    object
received_time          object
study_time             object
completed_time         object
over_read              object
physician_last_name    object
physician_first_name   object
gender                 object
notes                 float64
dtype: object
```

In [14]:

```
dcp["dob"] = pd.to_datetime(dcp["dob"])
dcp.dtypes
```

Out[14]:

```
exam_id                          int64
patient_id                       int32
patient_name                    object
age                              int64
dob                     datetime64[ns]
received_time                   object
study_time                      object
completed_time                  object
over_read                       object
physician_last_name             object
physician_first_name            object
gender                          object
notes                          float64
dtype: object
```

In [15]:

```
dcp.head()
```

Out[15]:

	exam_id	patient_id	patient_name	age	dob	received_time	study_time	completed_time
0	389081	1	EV	38	1980-10-11	09/07/18 09:31	08/18/18 02:19	09/07/18 16:27
1	389107	2	RF	52	1966-03-24	09/07/18 09:31	08/18/18 15:43	09/07/18 17:24
2	389106	2	RF	52	1966-03-24	09/07/18 09:31	08/18/18 10:42	09/07/18 17:00
3	389088	3	JE	46	1972-01-05	09/07/18 09:31	08/18/18 10:46	09/07/18 16:49
4	389109	4	RK	36	1981-12-02	09/07/18 09:31	08/18/18 11:01	09/07/18 16:29

In [16]:

```
dcp["physician_name"] = dcp["physician_last_name"].values + dcp["physician_first_name"]
.values
dcp.dtypes
```

Out[16]:

```
exam_id                          int64
patient_id                       int32
patient_name                    object
age                              int64
dob                     datetime64[ns]
received_time                   object
study_time                      object
completed_time                  object
over_read                       object
physician_last_name             object
physician_first_name            object
gender                          object
notes                          float64
physician_name                  object
dtype: object
```

In [17]:

```
dcp["physician_name"].values
```

Out[17]:

```
array([' ShiroffRobert', ' ShiroffRobert', ' ShiroffRobert',
       ' ShiroffRobert', ' ShiroffRobert', ' ShiroffRobert',
       ' ShiroffRobert', ' ShiroffRobert', ' ShiroffRobert',
       ' ShiroffRobert', ' ShiroffRobert', ' ShiroffRobert',
       ' ShiroffRobert', ' ShiroffRobert', ' ShiroffRobert',
       ' ShiroffRobert', ' ShiroffRobert', ' ShiroffRobert',
       ' ShiroffRobert', ' ShiroffRobert', ' ShiroffRobert',
       ' ShiroffRobert', ' ShiroffRobert'], dtype=object)
```

In [18]:

```
dcp["physician_name"] = dcp["physician_name"].str.replace("ShiroffRobert", "shiroff_rob
ert").str.strip()
dcp["physician_name"].values
```

Out[18]:

```
array(['shiroff_robert', 'shiroff_robert', 'shiroff_robert',
       'shiroff_robert', 'shiroff_robert', 'shiroff_robert',
       'shiroff_robert', 'shiroff_robert', 'shiroff_robert',
       'shiroff_robert', 'shiroff_robert', 'shiroff_robert',
       'shiroff_robert', 'shiroff_robert', 'shiroff_robert',
       'shiroff_robert', 'shiroff_robert', 'shiroff_robert',
       'shiroff_robert', 'shiroff_robert', 'shiroff_robert',
       'shiroff_robert', 'shiroff_robert'], dtype=object)
```

In [19]:

```
dcp["completed_time"] = pd.to_datetime(dcp["completed_time"])
dcp.dtypes
```

Out[19]:

```
exam_id                          int64
patient_id                       int32
patient_name                    object
age                              int64
dob                     datetime64[ns]
received_time                   object
study_time                      object
completed_time          datetime64[ns]
over_read                       object
physician_last_name             object
physician_first_name            object
gender                          object
notes                          float64
physician_name                  object
dtype: object
```

In [20]:

```
%matplotlib inline
import matplotlib.pyplot as plt
```

In [21]:

```
y_values = dcp["age"]
x_values = dcp["gender"]
plt.scatter(x_values, y_values)
plt.show()
```

In [22]:

```
dcp.head()
```

Out[22]:

	exam_id	patient_id	patient_name	age	dob	received_time	study_time	completed_time
0	389081	1	EV	38	1980-10-11	09/07/18 09:31	08/18/18 02:19	2018-09-07 16:27:00
1	389107	2	RF	52	1966-03-24	09/07/18 09:31	08/18/18 15:43	2018-09-07 17:24:00
2	389106	2	RF	52	1966-03-24	09/07/18 09:31	08/18/18 10:42	2018-09-07 17:00:00
3	389088	3	JE	46	1972-01-05	09/07/18 09:31	08/18/18 10:46	2018-09-07 16:49:00
4	389109	4	RK	36	1981-12-02	09/07/18 09:31	08/18/18 11:01	2018-09-07 16:29:00

In [23]:

```
dcp.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 29 entries, 0 to 28
Data columns (total 14 columns):
exam_id                 29 non-null int64
patient_id              29 non-null int32
patient_name            29 non-null object
age                     29 non-null int64
dob                     29 non-null datetime64[ns]
received_time           29 non-null object
study_time              29 non-null object
completed_time          29 non-null datetime64[ns]
over_read               29 non-null object
physician_last_name     29 non-null object
physician_first_name    29 non-null object
gender                  29 non-null object
notes                   0 non-null float64
physician_name          29 non-null object
dtypes: datetime64[ns](2), float64(1), int32(1), int64(2), object(8)
memory usage: 3.1+ KB
```

In [24]:

```
dcp.to_csv("dcp_v1.csv")
```

In [ ]:

```

```