

Lab 6 - Heart rate sensor data

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

```
In [3]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
hr02 = pd.read_csv("results.csv")
```

hr02

```
Out[3]:
```

	HRvalue	ts	HRvalid	SpO2vaild	SpO2value
0	115	1712373557000	True	True	69.81524999999999
1	75	1712372746445	True	True	86.398056
2	65	1712373346868	True	False	'-999
3	65	1712373617034	True	True	99.519096
4	166	1712373376885	True	False	'-999
5	55	1712373286829	True	True	99.462504
6	150	1712373466958	True	False	'-999
7	93	1712372956571	True	True	47.63735399999999
8	136	1712373316844	True	False	'-999
9	71	1712372566340	True	False	'-999
10	71	1712372596358	True	True	2.9477999999999724
11	71	1712372656388	True	True	99.016626
12	71	1712372686416	True	True	99.0534
13	71	1712372716425	True	True	62.00625
14	71	1712372776462	True	True	77.36426399999999
15	71	1712372806488	True	True	66.02340000000001
16	71	1712372836498	True	True	90.46571399999999
17	71	1712373226788	True	True	97.74405
18	71	1712373526979	True	True	73.3818
19	214	1712373076651	True	False	'-999
20	88	1712373046631	True	False	'-999
21	88	1712373406907	True	True	62.827704
22	78	1712372926549	True	True	92.771946
23	78	1712373106671	True	True	52.37546399999999
24	78	1712373166725	True	False	'-999
25	107	1712373256806	True	True	99.275976
26	68	1712372626373	True	False	'-999
27	68	1712372986592	True	True	96.6558
28	68	1712373016609	True	True	74.74533599999998
29	68	1712373496963	True	True	88.747986
30	83	1712372866536	True	True	72.68651399999999
31	83	1712372896533	True	True	75.413586
32	83	1712373196727	True	False	'-999
33	83	1712373436934	True	True	68.32554599999999
34	83	1712373587014	True	True	99.43662599999999
35	60	1712373136689	True	True	98.169384

```
In [4]: hr02.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 36 entries, 0 to 35
Data columns (total 5 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   HRvalue     36 non-null    int64
1   ts          36 non-null    int64
2   HRvalid     36 non-null    bool
3   SpO2vaild   36 non-null    bool
4   SpO2value   36 non-null    object
dtypes: bool(2), int64(2), object(1)
memory usage: 1.0+ KB
```

```
In [5]: # Converting the ts value to date-time format
hr02.ts=pd.to_datetime(hr02.ts, unit='ms')
```

hr02

Out[5]:

	HRvalue	ts	HRvalid	SpO2vaild	SpO2value
0	115	2024-04-06 03:19:17.000	True	True	69.81524999999999
1	75	2024-04-06 03:05:46.445	True	True	86.398056
2	65	2024-04-06 03:15:46.868	True	False	'-999
3	65	2024-04-06 03:20:17.034	True	True	99.519096
4	166	2024-04-06 03:16:16.885	True	False	'-999
5	55	2024-04-06 03:14:46.829	True	True	99.462504
6	150	2024-04-06 03:17:46.958	True	False	'-999
7	93	2024-04-06 03:09:16.571	True	True	47.63735399999999
8	136	2024-04-06 03:15:16.844	True	False	'-999
9	71	2024-04-06 03:02:46.340	True	False	'-999
10	71	2024-04-06 03:03:16.358	True	True	2.9477999999999724
11	71	2024-04-06 03:04:16.388	True	True	99.016626
12	71	2024-04-06 03:04:46.416	True	True	99.0534
13	71	2024-04-06 03:05:16.425	True	True	62.00625
14	71	2024-04-06 03:06:16.462	True	True	77.36426399999999
15	71	2024-04-06 03:06:46.488	True	True	66.02340000000001
16	71	2024-04-06 03:07:16.498	True	True	90.46571399999999
17	71	2024-04-06 03:13:46.788	True	True	97.74405
18	71	2024-04-06 03:18:46.979	True	True	73.3818
19	214	2024-04-06 03:11:16.651	True	False	'-999
20	88	2024-04-06 03:10:46.631	True	False	'-999
21	88	2024-04-06 03:16:46.907	True	True	62.827704
22	78	2024-04-06 03:08:46.549	True	True	92.771946
23	78	2024-04-06 03:11:46.671	True	True	52.37546399999999
24	78	2024-04-06 03:12:46.725	True	False	'-999
25	107	2024-04-06 03:14:16.806	True	True	99.275976
26	68	2024-04-06 03:03:46.373	True	False	'-999
27	68	2024-04-06 03:09:46.592	True	True	96.6558
28	68	2024-04-06 03:10:16.609	True	True	74.74533599999998
29	68	2024-04-06 03:18:16.963	True	True	88.747986
30	83	2024-04-06 03:07:46.536	True	True	72.68651399999999
31	83	2024-04-06 03:08:16.533	True	True	75.413586
32	83	2024-04-06 03:13:16.727	True	False	'-999
33	83	2024-04-06 03:17:16.934	True	True	68.32554599999999
34	83	2024-04-06 03:19:47.014	True	True	99.43662599999999
35	60	2024-04-06 03:12:16.689	True	True	98.169384

In [6]:

```
#Make the date time sequential  
hr02.sort_values('ts', inplace=True)
```

In [7]:

hr02

Out[7]:

	HRvalue	ts	HRvalid	SpO2vaild	SpO2value
9	71	2024-04-06 03:02:46.340	True	False	'-999
10	71	2024-04-06 03:03:16.358	True	True	2.9477999999999724
26	68	2024-04-06 03:03:46.373	True	False	'-999
11	71	2024-04-06 03:04:16.388	True	True	99.016626
12	71	2024-04-06 03:04:46.416	True	True	99.0534
13	71	2024-04-06 03:05:16.425	True	True	62.00625
1	75	2024-04-06 03:05:46.445	True	True	86.398056
14	71	2024-04-06 03:06:16.462	True	True	77.36426399999999
15	71	2024-04-06 03:06:46.488	True	True	66.02340000000001
16	71	2024-04-06 03:07:16.498	True	True	90.46571399999999
30	83	2024-04-06 03:07:46.536	True	True	72.68651399999999
31	83	2024-04-06 03:08:16.533	True	True	75.413586
22	78	2024-04-06 03:08:46.549	True	True	92.771946
7	93	2024-04-06 03:09:16.571	True	True	47.63735399999999
27	68	2024-04-06 03:09:46.592	True	True	96.6558
28	68	2024-04-06 03:10:16.609	True	True	74.74533599999998
20	88	2024-04-06 03:10:46.631	True	False	'-999
19	214	2024-04-06 03:11:16.651	True	False	'-999
23	78	2024-04-06 03:11:46.671	True	True	52.37546399999999
35	60	2024-04-06 03:12:16.689	True	True	98.169384
24	78	2024-04-06 03:12:46.725	True	False	'-999
32	83	2024-04-06 03:13:16.727	True	False	'-999
17	71	2024-04-06 03:13:46.788	True	True	97.74405
25	107	2024-04-06 03:14:16.806	True	True	99.275976
5	55	2024-04-06 03:14:46.829	True	True	99.462504
8	136	2024-04-06 03:15:16.844	True	False	'-999
2	65	2024-04-06 03:15:46.868	True	False	'-999
4	166	2024-04-06 03:16:16.885	True	False	'-999
21	88	2024-04-06 03:16:46.907	True	True	62.827704
33	83	2024-04-06 03:17:16.934	True	True	68.32554599999999
6	150	2024-04-06 03:17:46.958	True	False	'-999
29	68	2024-04-06 03:18:16.963	True	True	88.747986
18	71	2024-04-06 03:18:46.979	True	True	73.3818
0	115	2024-04-06 03:19:17.000	True	True	69.81524999999999
34	83	2024-04-06 03:19:47.014	True	True	99.43662599999999
3	65	2024-04-06 03:20:17.034	True	True	99.519096

```
In [8]: ### Use the pd.to_numeric( ) method with an option errors='coerce' to cast the column to float64 and set the non-numerical values to NaN  
hr_1 = pd.to_numeric(hr02['HRvalue'], errors='coerce')  
hr_1
```

```

Out[8]:
 9    71
10    71
26    68
11    71
12    71
13    71
 1    75
14    71
15    71
16    71
30    83
31    83
22    78
 7    93
27    68
28    68
20    88
19   214
23    78
35    60
24    78
32    83
17    71
25   107
 5    55
 8   136
 2    65
 4   166
21    88
33    83
 6   150
29    68
18    71
 0   115
34    83
 3    65
Name: HRvalue, dtype: int64

```

```

In [9]: ### The row index with NaN is removed
hr_1.dropna(inplace=True) # inplace=True will change hr_1
hr_1

```

```

Out[9]:
 9    71
10    71
26    68
11    71
12    71
13    71
 1    75
14    71
15    71
16    71
30    83
31    83
22    78
 7    93
27    68
28    68
20    88
19   214
23    78
35    60
24    78
32    83
17    71
25   107
 5    55
 8   136
 2    65
 4   166
21    88
33    83
 6   150
29    68
18    71
 0   115
34    83
 3    65
Name: HRvalue, dtype: int64

```

```

In [10]: hr_1.index

```

```

Out[10]: Index([ 9, 10, 26, 11, 12, 13,  1, 14, 15, 16, 30, 31, 22,  7, 27, 28, 20, 19,
                23, 35, 24, 32, 17, 25,  5,  8,  2,  4, 21, 33,  6, 29, 18,  0, 34,  3],
                dtype='int64')

```

To plot the HR chart, we need to reset the index of hr_1

```

In [11]: hr_1.reset_index(drop=True, inplace=True) # drop=True will remove the old index
hr_1

```

```

Out[11]: 0    71
         1    71
         2    68
         3    71
         4    71
         5    71
         6    75
         7    71
         8    71
         9    71
        10    83
        11    83
        12    78
        13    93
        14    68
        15    68
        16    88
        17   214
        18    78
        19    60
        20    78
        21    83
        22    71
        23   107
        24    55
        25   136
        26    65
        27   166
        28    88
        29    83
        30   150
        31    68
        32    71
        33   115
        34    83
        35    65
Name: HRvalue, dtype: int64

```

```

In [12]: #Plot the Line chart for Heart Rate
plt.figure(figsize=(10,6))

plt.plot(hr_1.index, hr_1, color='blue', marker='o')

plt.ylabel("Heart Rate")
plt.xlabel("Time")
plt.title("Heart Rate Over Time")
plt.grid(True)
plt.show()

```



```

In [13]: #Clean up the SpO2 values
O2_1 = pd.to_numeric(hr02['SpO2value'], errors='coerce')
O2_1

```

```

Out[13]:
9      NaN
10    2.947800
26      NaN
11    99.016626
12    99.053400
13    62.006250
1     86.398056
14    77.364264
15    66.023400
16    90.465714
30    72.686514
31    75.413586
22    92.771946
7     47.637354
27    96.655800
28    74.745336
20      NaN
19      NaN
23    52.375464
35    98.169384
24      NaN
32      NaN
17    97.744050
25    99.275976
5     99.462504
8      NaN
2      NaN
4      NaN
21    62.827704
33    68.325546
6      NaN
29    88.747986
18    73.381800
0     69.815250
34    99.436626
3     99.519096
Name: SpO2value, dtype: float64

```

```

In [14]: # Remove the NaN and reset the index

O2_1.dropna(inplace=True)
O2_1.reset_index(drop=True, inplace=True)
O2_1

```

```

Out[14]:
0     2.947800
1    99.016626
2    99.053400
3    62.006250
4    86.398056
5    77.364264
6    66.023400
7    90.465714
8    72.686514
9    75.413586
10   92.771946
11   47.637354
12   96.655800
13   74.745336
14   52.375464
15   98.169384
16   97.744050
17   99.275976
18   99.462504
19   62.827704
20   68.325546
21   88.747986
22   73.381800
23   69.815250
24   99.436626
25   99.519096
Name: SpO2value, dtype: float64

```

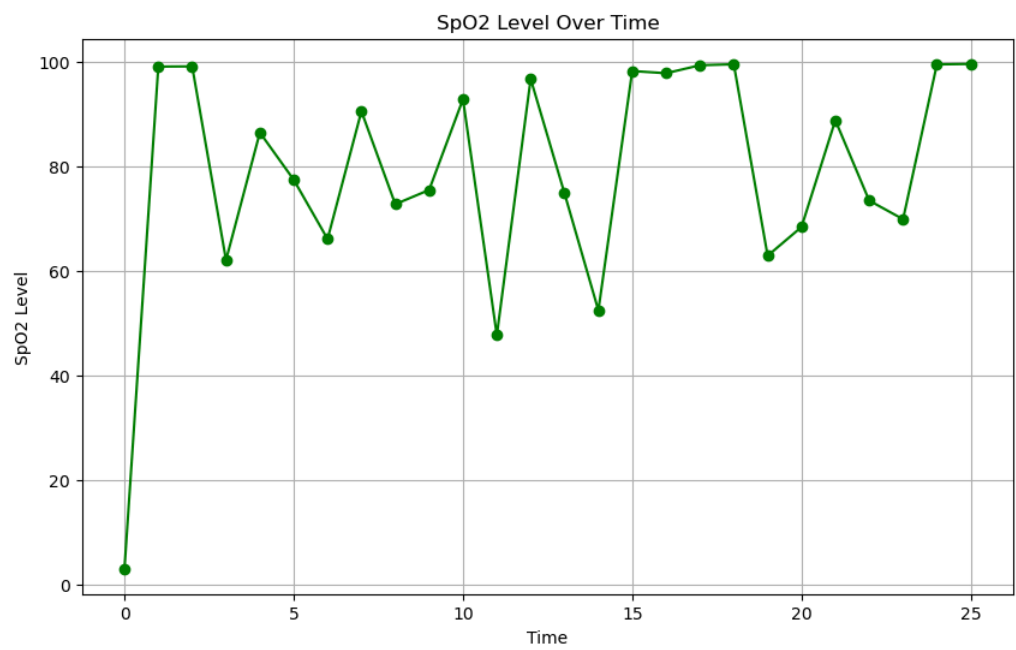
```

In [15]: #Plot the Line chart for the SpO2 values
plt.figure(figsize=(10,6))

plt.plot(O2_1.index, O2_1, color='green', marker='o')

plt.ylabel("SpO2 Level")
plt.xlabel("Time")
plt.title("SpO2 Level Over Time")
plt.grid(True)
plt.show()

```



In []: