STAT40800 Midterm Assignment

Exploratory data analysis of the Irish weather

Author: Shubh Gaur(23200555)

In [1]:
```
# Load in necessary packages
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

1. Load in the weather.csv dataset into Python as a pandas DataFrame. Describe the data. How many years of recordings are included? What is the temporal resolution of the data? Which weather measurements are reported? (8 marks)

In [2]:
```
df=pd.read_csv('weather_1819.csv',skiprows=11)
df.head()
```

Out[2]:

	day	month	year	station	maxtp	mintp	rain	wdsp	hg	sun
0	1	jan	2018	Dublin Airport	7.5	3.2	0.6	18.5	41.0	2.7
1	2	jan	2018	Dublin Airport	11.1	3.4	8.4	17.0	54.0	0.8
2	3	jan	2018	Dublin Airport	8.1	4.6	1.3	23.8	51.0	0.9
3	4	jan	2018	Dublin Airport	9.3	3.4	10.7	14.6	48.0	1.3
4	5	jan	2018	Dublin Airport	6.7	-1.0	0.0	6.6	16.0	3.3

In [3]:
```
#Describing the data
df.describe()
```

Out[3]:

	day	year	maxtp	mintp	rain	wdsp	hg	sun
count	2920.000000	2920.000000	2902.000000	2902.000000	2897.000000	2915.000000	2904.000000	2913.000000
mean	15.720548	2018.500000	13.283150	6.432977	3.063583	9.481475	25.443871	3.783797
std	8.797754	0.500086	5.146289	4.368755	5.053881	3.820605	9.278313	3.850012
min	1.000000	2018.000000	-1.800000	-7.000000	0.000000	2.300000	7.000000	0.000000
25%	8.000000	2018.000000	9.500000	3.100000	0.000000	6.500000	19.000000	0.300000
50%	16.000000	2018.500000	12.800000	6.400000	0.700000	8.900000	24.000000	2.600000
75%	23.000000	2019.000000	17.100000	9.600000	4.000000	11.800000	30.000000	6.300000
max	31.000000	2019.000000	32.000000	18.900000	54.600000	28.500000	84.000000	15.900000

In [4]:
```
#Finding no. of unique values for year column
df.year.unique()
```

Out[4]:
```
array([2018, 2019])
```

As can be seen there are two unique values for the year column.Therefore, the dataset contains data for the span of two years i.e 2018 -> 2019

In [5]:
```
#Finding no. of unique values for year column
print(df.month.unique())
print(df.day.unique())
```

```
['jan' 'feb' 'mar' 'apr' 'may' 'jun' 'jul' 'aug' 'sep' 'oct' 'nov' 'dec']
[ 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
 25 26 27 28 29 30 31]
```

In [6]:
```
#changing month names

# importing library for modifying month names
import calendar

#dictionary by month order for sorting
months={'jan':1,'feb':2,'mar':3,'apr':4,'may':5,'jun':6,'jul':7,
        'aug':8,'sep':9,'oct':10,'nov':11,'dec':12}

#creating a new column by mapping it to the dictiionary
df['month_index']=df.month.map(months)

#changing month names
df['month']=df.month_index.apply(lambda a: calendar.month_name[a])

#sorting by month_index
df=df.sort_values('month_index').reset_index(drop=True)
df.head()
```

Out[6]:

	day	month	year	station	maxtp	mintp	rain	wdsp	hg	sun	month_index
0	1	January	2018	Dublin Airport	7.5	3.2	0.6	18.5	41.0	2.7	1
1	14	January	2018	Knock Airport	9.3	3.8	6.6	14.1	38.0	0.1	1
2	15	January	2018	Knock Airport	8.3	1.0	9.9	14.8	36.0	2.2	1
3	16	January	2018	Knock Airport	2.6	-0.8	8.6	18.5	47.0	0.2	1
4	17	January	2018	Knock Airport	8.8	0.7	18.8	16.8	50.0	0.1	1

The temporal resolution for data is **daily** as can be identified by seeing the unique values of columns: month and day which contain all the possible values for the respective columns.

In [7]:
```
df.columns
#note: month_index is a newly added column
```

Out[7]:
```
Index(['day', 'month', 'year', 'station', 'maxtp', 'mintp', 'rain', 'wdsp',
       'hg', 'sun', 'month_index'],
      dtype='object')
```

The reported weather measurements in the dataset are as follows:-
maxtp: Maximum Air Temperature (C)
mintp: Minimum Air Temperature (C)
rain: Precipitation Amount (mm)
wdsp: Mean Wind Speed (knot)
hg: Highest Gust (knot)
sun: Sunshine duration (hours)

2. Determine how many missing values there are in each column of the dataset. Can you think of a reason why these values are missing? Discuss different strategies for filling the missing values, highlighting the advantages and disadvantages of each strategy, in the context of this dataset. (8 marks)

   Note: You do not need to implement any of your suggested strategies.

In [8]:
```
#finding null values for each column in the dataset
df.isnull().sum()
```

Out[8]:
```
day             0
month           0
year            0
station         0
maxtp          18
mintp          18
rain           23
wdsp            5
hg             16
sun             7
month_index     0
dtype: int64
```

There can be many reasons for the missing data in context of the current dataset which are mentioned below:

- Mistakes in data preprocessing.
- Malfunctions and calibration issues with different sensors.
- Transmission problems.
- Extreme conditions such as storm, heavy snowfall, etc. can sometimes interfere with weather station's equipment.
- Maintenance activities at weather stations contribute to service downtime for the affected period.

The different strategies for filling NA values:

- The data can be grouped based on location and month and the mean weather measurement for the particular month can be used to fill the missing values.
  Pros: Higher Accuracy
  Cons: Higher complexity in filling NA values
- The missing data can either be frontfilled or backfilled in all the columns which may or may not work depending on whether the data is ordered based on location and months.
  Pros: Lower complexity in filling NA values
  Cons: Accuracy may be compromised

3. Write code to answer the following questions: (15 marks)
   A. At what station and on what date was the highest wind speed recorded?
   B. At what station and on what date was the highest maximum air temperature recorded?
   C. At what station and on what date was the largest amount of rain recorded?

A

In [9]:
```
#finding row index which has the maximum: windspeed
maxIndex=df.wdsp.idxmax()
print('The highest windspeed with magnitude of',
      df.loc[maxIndex,'wdsp'],'knots was recorded on',
      str(df.loc[maxIndex,'day']),df.loc[maxIndex,'month'],
      (df.loc[maxIndex,'year']),'at',df.loc[maxIndex,'station']
      +'.')

df.loc[maxIndex]
```

```
The highest windspeed with magnitude of 28.5 knots was recorded on 2 March 2018 at
Dublin Airport.
```

Out[9]:
```
day                         2
month                   March
year                     2018
station        Dublin Airport
maxtp                    -0.2
mintp                    -1.2
rain                      5.6
wdsp                     28.5
hg                       50.0
sun                       0.0
month_index                 3
Name: 678, dtype: object
```

B

In [10]:
```
#finding row index which has the maximum: maxtp
maxIndex=df.maxtp.idxmax()
print('The highest air temperature with magnitude of',
      str(df.loc[maxIndex,'maxtp'])+'°C','was recorded on',
      str(df.loc[maxIndex,'day']),df.loc[maxIndex,'month'],
      str(df.loc[maxIndex,'year']),
      'at',df.loc[maxIndex,'station']
      +'.')

df.loc[maxIndex]
```

```
The highest air temperature with magnitude of 32.0°C was recorded on 28 June 2018
at Shannon Airport.
```

Out[10]:
```
day                         28
month                     June
year                      2018
station        Shannon Airport
maxtp                     32.0
mintp                     12.4
rain                       0.0
wdsp                       4.9
hg                        20.0
sun                       15.6
month_index                  6
Name: 1391, dtype: object
```

C

In [11]:
```
#finding row index which has the maximum windspeed
maxIndex=df.rain.idxmax()
print('The largest rainfall with magnitude of',
      str(df.loc[maxIndex,'rain'])+'mm','was recorded on',
      str(df.loc[maxIndex,'day']),df.loc[maxIndex,'month'],
      str(df.loc[maxIndex,'year']),'at',df.loc[maxIndex,'station']
      +'.')

df.loc[maxIndex]
```

```
The largest rainfall with magnitude of 54.6mm was recorded on 15 April 2019 at Cor
k Airport.
```

Out[11]:
```
day                      15
month                 April
year                   2019
station        Cork Airport
maxtp                   8.3
mintp                   5.2
rain                   54.6
wdsp                   19.6
hg                     45.0
sun                     0.0
month_index               4
Name: 920, dtype: object
```

4. Create a numerical summary (mean, standard deviation, minimum, maximum, etc.) for each of the weather measurements. Discuss and interpret your results. (8 marks)

Numerical Summary for maxtp

In [12]:
```
print('Number of rows in the dataset:-',df.shape[0])
```

```
Number of rows in the dataset:- 2920
```

In [13]:
```
df.maxtp.describe()
```

Out[13]:
```
count    2902.000000
mean       13.283150
std         5.146289
min        -1.800000
25%         9.500000
50%        12.800000
75%        17.100000
max        32.000000
Name: maxtp, dtype: float64
```

According to the numerical summary of maximum temperature values, we interpret:

- There are 2902 datapoints for the maximum temperature values in a day.
- Average maximum temperature for the day throughout the 2 years timeframe is approximately 13.2°C.
- There is some variation in data as can be seen from the standard deviation value which is approximately 5.14°C and it suggests that daily maximum temperatures highly vary which is justified by Ireland's variable climates.
- The minimum value amongst maximum temperature values in a day is -1.8°C.
- The maximum value amongst maximum temperature values in a day is 32°C.
- The median temperature amongst maximum temperature values in a day is 12.8°C.

Numerical Summary for mintp

In [14]:
```
df.mintp.describe()
```

Out[14]:
```
count    2902.000000
mean        6.432977
std         4.368755
min        -7.000000
25%         3.100000
50%         6.400000
75%         9.600000
max        18.900000
Name: mintp, dtype: float64
```

According to the numerical summary of minimum temperature values, we interpret:

- There are 2902 datapoints for the minimum temperature values in a day.
- Average maximum temperature for the day throughout the 2 years timeframe is approximately 6.4°C.
- There is some variation in data as can be seen from the standard deviation value which is approximately 4.36°C and it suggests that daily minimum temperatures highly vary but not as much as maximum temperatures which is justified by Ireland's variable climates.
- The minimum value amongst minimum temperature values in a day is -7°C.
- The maximum value amongst minimum temperature values in a day is 18°C.
- The median temperature amongst maximum temperature values in a day is 6.4°C.

Numerical Summary for rain

In [15]:
```
df.rain.describe()
```

Out[15]:
```
count    2897.000000
mean        3.063583
std         5.053881
min         0.000000
25%         0.000000
50%         0.700000
75%         4.000000
max        54.600000
Name: rain, dtype: float64
```

According to the numerical summary of rainfall values, we interpret:

- There are 2897 datapoints for the rainfall values in a day.
- Average rainfall for the day throughout the 2 years timeframe is approximately 3 mm.
- There is some variation in data as can be seen from the standard deviation value which is approximately 5.05 mm and it suggests that variations in daily rainfall is significantly low due to the fact that Ireland offers a relatively mild and temperate climate, characterised by frequent and relatively even rainfall throughout the year. Also, Ireland has seasonal variations i.e higher rainfall mostly occurring during winter months which can contribute to the standard deviation values.
- The minimum value amongst rainfall values in a day is 0 mm which makes sense as rainfall amount cannot be negative.
- The maximum value amongst minimum temperature values in a day is 54.6 mm.
- The median rainfall amongst rainfall values in a day is 0.7 mm.

Numerical Summary for wdsp

In [16]:
```
df.wdsp.describe()
```

Out[16]:
```
count    2915.000000
mean        9.481475
std         3.820605
min         2.300000
25%         6.500000
50%         8.900000
75%        11.800000
max        28.500000
Name: wdsp, dtype: float64
```

According to the numerical summary of windspeed values, we interpret:

- There are 2915 datapoints for the windspeed values in a day.
- Average windspeed for the day throughout the 2 years timeframe is approximately 9.4 knots.
- There is some variation in data as can be seen from the standard deviation value which is approximately 3.82 knots and this suggests that windspeeds may vary a noticeable amount from the mean which may be due to the fact that Ireland exhibits changing wind patterns as its climate is influenced by Atlantic ocean.
- The minimum value amongst windspeed values in a day is 2.3 knots.
- The maximum value amongst windspeed values in a day is 28.5 knots.
- The median windspeed amongst windspeed values in a day is 8.9 knots.

Numerical Summary for hg

In [17]:
```
df.hg.describe()
```

Out[17]:
```
count    2904.000000
mean       25.443871
std         9.278313
min         7.000000
25%        19.000000
50%        24.000000
75%        30.000000
max        84.000000
Name: hg, dtype: float64
```

According to the numerical summary of highest gust values, we interpret:

- There are 2904 datapoints for the highest gust values in a day.
- Average highest gust for the day throughout the 2 years timeframe is approximately 25.44 knots.
- There is some variation in data as can be seen from the standard deviation value which is approximately 9.27 knots and it suggests that wind gusts can vary significantly from day to day and this may be due to the fact that wind gusts are highly influenced by ocean currents and Ireland's close proximity to Atlantic ocean seems to do the job well.
- The minimum value amongst highest gust values in a day is 7 knots.
- The maximum value amongst highest gust values in a day is 84 knots.
- The median value amongst highest gust values in a day is 24 knots.

Numerical Summary for sun

In [18]:
```
df.sun.describe()
```

Out[18]:
```
count    2913.000000
mean        3.783797
std         3.850012
min         0.000000
25%         0.300000
50%         2.600000
75%         6.300000
max        15.900000
Name: sun, dtype: float64
```

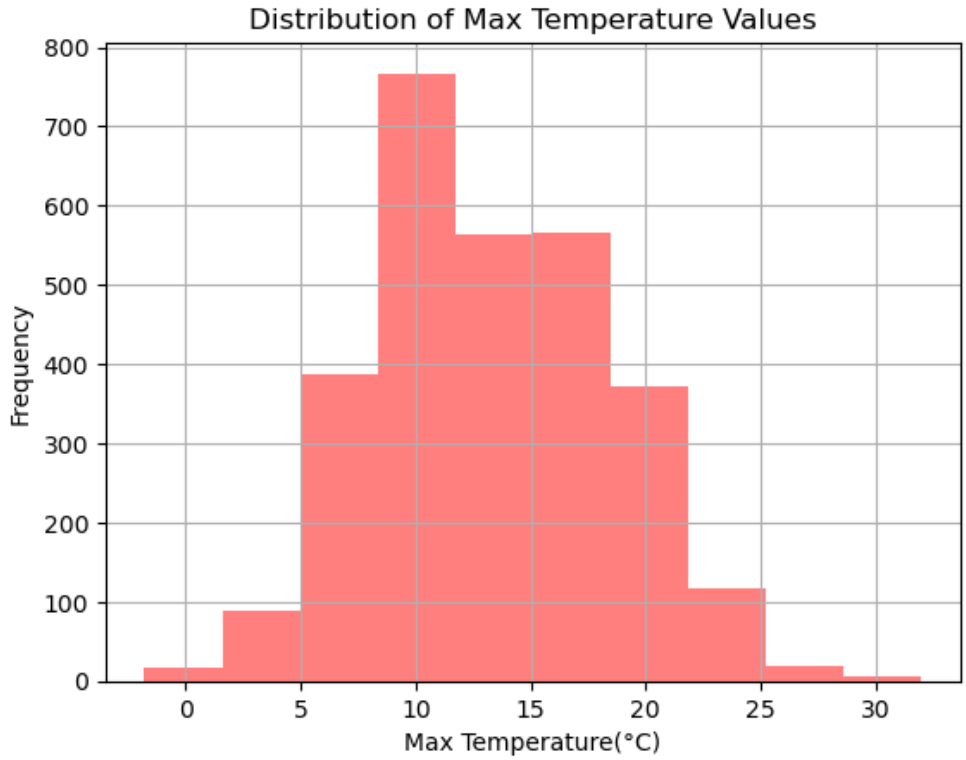
According to the numerical summary of sunshine duration values, we interpret:

- There are 2913 datapoints for the sunshine duration values in a day.
- Average sunshine duration for the day throughout the 2 years timeframe is approximately 3.7 hrs.
- There is some variation in data as can be seen from the standard deviation value which is approximately 3.85 hours which is high relative to the mean which suggests that some days may have extended hours of sunshine while other days may receive lesser or even no sunshine at all.
- The minimum value amongst sunshine duration values in a day is 0 hours which makes sense as sunshine duration cannot be negative.
- The maximum value amongst sunshine duration values in a day is 15 hours.
- The median sunshine duration amongst sunshine duration values in a day is 15.9 hours.

5. Create a graphical summary for each of the weather measurements. Discuss your plots in relation to the summary statistics found in question 4. (10 marks)

Graphical Summary for maxtp

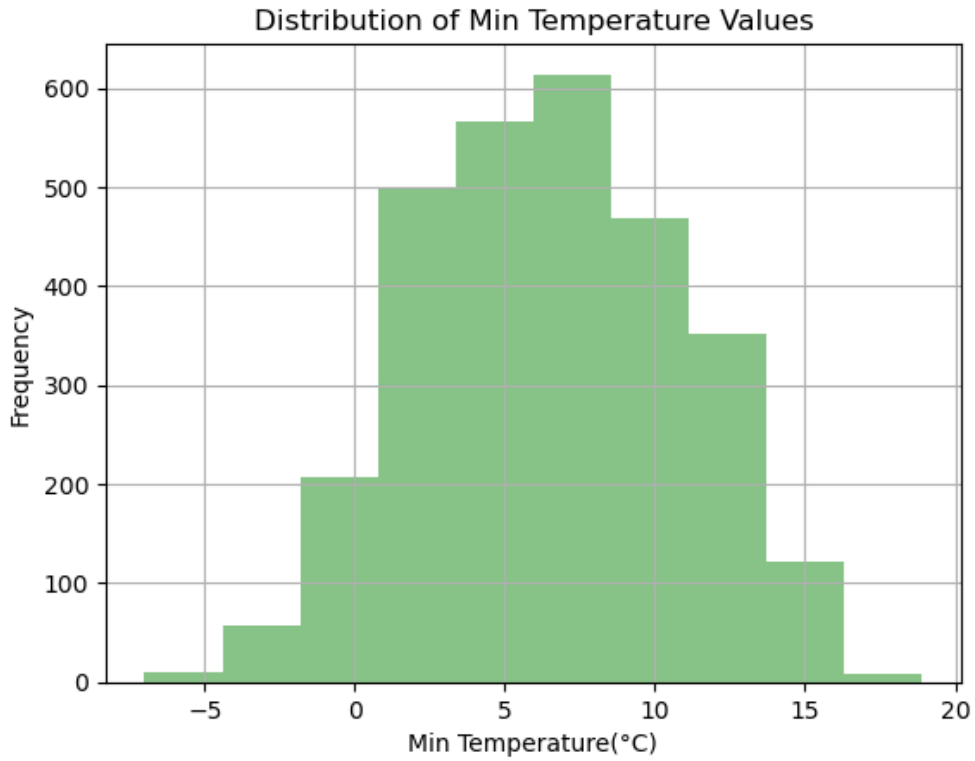
In [19]:
```
df.maxtp.hist(color='r',alpha=0.5)
plt.title('Distribution of Max Temperature Values')
plt.xlabel('Max Temperature(°C)')
plt.ylabel('Frequency')
plt.show()
```



- The minimum value amongst maximum temperature values in a day is a bit lesser than 0°C which is consistent with the minimum value in numerical summary i.e 32°C.
- The maximum value amongst maximum temperature values in a day is a little bit greater than 30°C which is consistent with the maximum value in numerical summary i.e 32°C.
- The median value for maximum temperature in a day as can be seen from the histogram is between 10 and 15 amongst maximum temperature values in a day as some temperature value between the specified boundaries will divide the data distribution into two equal halves which is consistent with the numerical summary with median max temperature value of 12.8°C.

Graphical Summary for mintp

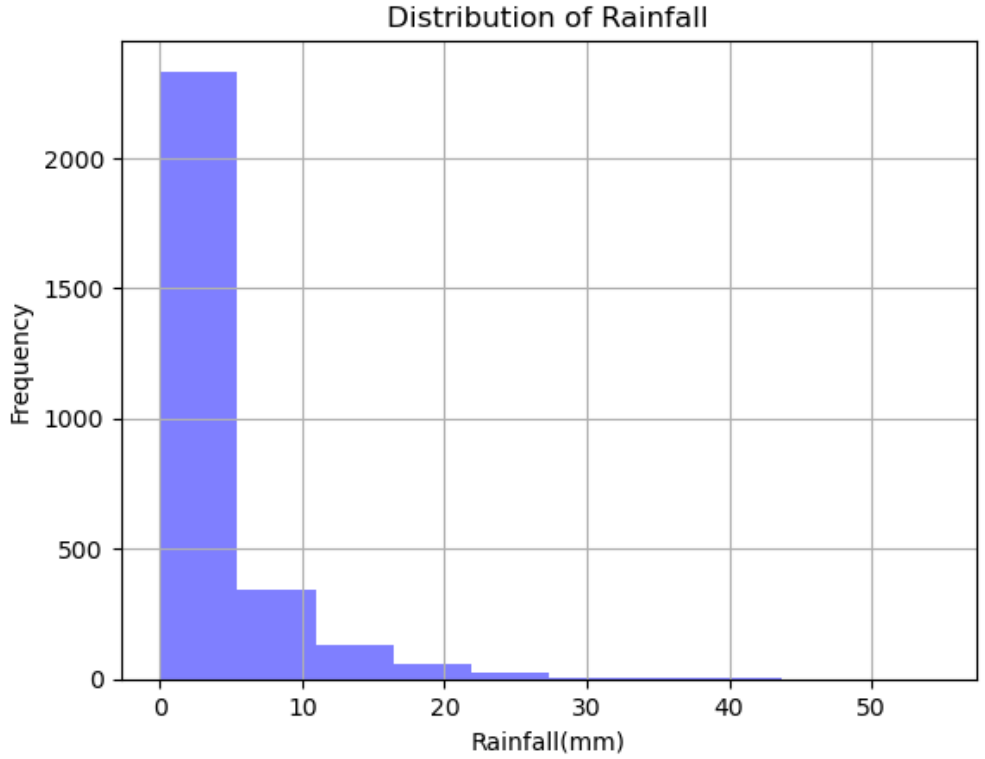
In [20]:
```
df.mintp.hist(color='g',alpha=0.47)
plt.title('Distribution of Min Temperature Values')
plt.xlabel('Min Temperature(°C)')
plt.ylabel('Frequency')
plt.show()
```



- The minimum value amongst minimum temperature values in a day is a bit lesser than -5°C which is consistent with the minimum value in numerical summary i.e -7°C.
- The maximum value amongst minimum temperature values in a day is between 15°C and 20°C which is consistent with the maximum value in numerical summary i.e 18.9°C.
- The median value for minimum temperature in a day as can be seen from the histogram is between 5°C and 10°C amongst minimum temperature values in a day as some temperature value between the specified boundaries will divide the data distribution into two equal halves which is consistent with the numerical summary with median min temperature value of 6.4°C.

Graphical Summary for rain

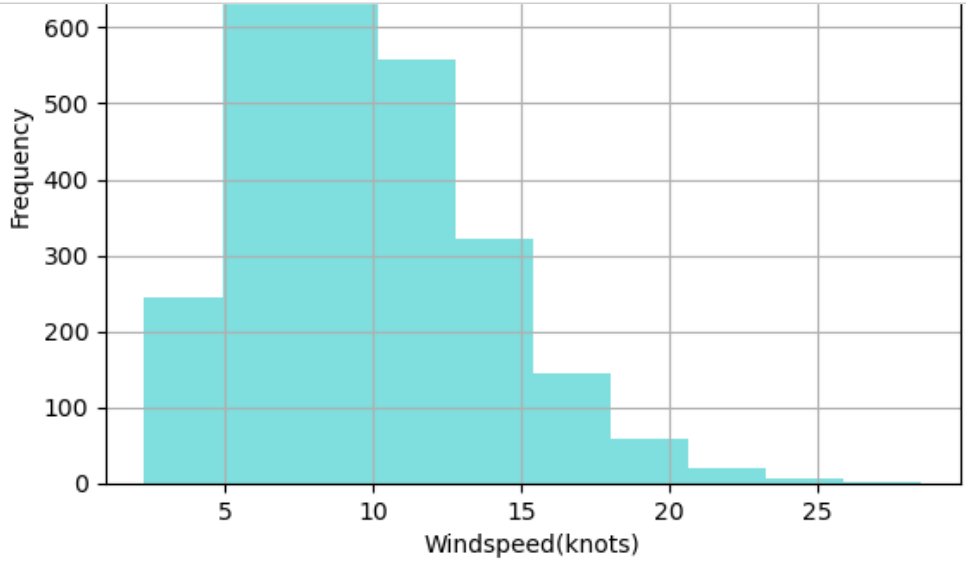
In [21]:
```
df.rain.hist(color='b',alpha=0.5)
plt.title('Distribution of Rainfall')
plt.xlabel('Rainfall(mm)')
plt.ylabel('Frequency')
plt.show()
```



- The minimum value amongst rainfall values in a day is 0 mm which is consistent with the minimum value in numerical summary.
- The maximum value amongst rainfall values in a day is greater than 50 mm which is consistent with the maximum value in numerical summary i.e 54.6 mm.
- The median value for rainfall in a day as can be seen from the histogram is between 0mm and 5mm amongst rainfall values in a day as some rainfall value between the specified boundaries will divide the data distribution into two equal halves which is consistent with the numerical summary with median rainfall value of 0.7 mm.

Graphical Summary for wdsp

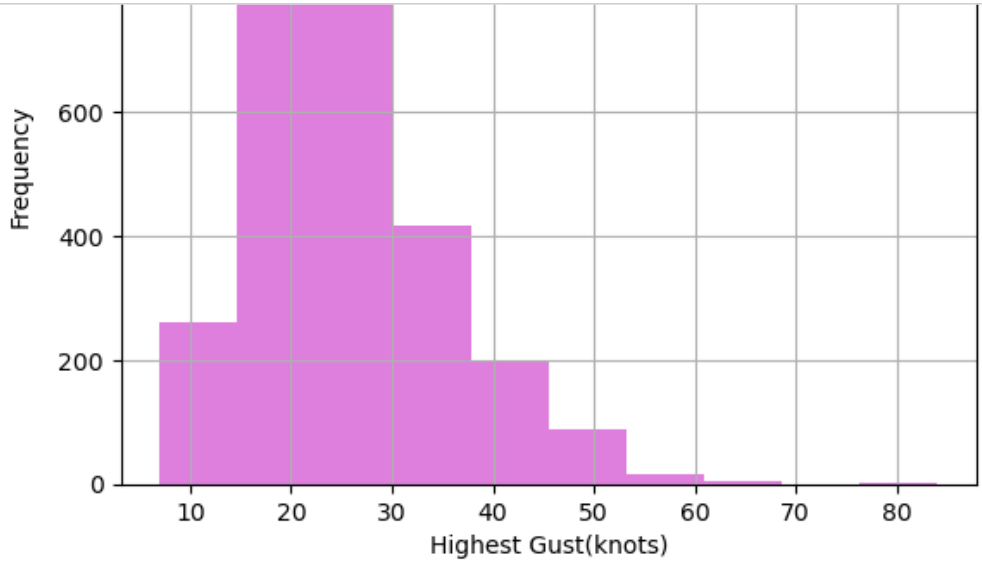
In [22]:
```
df.wdsp.hist(color='c',alpha=0.5)
plt.title('Distribution of Windspeed')
plt.xlabel('Windspeed(knots)')
plt.ylabel('Frequency')
plt.show()
```



- The minimum value amongst windspeed values in a day is between 0 and 5 knots which is consistent with the minimum value in numerical summary i.e 2.3 knots .
- The maximum value amongst windspeed values in a day is greater than 25 knots which is consistent with the maximum value in numerical summary i.e 28.5 knots.
- The median value for windspeed in a day as can be seen from the histogram is between 5 and 10 knots amongst windspeed values in a day as some windspeed value between the specified boundaries will divide the data distribution into two equal halves which is consistent with the numerical summary with median windspeed value of 8.9 knots.

Graphical Summary for hg

In [23]:
```
df.hg.hist(color='m',alpha=0.5)
plt.title('Distribution of Highest Gust')
plt.xlabel('Highest Gust(knots)')
plt.ylabel('Frequency')
plt.show()
```

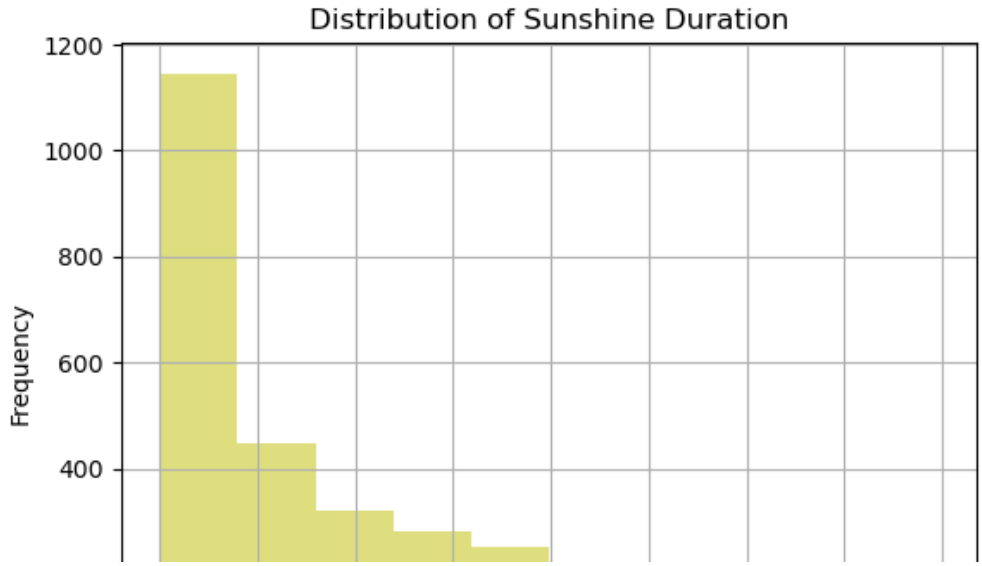


- The minimum value amongst highest gust values in a day is lesser than 10 knots which is consistent with the minimum value in numerical summary i.e 7 knots .

- The maximum value amongst highest gust values in a day is greater than 80 knots which is consistent with the maximum value in numerical summary i.e 84 knots.
- The median value for highest gust in a day as can be seen from the histogram is between 20 and 30 knots amongst highest gust values in a day as some highest gust value between the specified boundaries will divide the data distribution into two equal halves which is consistent with the numerical summary with median highest gust value of 24 knots

Graphical Summary for sun

In [24]:
```
df.sun.hist(color='y',alpha=0.5)
plt.title('Distribution of Sunshine Duration')
plt.xlabel('Sunshine Duration(Hours)')
plt.ylabel('Frequency')
plt.show()
```



- The minimum value amongst sunlight duration values in a day is 0 hours which is consistent with the minimum value in numerical summary.
- The maximum value amongst sunlight duration values in a day is between 14 and 16 hours which is consistent with the maximum value in numerical summary i.e 15.9 hours.
- The median value for sunlight duration in a day as can be seen from the histogram is between 2 and 4 hours amongst sunlight duration values in a day as some sunlight duration value between the specified boundaries will divide the data distribution into two equal halves which is consistent with the numerical summary with median sunlight duration value of 2.6 hours.

6. Produce a scatter plot of the mean wind speed versus the highest gust and colour your points based on month. Interpret your plot. (8 marks)

In [25]:
```
df
```

Out[25]:

	day	month	year	station	maxtp	mintp	rain	wdsp	hg	sun	month_index
0	1	January	2018	Dublin Airport	7.5	3.2	0.6	18.5	41.0	2.7	1
1	14	January	2018	Knock Airport	9.3	3.8	6.6	14.1	38.0	0.1	1
2	15	January	2018	Knock Airport	8.3	1.0	9.9	14.8	36.0	2.2	1
3	16	January	2018	Knock Airport	2.6	-0.8	8.6	18.5	47.0	0.2	1
4	17	January	2018	Knock Airport	8.8	0.7	18.8	16.8	50.0	0.1	1
...	...	...	...	...	...	...	...	...	...	...	...
2915	15	December	2018	Cork Airport	11.9	2.3	19.5	15.5	50.0	0.0	12
2916	14	December	2018	Cork Airport	10.4	7.1	16.1	15.1	32.0	0.3	12
2917	13	December	2018	Cork Airport	8.8	7.1	18.6	20.0	45.0	0.0	12
2918	24	December	2018	Cork Airport	10.5	7.7	4.2	9.2	23.0	0.0	12
2919	31	December	2019	Knock Airport	6.0	1.3	0.0	7.1	18.0	2.1	12

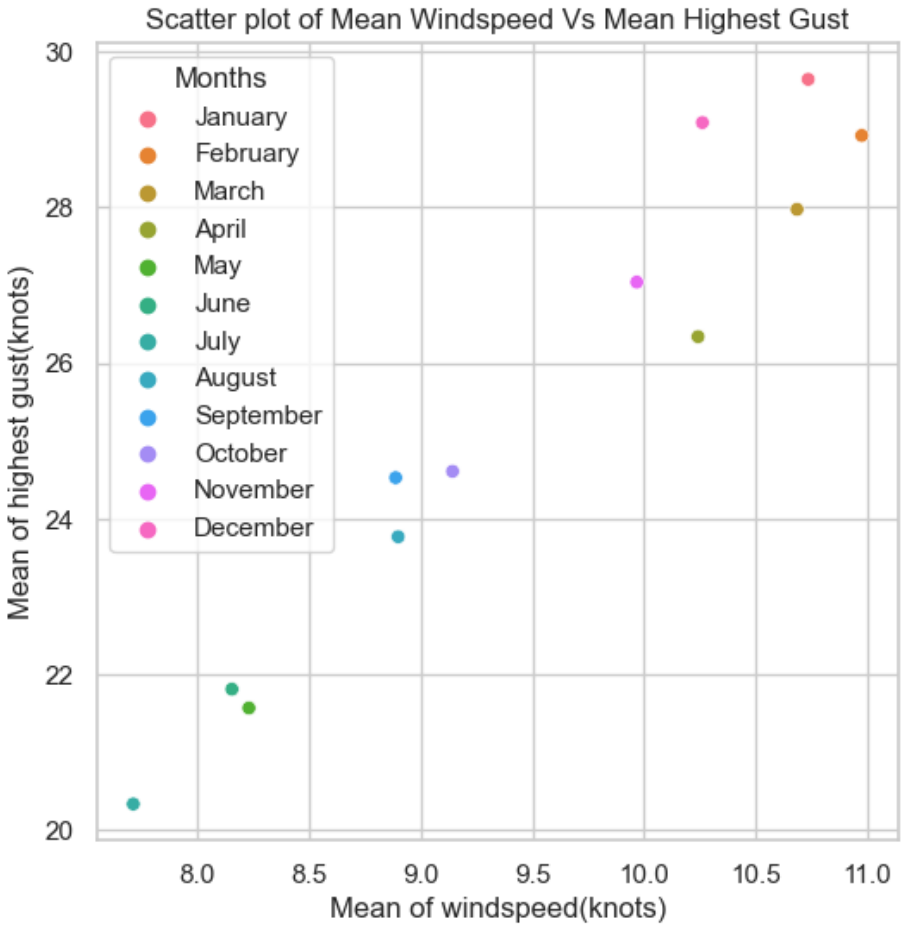
2920 rows × 11 columns

In [26]:
```
# Group by 'Category' and calculate the means of 'wdsp' and 'hg'
grouped_means = df.groupby('month')[['wdsp', 'hg','month_index']].mean().sort_index

#changing month names
grouped_means['month']=grouped_means.month_index.apply(lambda a: calendar.month_name

#sorting by month_index
grouped_means=grouped_means.sort_values('month_index').drop(columns=['month_index']
grouped_means
```

Out[26]:

	wdsp	hg	month
0	10.734274	29.641129	January
1	10.974107	28.919643	February
2	10.684274	27.971660	March
3	10.241250	26.337500	April
4	8.230769	21.570850	May
5	8.154583	21.812500	June
6	7.712955	20.338710	July
7	8.898790	23.769231	August
8	8.887500	24.527426	September
9	9.142276	24.609053	October
10	9.967083	27.037657	November
11	10.260729	29.086066	December

In [27]:
```
# Create a scatter plot using Seaborn
sns.set(style="whitegrid")

plt.figure(figsize=(6,6))
sns.scatterplot(data=grouped_means, x='wdsp', y='hg',
                hue='month', legend='full')

# Add a legend
plt.legend(title='Months')

plt.xlabel('Mean of windspeed(knots)')
plt.ylabel('Mean of highest gust(knots)')
plt.title('Scatter plot of Mean Windspeed Vs Mean Highest Gust')
plt.grid(True)
plt.show()
```



As can be seen in the plot:-

- The mean windspeed and highest gust tend to follow a positive linear relationship.
- The mean windspeed as well as highest gust values are highest during winter months(November-March) and lowest in summer months(June-July)
- The windspeed and highest gust values are moderate in the period from August-October.

7. Compute the daily temperature range, and add this as an additional variable to your DataFrame. Print out the last 10 rows of your DataFrame to show that the column has been added correctly. (5 marks)

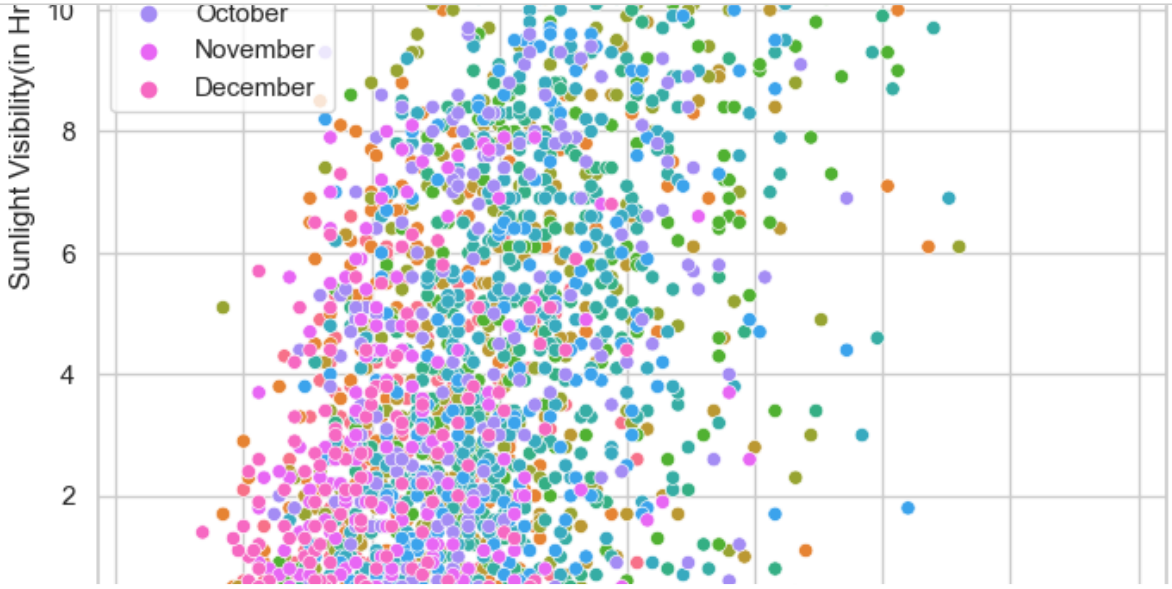
In [28]:
```
# Adding temperature_range column to dataframe
df['temperature_range']=df.maxtp-df.mintp
df.tail(10)
```

Out[28]:

	day	month	year	station	maxtp	mintp	rain	wdsp	hg	sun	month_index	temperature_range
2910	20	December	2018	Cork Airport	10.0	4.4	4.3	9.0	28.0	2.2	12	5.6
2911	19	December	2018	Cork Airport	7.9	3.1	6.7	11.8	27.0	1.6	12	4.8
2912	18	December	2018	Cork Airport	11.0	3.7	11.0	NaN	52.0	3.0	12	7.3
2913	17	December	2018	Cork Airport	11.1	2.3	1.8	16.5	44.0	0.0	12	8.8
2914	16	December	2018	Cork Airport	7.8	1.8	5.8	6.5	23.0	3.1	12	6.0
2915	15	December	2018	Cork Airport	11.9	2.3	19.5	15.5	50.0	0.0	12	9.6
2916	14	December	2018	Cork Airport	10.4	7.1	16.1	15.1	32.0	0.3	12	3.3
2917	13	December	2018	Cork Airport	8.8	7.1	18.6	20.0	45.0	0.0	12	1.7
2918	24	December	2018	Cork Airport	10.5	7.7	4.2	9.2	23.0	0.0	12	2.8
2919	31	December	2019	Knock Airport	6.0	1.3	0.0	7.1	18.0	2.1	12	4.7

8. Plot the daily temperature range versus the hours of sunlight per day, colouring the points based on month. Interpret your plot. (8 marks)

In [29]:
```
#Plotting scatterplot b/w temperature range vs sunlight hours
sns.set(style="whitegrid")

plt.figure(figsize=(8,8))

sns.scatterplot(data=df, x='temperature_range', y='sun',
                hue='month', legend='full')


plt.xlabel('Temperature Range(°C)')
plt.ylabel('Sunlight Visibility(in Hrs)')
plt.title('Scatter plot of temperature range vs. sunlight hours')
plt.grid(True)
plt.legend(title='Months')
plt.show()
```



As can be seen in the plot:

- Sunlight visibility seems to be in a positive linear relationship with temperature range.
- Sunlight visibility is lowest in the winter months(November-February) and highest in summer months(May-July)
- Temperature Range or the variability of temperature over a day is highest in summer months and low-medium during winter months.

9. Perform a comparative analysis of the weather at Dublin Airport, Shannon Airport and Cork Airport. (20 marks)

   For full marks on this question you should create numerical and graphical summaries of the weather measurements at each weather station and discuss how the weather differs (or is similar) across these locations.

In [30]:
```
#defining a dictionary to store dataframes using keys as station names
station_dict={'Dublin Airport': None,'Shannon Airport':None,
              'Cork Airport':None}

for i in station_dict:
    station_dict[i]=df[df.station==i]
```

In [31]:
```
station_dict['Dublin Airport'].head()
```

Out[31]:

	day	month	year	station	maxtp	mintp	rain	wdsp	hg	sun	month_index	temperature_range
0	1	January	2018	Dublin Airport	7.5	3.2	0.6	18.5	41.0	2.7	1	4.3
19	31	January	2019	Dublin Airport	4.0	-5.8	4.0	11.7	31.0	0.0	1	9.8
20	30	January	2019	Dublin Airport	3.4	-3.4	0.0	8.5	27.0	6.6	1	6.8
21	29	January	2019	Dublin Airport	3.9	-0.4	3.0	11.7	28.0	3.7	1	4.3
22	28	January	2019	Dublin Airport	6.2	-0.7	0.0	9.0	20.0	4.8	1	6.9

In [32]:
```
station_dict['Shannon Airport'].head()
```

Out[32]:

	day	month	year	station	maxtp	mintp	rain	wdsp	hg	sun	month_index	temperature_range
41	18	January	2019	Shannon Airport	10.1	4.2	3.6	8.0	27.0	5.1	1	5.9
110	30	January	2019	Shannon Airport	4.8	-0.1	0.9	4.8	18.0	6.2	1	4.9
111	31	January	2019	Shannon Airport	5.6	-0.3	10.0	13.4	32.0	1.0	1	5.9
133	29	January	2019	Shannon Airport	5.2	0.8	8.8	8.8	36.0	3.3	1	4.4
135	28	January	2019	Shannon Airport	7.3	3.3	2.5	8.1	28.0	3.9	1	4.0

In [33]:
```
station_dict['Cork Airport'].head()
```

Out[33]:

	day	month	year	station	maxtp	mintp	rain	wdsp	hg	sun	month_index	temperature_range
28	19	January	2019	Cork Airport	8.1	3.9	1.1	5.2	12.0	4.2	1	4.2
29	20	January	2019	Cork Airport	7.2	1.9	0.8	11.3	35.0	5.9	1	5.3
30	21	January	2019	Cork Airport	6.6	-0.1	5.0	9.6	33.0	0.0	1	6.7
31	22	January	2019	Cork Airport	4.6	-0.5	3.8	7.2	22.0	0.9	1	5.1
32	23	January	2019	Cork Airport	8.8	3.6	4.5	5.2	17.0	0.0	1	5.2

Numerical Summary for Dublin Airport

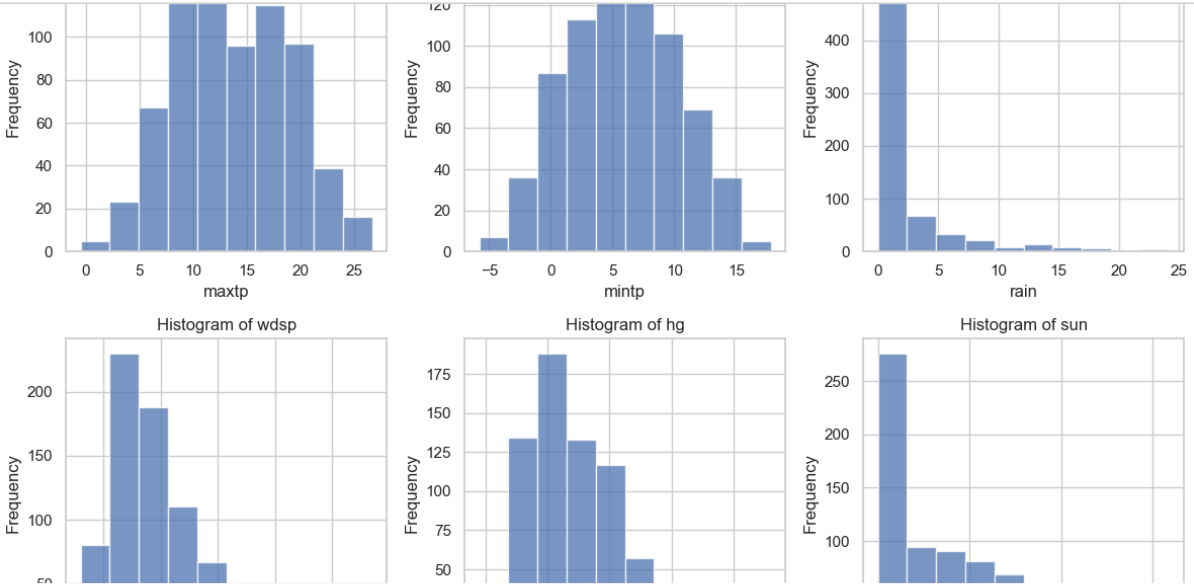
In [34]:
```
#Fetching weather data for Dublin Airport
weather=station_dict['Dublin Airport'][['maxtp','mintp',
                                        'rain','wdsp','hg','sun']]
weather.describe()
```

Out[34]:

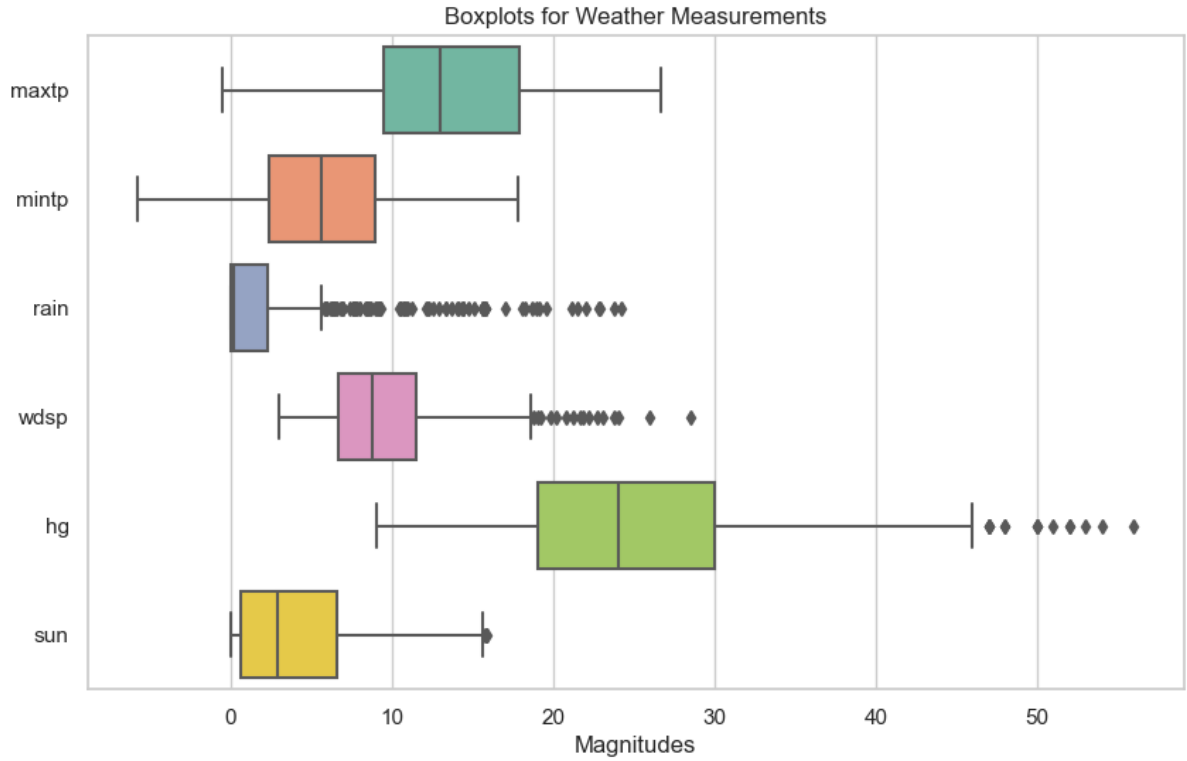
	maxtp	mintp	rain	wdsp	hg	sun
count	730.000000	730.000000	726.000000	728.000000	728.000000	729.000000
mean	13.592603	5.724247	2.147383	9.542720	24.943681	4.046091
std	5.265505	4.488748	4.120571	3.829729	8.438033	3.952654
min	-0.500000	-5.800000	0.000000	3.000000	9.000000	0.000000
25%	9.500000	2.400000	0.000000	6.700000	19.000000	0.600000
50%	13.000000	5.600000	0.200000	8.800000	24.000000	2.900000
75%	17.875000	8.900000	2.300000	11.500000	30.000000	6.600000
max	26.700000	17.800000	24.200000	28.500000	56.000000	15.900000

Distribution for various weather measurements at Dublin airport

In [35]:
```
fig, axes = plt.subplots(2, 3, figsize=(12, 8))

axes = axes.flatten()

for i, column in enumerate(weather.columns):
    ax = axes[i]
    ax.hist(weather[column],alpha=0.75)
    ax.set_title(f'Histogram of {column}')
    ax.set_xlabel(column)
    ax.set_ylabel('Frequency')

# Adjust layout
plt.tight_layout()

plt.show()
```



In [36]:
```
sns.set(style="whitegrid")
plt.figure(figsize=(10, 6))
sns.boxplot(data=weather, orient="h",
            palette="Set2")#horizontally oriented

# Set labels for the boxplots
plt.xlabel("Magnitudes")
plt.title("Boxplots for Weather Measurements")

# Show the plot
plt.show()
```



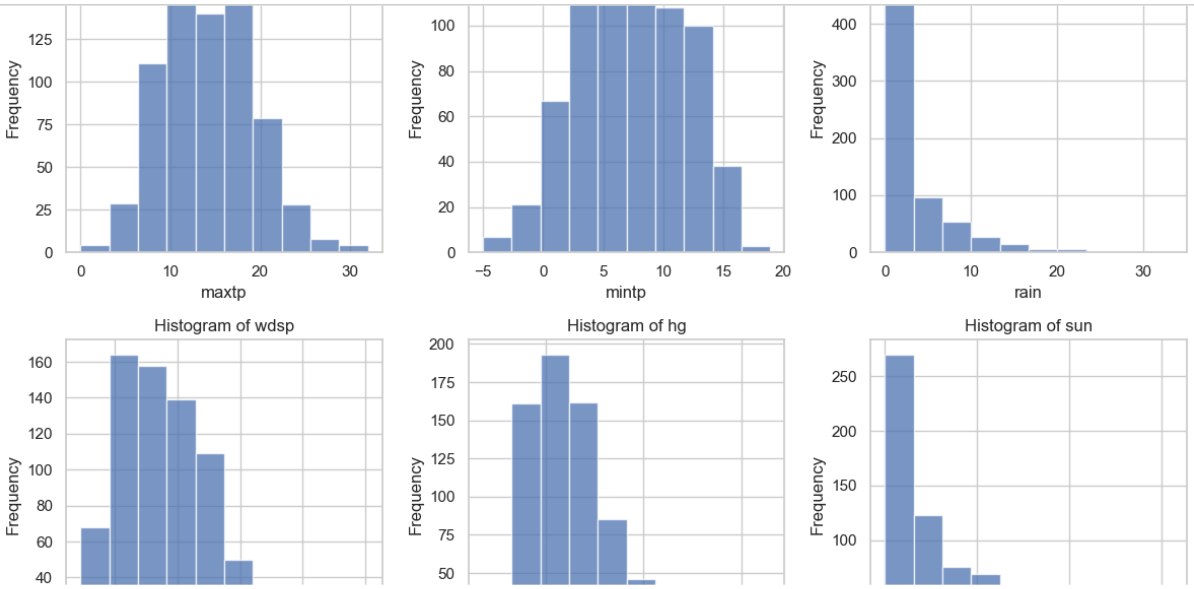
Distribution for various weather measurements at Shannon airport

In [37]:
```
#Fetching weather data for Shannon Airport
weather=station_dict['Shannon Airport'][['maxtp','mintp',
                                         'rain','wdsp','hg','sun']]
weather.describe()
```

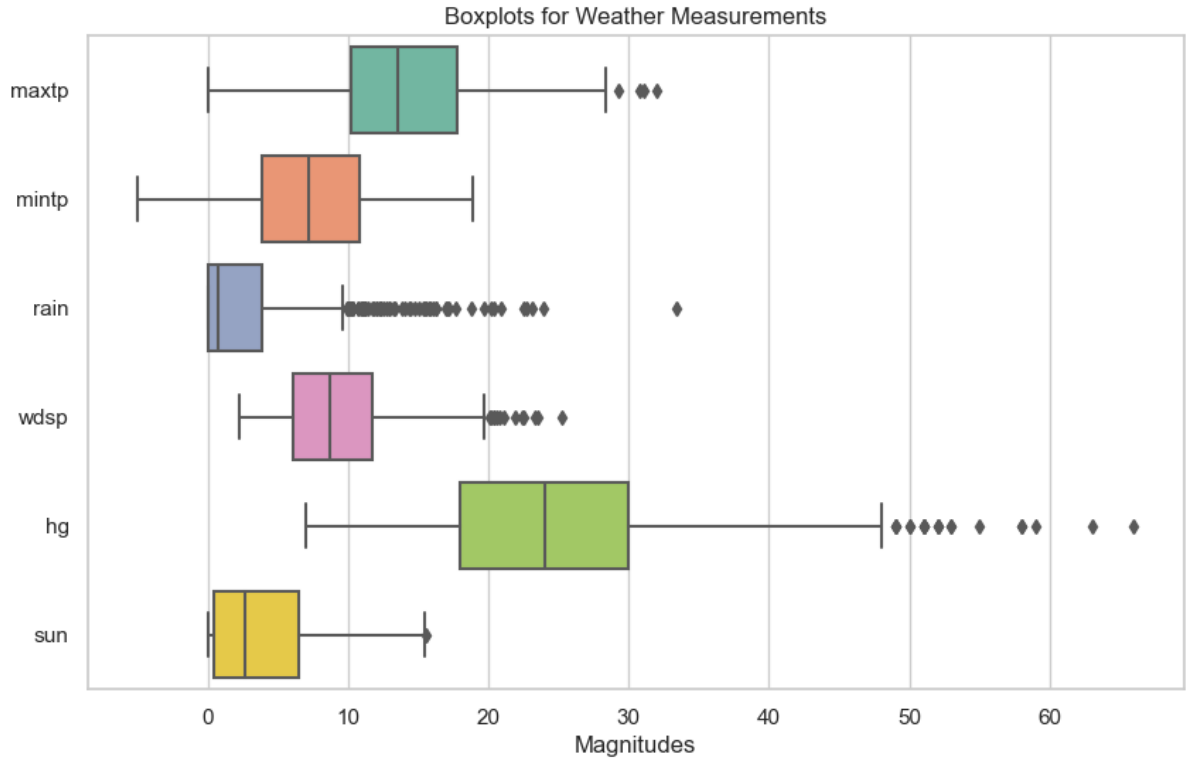
Out[37]:

	maxtp	mintp	rain	wdsp	hg	sun
count	723.000000	723.000000	721.000000	730.000000	724.000000	728.000000
mean	14.160028	7.267082	2.844383	9.264658	24.968232	3.900275
std	5.106169	4.442844	4.402685	4.013313	9.576674	3.883124
min	0.000000	-5.000000	0.000000	2.300000	7.000000	0.000000
25%	10.250000	3.850000	0.000000	6.100000	18.000000	0.500000
50%	13.600000	7.200000	0.800000	8.750000	24.000000	2.700000
75%	17.750000	10.850000	3.900000	11.700000	30.000000	6.500000
max	32.000000	18.900000	33.400000	25.200000	66.000000	15.600000

In [38]:
```
fig, axes = plt.subplots(2, 3, figsize=(12, 8))


axes = axes.flatten()


for i, column in enumerate(weather.columns):
    ax = axes[i]
    ax.hist(weather[column],alpha=0.75)
    ax.set_title(f'Histogram of {column}')
    ax.set_xlabel(column)
    ax.set_ylabel('Frequency')

# Adjust layout
plt.tight_layout()

plt.show()
```



In [39]:
```
sns.set(style="whitegrid")
plt.figure(figsize=(10, 6))
sns.boxplot(data=weather, orient="h",
            palette="Set2")#horizontally oriented

# Set labels for the boxplots
plt.xlabel("Magnitudes")
plt.title("Boxplots for Weather Measurements")

# Show the plot
plt.show()
```



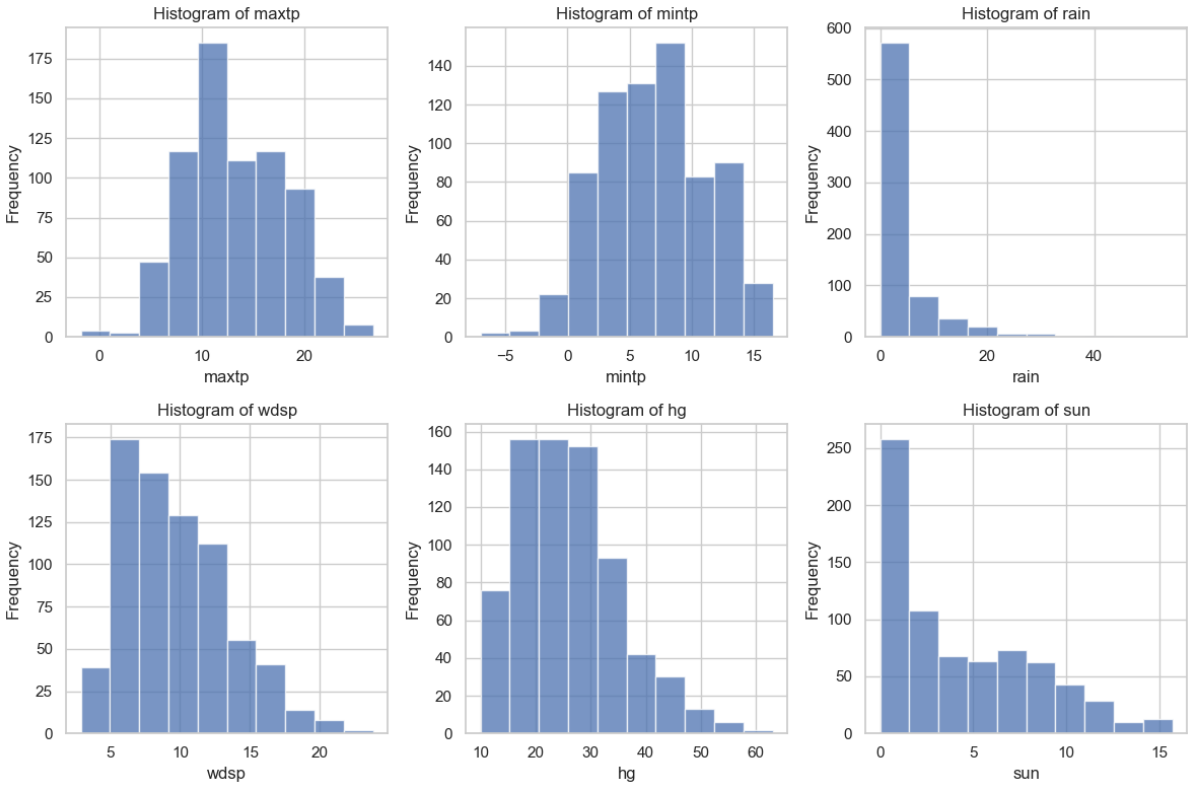
Distribution for various weather measurements at Cork airport

In [40]:
```
#Fetching weather data for Cork Airport
weather=station_dict['Cork Airport'][['maxtp','mintp','rain',
                                      'wdsp','hg','sun']]
weather.describe()
```

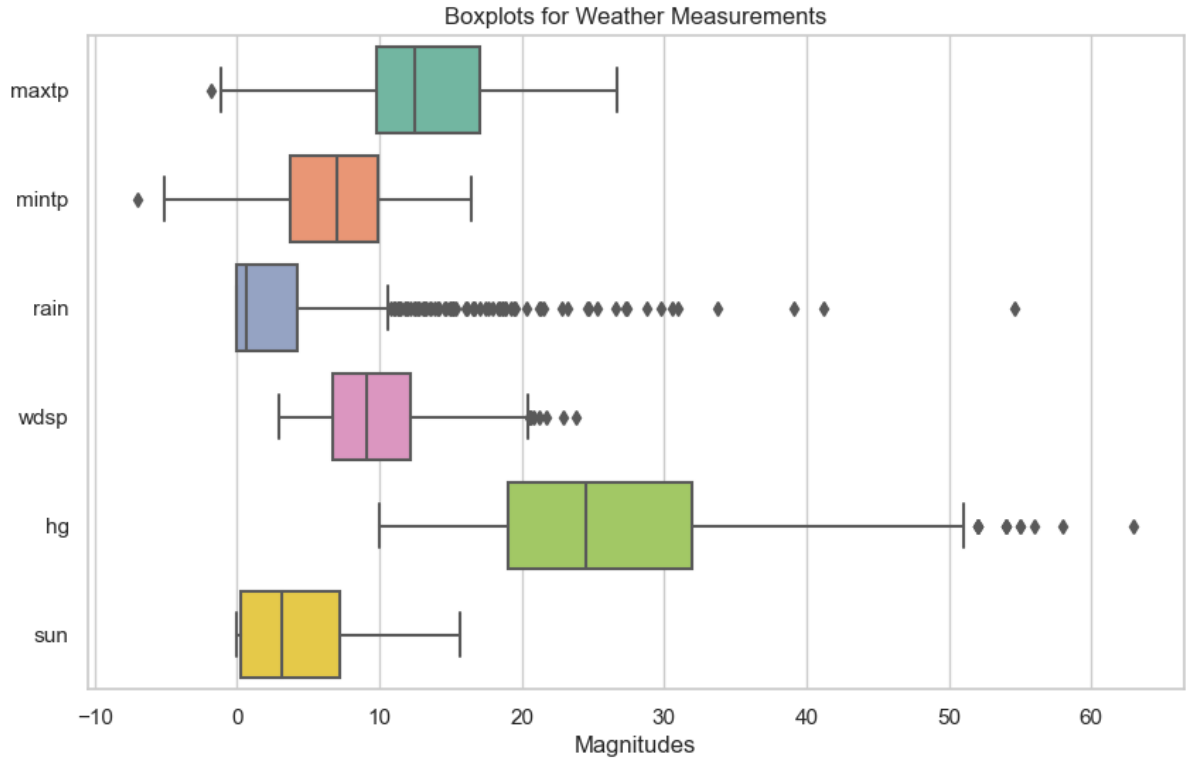
Out[40]:

	maxtp	mintp	rain	wdsp	hg	sun
count	723.000000	723.000000	723.000000	728.000000	726.000000	727.000000
mean	13.262517	6.955325	3.470124	9.758516	25.976584	4.225860
std	4.894583	4.203288	6.137343	3.749587	9.336514	4.055068
min	-1.800000	-7.000000	0.000000	2.900000	10.000000	0.000000
25%	9.750000	3.750000	0.000000	6.700000	19.000000	0.300000
50%	12.500000	7.000000	0.700000	9.100000	24.500000	3.100000
75%	17.100000	9.900000	4.250000	12.200000	32.000000	7.250000
max	26.700000	16.500000	54.600000	23.800000	63.000000	15.700000

In [41]:
```
fig, axes = plt.subplots(2, 3, figsize=(12, 8))

axes = axes.flatten()

for i, column in enumerate(weather.columns):
    ax = axes[i]
    ax.hist(weather[column],alpha=0.75)
    ax.set_title(f'Histogram of {column}')
    ax.set_xlabel(column)
    ax.set_ylabel('Frequency')

# Adjust layout
plt.tight_layout()

plt.show()
```



In [42]:
```
sns.set(style="whitegrid")
plt.figure(figsize=(10, 6))
sns.boxplot(data=weather, orient="h", palette="Set2")#horizontally oriented

# Set labels for the boxplots
plt.xlabel("Magnitudes")
plt.title("Boxplots for Weather Measurements")

# Show the plot
plt.show()
```



Comparative Summary

After observing the numerical and graphical summaries for weather measurements at the given locations, we infer that:-

- All the airports have similar mean values of minimum temperature and maximum temperature in a day which suggests that the mean range of temperature in a day stays mostly stagnant.
- The temperature values fluctuate the most at Dublin airport in comparison with Shannon and Cork airport due to a greater standard deviation in temperature values at Dublin airport.
- Cork airport receives the highest mean rainfall followed by Shannon and Dublin airport.
- The rainfall varies the most at Cork airport due to a greater standard deviation value than its counterparts(Shannon airport,Dublin airport) where the variability in rainfall is almost similar.
- The average windspeed and highest gust at Cork airport is slightly greater than its counterparts(Shannon airport,Dublin airport) where the mean windspeed is almost similar
- The windspeed varies the most at Shannon Airport and windspeed variations at Cork and Dublin airport is almost similar.
- Cork Airport receives the highest average sunlight duration followed by Dublin Airport and Shannon Airport.
- The sunlight duration fluctuates the most at Cork airport followed by Dublin airport and Shannon airport.
- By looking at the histograms of distribution for various weather measurements and different airports we can conclude that the weather at the given locations is similar enough and there is not sufficient evidence(in the given dataset) to prove otherwise.