

In [17]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [18]:

```
#reading dataset
dataset = pd.read_csv('encoders (1).csv')
```

In [19]:

```
dataset.head()
```

Out[19]:

	id	date	scanner	min	max	err	pixels	minf	maxf	errf	created_at	update_at
0	12	2017-12-17	K219	35435	35933	1.40	6	35681	35688	0.02	NaN	2017-12-17 16:26:16.4
1	30	2017-12-18	H161	35155	36382	3.43	14	35731	35761	0.08	NaN	2017-12-18 16:26:16.4
2	47	2017-12-18	K211	35305	36042	2.07	43	35692	35739	0.13	NaN	2017-12-18 18:10:57.4
3	48	2017-12-18	K212	35216	36225	2.82	61	35686	35726	0.11	NaN	2017-12-18 18:10:57.4
4	49	2017-12-18	K220	35196	36259	2.98	11	35709	35724	0.04	NaN	2017-12-18 18:10:57.4

In [20]:

```
dataset.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2752 entries, 0 to 2751
Data columns (total 12 columns):
#   Column      Non-Null Count  Dtype
---  -
0   id           2752 non-null   int64
1   date         2752 non-null   object
2   scanner      2752 non-null   object
3   min          2752 non-null   int64
4   max          2752 non-null   int64
5   err          2752 non-null   float64
6   pixels       2752 non-null   int64
7   minf         2752 non-null   int64
8   maxf         2752 non-null   int64
9   errf         2752 non-null   float64
10  created_at   0 non-null      float64
11  updated_at   1921 non-null   object
dtypes: float64(3), int64(6), object(3)
memory usage: 258.1+ KB
```

In [21]:

```
# make string version of original column, call it 'col'
dataset['updated_at1'] = dataset['updated_at'].astype(str)
```

In [22]:

```
# make the new columns using string indexing
dataset['created_at1'] = dataset['updated_at1'].str[0:10]
dataset['updated_at2'] = dataset['updated_at1'].str[10:19]

# get rid of the extra variable (if you want)
dataset.drop('updated_at1', axis=1, inplace=True)
```

In [23]:

```
dataset.head()
```

Out[23]:

	id	date	scanner	min	max	err	pixels	minf	maxf	errf	created_at	update
0	12	2017-12-17	K219	35435	35933	1.40	6	35681	35688	0.02	NaN	2017-12-17 16:26:16.4
1	30	2017-12-18	H161	35155	36382	3.43	14	35731	35761	0.08	NaN	2017-12-18 16:26:16.1
2	47	2017-12-18	K211	35305	36042	2.07	43	35692	35739	0.13	NaN	2017-12-18 18:10:57.4
3	48	2017-12-18	K212	35216	36225	2.82	61	35686	35726	0.11	NaN	2017-12-18 18:10:57.4
4	49	2017-12-18	K220	35196	36259	2.98	11	35709	35724	0.04	NaN	2017-12-18 18:10:57.4

In [24]:

```
dataset.columns
```

Out[24]:

```
Index(['id', 'date', 'scanner', 'min', 'max', 'err', 'pixels', 'minf', 'maxf', 'errf', 'created_at', 'updated_at', 'created_at1', 'updated_at2'],
      dtype='object')
```

In [25]:

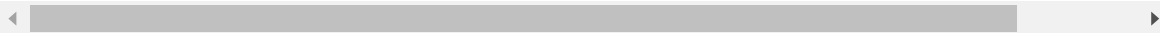
```
# dropping passed columns
dataset.drop(["created_at", "updated_at"], axis = 1, inplace = True)

# display
dataset
```

Out[25]:

	id	date	scanner	min	max	err	pixels	minf	maxf	errf	created_at1	u
0	12	2017-12-17	K219	35435	35933	1.40	6	35681	35688	0.02	2017-12-29	
1	30	2017-12-18	H161	35155	36382	3.43	14	35731	35761	0.08	2017-12-29	
2	47	2017-12-18	K211	35305	36042	2.07	43	35692	35739	0.13	2017-12-19	
3	48	2017-12-18	K212	35216	36225	2.82	61	35686	35726	0.11	2017-12-19	
4	49	2017-12-18	K220	35196	36259	2.98	11	35709	35724	0.04	2017-12-19	
...
2747	2748	2018-03-07	K171	35109	35912	2.26	50	35504	35531	0.08	2018-03-08	
2748	2749	2017-12-07	K211	-1	-1	-0.00	-1	-1	-1	-0.00	nan	
2749	2750	2017-12-07	K212	-1	-1	-0.00	-1	-1	-1	-0.00	nan	
2750	2751	2018-03-07	K220	35181	36229	2.94	9	35702	35715	0.04	2018-03-08	
2751	2752	2018-03-07	K221	35288	36266	2.73	3	35734	35737	0.01	2018-03-08	

2752 rows × 12 columns



updated_at2

- not a categorical data
- its dtype is string
- if empty space found, replace with nan values and delete rows

In [26]:

```
dataset['updated_at2'].replace('', np.nan, inplace=True)  
print(dataset['updated_at2'])
```

```
0      16:26:16  
1      16:26:16  
2      18:10:57  
3      18:10:57  
4      18:10:57  
...  
2747    12:24:51  
2748         NaN  
2749         NaN  
2750    14:42:19  
2751    14:42:19  
Name: updated_at2, Length: 2752, dtype: object
```

In [27]:

```
dataset.dropna(subset=['updated_at2'], inplace=True)  
print(dataset['updated_at2'])
```

```
0      16:26:16  
1      16:26:16  
2      18:10:57  
3      18:10:57  
4      18:10:57  
...  
2745    12:24:51  
2746    12:24:51  
2747    12:24:51  
2750    14:42:19  
2751    14:42:19  
Name: updated_at2, Length: 1921, dtype: object
```

In [28]:

dataset

Out[28]:

	id	date	scanner	min	max	err	pixels	minf	maxf	errf	created_at1	up
0	12	2017-12-17	K219	35435	35933	1.40	6	35681	35688	0.02	2017-12-29	
1	30	2017-12-18	H161	35155	36382	3.43	14	35731	35761	0.08	2017-12-29	
2	47	2017-12-18	K211	35305	36042	2.07	43	35692	35739	0.13	2017-12-19	
3	48	2017-12-18	K212	35216	36225	2.82	61	35686	35726	0.11	2017-12-19	
4	49	2017-12-18	K220	35196	36259	2.98	11	35709	35724	0.04	2017-12-19	
...
2745	2745	2018-03-07	K168	35185	35958	2.17	48	35541	35565	0.07	2018-03-08	
2746	2747	2018-03-07	K114	35220	36071	2.39	17	35715	35735	0.06	2018-03-08	
2747	2748	2018-03-07	K171	35109	35912	2.26	50	35504	35531	0.08	2018-03-08	
2750	2751	2018-03-07	K220	35181	36229	2.94	9	35702	35715	0.04	2018-03-08	
2751	2752	2018-03-07	K221	35288	36266	2.73	3	35734	35737	0.01	2018-03-08	

1921 rows × 12 columns

In [29]:

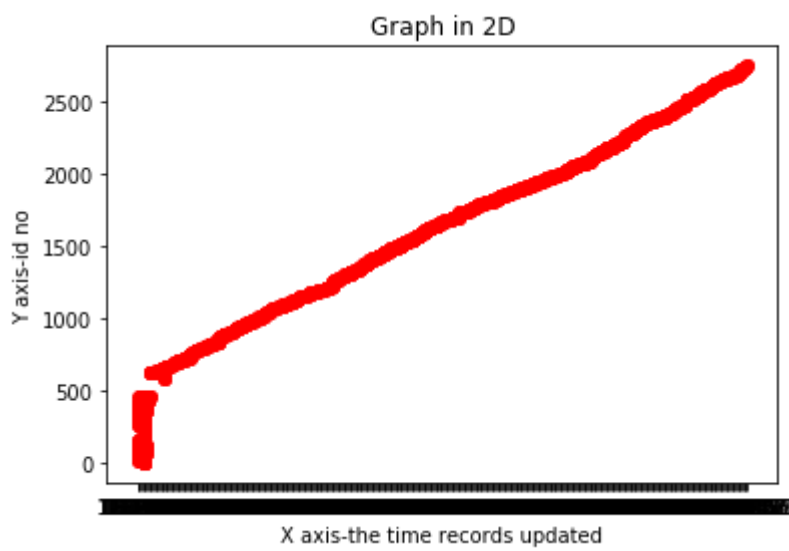
```
x = dataset['updated_at2']
y = dataset['id']
```

In [30]:

```
##plotting using matplotlib  
  
##plt scatter  
  
plt.scatter(x,y,c='r')  
plt.xlabel('X axis-the time records updated')  
plt.ylabel('Y axis-id no')  
plt.title('Graph in 2D')
```

Out[30]:

Text(0.5, 1.0, 'Graph in 2D')



In [31]:

```
dataset.isnull()
```

Out[31]:

	id	date	scanner	min	max	err	pixels	minf	maxf	errf	created_at1	up
0	False	False	False	False	False	False	False	False	False	False	False	False
1	False	False	False	False	False	False	False	False	False	False	False	False
2	False	False	False	False	False	False	False	False	False	False	False	False
3	False	False	False	False	False	False	False	False	False	False	False	False
4	False	False	False	False	False	False	False	False	False	False	False	False
...
2745	False	False	False	False	False	False	False	False	False	False	False	False
2746	False	False	False	False	False	False	False	False	False	False	False	False
2747	False	False	False	False	False	False	False	False	False	False	False	False
2750	False	False	False	False	False	False	False	False	False	False	False	False
2751	False	False	False	False	False	False	False	False	False	False	False	False

1921 rows × 12 columns



In [32]:

```
dataset.to_csv('encoders (123).csv', index=False)
```

In []: