```
!pip install pandas
!pip install kaggle
```

```
Requirement already satisfied: pandas in /usr/local/lib/python3.11/dist-packages (2.2.2)
Requirement already satisfied: numpy>=1.23.2 in /usr/local/lib/python3.11/dist-packages (from pandas
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.11/dist-packages (fr
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from panda
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-date
Requirement already satisfied: kaggle in /usr/local/lib/python3.11/dist-packages (1.7.4.5)
Requirement already satisfied: bleach in /usr/local/lib/python3.11/dist-packages (from kaggle) (6.2.
Requirement already satisfied: certifi>=14.05.14 in /usr/local/lib/python3.11/dist-packages (from ka
Requirement already satisfied: charset-normalizer in /usr/local/lib/python3.11/dist-packages (from k
Requirement already satisfied: idna in /usr/local/lib/python3.11/dist-packages (from kaggle) (3.10)
Requirement already satisfied: protobuf in /usr/local/lib/python3.11/dist-packages (from kaggle) (5.
Requirement already satisfied: python-dateutil>=2.5.3 in /usr/local/lib/python3.11/dist-packages (fr
Requirement already satisfied: python-slugify in /usr/local/lib/python3.11/dist-packages (from kaggl
Requirement already satisfied: requests in /usr/local/lib/python3.11/dist-packages (from kaggle) (2.
Requirement already satisfied: setuptools>=21.0.0 in /usr/local/lib/python3.11/dist-packages (from k
Requirement already satisfied: six>=1.10 in /usr/local/lib/python3.11/dist-packages (from kaggle) (1
Requirement already satisfied: text-unidecode in /usr/local/lib/python3.11/dist-packages (from kaggl
Requirement already satisfied: tqdm in /usr/local/lib/python3.11/dist-packages (from kaggle) (4.67.1
Requirement already satisfied: urllib3>=1.15.1 in /usr/local/lib/python3.11/dist-packages (from kagg
Requirement already satisfied: webencodings in /usr/local/lib/python3.11/dist-packages (from kaggle)
```

```
import os
from google.colab import files
os.makedirs('/root/.kaggle', exist_ok=True)

print("Please upload your kaggle.json file:")
uploaded = files.upload()

for filename in uploaded.keys():
  os.rename(filename, '/root/.kaggle/' + filename)
os.chmod('/root/.kaggle/kaggle.json', 0o600)

print("\nKaggle API key uploaded and configured successfully!")
print("You can now use the Kaggle API.")
```

```
Please upload your kaggle.json file:
Choose Files kaggle.json
```

- **kaggle.json**(application/json) - 65 bytes, last modified: 6/19/2025 - 100% done

```
Saving kaggle.json to kaggle.json

Kaggle API key uploaded and configured successfully!
You can now use the Kaggle API.
```

```
import pandas as pd
import zipfile
import kaggle
```

```
!kaggle datasets download -d hmavrodiev/london-bike-sharing-dataset
```

```
Dataset URL: https://www.kaggle.com/datasets/hmavrodiev/london-bike-sharing-dataset
License(s): other
Downloading london-bike-sharing-dataset.zip to /content
  0% 0.00/165k [00:00<?, ?B/s]
```

```
100% 165k/165k [00:00<00:00, 601MB/s]
```

```
zipfile_name = 'london-bike-sharing-dataset.zip'
with zipfile.ZipFile(zipfile_name, 'r') as file:
    file.extractall()
```

```
bikes = pd.read_csv("london_merged.csv")
```

```
bikes.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 17414 entries, 0 to 17413
Data columns (total 10 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   timestamp     17414 non-null  object 
 1   cnt           17414 non-null  int64  
 2   t1            17414 non-null  float64
 3   t2            17414 non-null  float64
 4   hum           17414 non-null  float64
 5   wind_speed    17414 non-null  float64
 6   weather_code  17414 non-null  float64
 7   is_holiday    17414 non-null  float64
 8   is_weekend    17414 non-null  float64
 9   season        17414 non-null  float64
dtypes: float64(8), int64(1), object(1)
memory usage: 1.3+ MB
```

```
bikes.shape
```

```
(17414, 10)
```

```
bikes
```

	timestamp	cnt	t1	t2	hum	wind_speed	weather_code	is_holiday	is_weekend	season
0	2015-01-04 00:00:00	182	3.0	2.0	93.0	6.0	3.0	0.0	1.0	3.0
1	2015-01-04 01:00:00	138	3.0	2.5	93.0	5.0	1.0	0.0	1.0	3.0
2	2015-01-04 02:00:00	134	2.5	2.5	96.5	0.0	1.0	0.0	1.0	3.0
3	2015-01-04 03:00:00	72	2.0	2.0	100.0	0.0	1.0	0.0	1.0	3.0
4	2015-01-04 04:00:00	47	2.0	0.0	93.0	6.5	1.0	0.0	1.0	3.0
...	...	...	...	...	...	...	...	...	...	...
17409	2017-01-03 19:00:00	1042	5.0	1.0	81.0	19.0	3.0	0.0	0.0	3.0
17410	2017-01-03 20:00:00	541	5.0	1.0	81.0	21.0	4.0	0.0	0.0	3.0
17411	2017-01-03 21:00:00	337	5.5	1.5	78.5	24.0	4.0	0.0	0.0	3.0

Next steps: Generate code with `bikes` View recommended plots New interactive sheet

```
bikes.weather_code.value_counts()
```

weather_code	count
1.0	6150
2.0	4034
3.0	3551
7.0	2141
4.0	1464
26.0	60
10.0	14

dtype: int64

```
bikes.season.value_counts()
```

season	count
0.0	4394
1.0	4387
3.0	4330
2.0	4303

dtype: int64

```
new_cols_dict ={
    'timestamp':'time',
    'cnt':'count',
    't1':'temp_real_C',
    't2':'temp_feels_like_C',
    'hum':'humidity_percent',
    'wind_speed':'wind_speed_kph',
    'weather_code':'weather',
    'is_holiday':'is_holiday',
    'is_weekend':'is_weekend',
    'season':'season'
}

bikes.rename(new_cols_dict, axis=1, inplace=True)


bikes.humidity_percent = bikes.humidity_percent / 100


season_dict = {
    '0.0':'spring',
    '1.0':'summer',
    '2.0':'autumn',
    '3.0':'winter'
}

weather_dict = {
    '1.0':'Clear',
    '2.0':'Scattered clouds',
    '3.0':'Broken clouds',
    '4.0':'Cloudy',
    '7.0':'Rain',
    '10.0':'Rain with thunderstorm',
    '26.0':'Snowfall'
}

bikes.season = bikes.season.astype('str')
bikes.season = bikes.season.map(season_dict)
bikes.weather = bikes.weather.astype('str')
bikes.weather = bikes.weather.map(weather_dict)


bikes.head()
```

	time	count	temp_real_C	temp_feels_like_C	humidity_percent	wind_speed_kph	weather	is_holi
0	2015-01-04 00:00:00	182	3.0	2.0	0.930	6.0	Broken clouds	
1	2015-01-04 01:00:00	138	3.0	2.5	0.930	5.0	Clear	
2	2015-01-04 02:00:00	134	2.5	2.5	0.965	0.0	Clear	
3	2015-01-04 03:00:00	72	2.0	2.0	1.000	0.0	Clear	
4	2015-01-04 04:00:00	47	2.0	0.0	0.930	6.5	Clear	

Next steps: Generate code with `bikes` View recommended plots New interactive sheet