1) Introduction

1.1) Project Statement

QR Space would like to gain insight on lower ground and third floor desk occupation of workshop 17 office.

Desk occupancy is monitered by the IoT occupancy sensor,the Infinity PIR1.The sensors measure movement over and output a signal based on whether or not motion was detected over a period.

The sensors output is as follows - 0 = no motion 1 = motion detected 3 = hearbeat signal (a signal that is generated at regular intervals to indicate that sensor is working correctly and should be ignored)

1.2) Objectives

- To create three insightful graphs from the data that could help a decision maker of the property.
- Build a dasboard for data visualisation.

2) Loading Libraries

In [2]:
```
import numpy as np
import pandas as pd
```

3) Loading CSV Datasets

In [3]:
```
root_path = ''
LG_floor_df= pd.read_csv(root_path + 'September 2021 - Lower Ground Floor.csv')
Third_floor_df= pd.read_csv(root_path + 'September 2021 - Third Floor.csv')
```

4) Explore Data Analyses

4.1) Dataframe Overview

In [4]:
```
LG_floor_df.head()
```

Out[4]:

	DateTime	Data	Reading
0	2021/09/01 8:58:30 PM	0	No Event
1	2021/09/01 8:45:54 PM	1	NaN
2	2021/09/01 8:12:23 PM	0	No Event
3	2021/09/01 7:59:30 PM	1	NaN
4	2021/09/01 5:18:07 PM	0	No Event

In [5]:
```
Third_floor_df.head()
```

Out[5]:

	DateTime	Data	Reading
0	2021/09/01 5:40:08 PM	0	No Event
1	2021/09/01 5:28:22 PM	3	NaN
2	2021/09/01 5:27:28 PM	1	NaN
3	2021/09/01 5:21:59 PM	0	No Event
4	2021/09/01 5:09:38 PM	1	NaN

- Both dataframe have three columns (features) named DateTime, Data and Reading .
- The DateTime feature should be treated for better data visualisation (corrected below).

In [6]:
```
Data_Unique_LG = list(LG_floor_df['Data'].unique())
```

```
Data_Unique_3rd = list(Third_floor_df['Data'].unique())
print(f'Lower ground Data column has the following unique enitres  {Data_Unique_LG}')
print(f'Third floor Data column has the following unique enitres  {Data_Unique_3rd}')
```

```
Lower ground Data column has the following unique enitres  [0, 1, 3]
Third floor Data column has the following unique enitres  [0, 3, 1]
```

The three unique entries are the expected output from the sensor.

In [7]:
```
Reading_Unique_LG = list(LG_floor_df['Reading'].unique())
Reading_Unique_3rd = list(Third_floor_df['Reading'].unique())
print(f'Lower ground Reading column has the following unique enitres  {Reading_Unique_LG}')
print(f'Third floor Reading column has the following unique enitres  {Reading_Unique_3rd}')
```

```
Lower ground Reading column has the following unique enitres  ['No Event', nan]
Third floor Reading column has the following unique enitres  ['No Event', nan]
```

- Both Reading coloumn only have ' No Event ' obsersations
- Other observations are ' NaN ' ,this indicates empty slots
- This is not as expected.The colomn should have ' No Event ' ,' Event ' and ' Heartbeat ' corresponding the three unique entries in the Data column.

In [8]:
```
LG_floor_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 360 entries, 0 to 359
Data columns (total 3 columns):
 #   Column    Non-Null Count  Dtype 
---  ------    --------------  ----- 
 0   DateTime  360 non-null    object
 1   Data      360 non-null    int64 
 2   Reading   166 non-null    object
dtypes: int64(1), object(2)
memory usage: 8.6+ KB
```

In [9]:
```
LG_floor_df.isnull().sum()
```

Out[9]:
```
DateTime      0
Data          0
Reading     194
dtype: int64
```

The Lower Ground dataframe has :

- 360 rows of entries
- DateTime and Data column have 360 observations
- Reading column only have 166 observations and the remaining 194 cells are empty

In [10]:
```
Third_floor_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 332 entries, 0 to 331
Data columns (total 3 columns):
 #   Column    Non-Null Count  Dtype 
---  ------    --------------  ----- 
 0   DateTime  332 non-null    object
 1   Data      332 non-null    int64 
 2   Reading   150 non-null    object
dtypes: int64(1), object(2)
memory usage: 7.9+ KB
```

In [11]:
```
Third_floor_df.isnull().sum()
```

Out[11]:
```
DateTime      0
```

```
Data          0
Reading     182
dtype: int64
```

The Lower Ground dataframe has :

- 332 rows of entries
- DateTime and Data column have 332 observations
- Reading coloumn only have 150 observations and the remaining 182 cells are empty

4.2) DataFrame Treatment

4.2.1) DateTime Feature treatment function

The function below will be used to split DateTime and to fill missing values in the Reading column

In [12]:
```
def dataframe_corrector(df):

    ## Creating new columns
    df['Date']=''
    df['Time']=''
    df['AM/PM']=''

    for index,row in df.iterrows():

        ## Splitting the DateTime column into Date ,Time and AM/PM
        date_time = df.iloc[index]['DateTime']
        splited_datetime = date_time.split()
        date_ = splited_datetime[0]
        time_ = splited_datetime[1]
        am_pm = splited_datetime[2]

        df.at[index,'Date']=date_
        df.at[index,'Time']=time_
        df.at[index,'AM/PM']=am_pm

        #### populating empty cell in the Reading column with values correnponding Data column

        if str(df.iloc[index]['Reading'])== 'nan' and df.iloc[index]['Data'] == 0 :
            df.at[index,'Reading'] ='No Event'

        if str(df.iloc[index]['Reading']) == 'nan'  and df.iloc[index]['Data'] == 1 :
            df.at[index,'Reading'] ='Event'
        if str(df.iloc[index]['Reading']) =='nan' and df.iloc[index]['Data'] == 3 :
            df.at[index,'Reading'] ='Heartbeat'



    return df
```

4.2.2) Applying the function to the dataframe

In [13]:
```
dataframe_corrector(LG_floor_df)
LG_floor_df=LG_floor_df.drop(['DateTime'], axis=1)
```

In [14]:
```
dataframe_corrector(Third_floor_df)
Third_floor_df=Third_floor_df.drop(['DateTime'], axis=1)
```

In [15]:
```
LG_floor_df.isnull().sum()
```

Out[15]:
```
Data       0
Reading    0
Date       0
Time       0
AM/PM      0
dtype: int64
```

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

Out[16]:
```
Data       0
Reading    0
Date       0
Time       0
AM/PM      0
dtype: int64
```

Finally both dataframe have no empty cell

5) Data Visualisation

For visualisation data column is redundant , because the sensor output is represented by the Reading column,hence should be dropped

In [17]:
```
## Dropping Data column
LG_floor_df = LG_floor_df.drop(['Data'], axis=1)
Third_floor_df = Third_floor_df.drop(['Data'], axis=1)
```

The Heartbeat observation has no usefull insight and should be ignored

In [18]:
```
## Dropping Heartbeat Observation
LG_floor_df = LG_floor_df[LG_floor_df['Reading'] != 'Heartbeat']
Third_floor_df = Third_floor_df[Third_floor_df['Reading'] != 'Heartbeat']
```

Detailed Data Visualisation is perfomed on Power BI