In [ ]:
```python
import os
# import re

import pandas as pd
```

In [ ]:
```python
script_dir = os.path.dirname(os.path.abspath(__file__))
```

In [ ]:
```python
listings_relative_path = os.path.join("..", "..", "data", "raw", "listings.csv")
neighbourhoods_relative_path = os.path.join(
    "..", "..", "data", "raw", "neighbourhoods.csv"
)

listings_file_path = os.path.abspath(os.path.join(script_dir, listings_relative_path))
neighbourhoods_file_path = os.path.abspath(
    os.path.join(script_dir, neighbourhoods_relative_path)
)
```

In [ ]:
```python
df1 = pd.read_csv(listings_file_path)
df2 = pd.read_csv(neighbourhoods_file_path)

df_listings = pd.DataFrame(df1)
df_neighbourhoods = pd.DataFrame(df2)

df_listings.head()
df_neighbourhoods.head()
```

Out[ ]:

	neighbourhood_group	neighbourhood
0	NaN	Adachi Ku
1	NaN	Akiruno Shi
2	NaN	Akishima Shi
3	NaN	Aogashima Mura
4	NaN	Arakawa Ku

In [ ]:
```python
df_listings.head()
```

Out[ ]:

	id	name	host_id	host_name	neighbourhood_group	neighbourhood	latitude	longitude	room_type	price	minimum_nights	number_of_reviews	last_review	reviews_per_month	calculated_host_listings_count	availability_365	number_of_reviews_ltm
0	197677	Oshiage Holiday Apartment ...	964081	Yoshimi & Marek ...		Sumida Ku	35.717070	139.826080	Entire home/apt	12000	3	176	2024-05-15	1.13	1	223	4
1	776070	Kero-kero house room 1 ...	801494	Kei ...		Kita Ku	35.738440	139.769170	Private room	9652	3	256	2024-06-03	1.81	1	173	13
2	905944	4F - Near Shinjuku & Shibuya ...	4847803	Best Stay In Tokyo! ...		Shibuya Ku	35.678780	139.678470	Entire home/apt	25738	3	219	2024-06-19	1.60	6	115	33
3	1016831	5 mins Shibuya Cat modern sunny Shimokita ...	5596383	Wakana ...		Setagaya Ku	35.658000	139.671340	Private room	23286	10	268	2024-06-26	1.96	2	129	24
4	1196177	Stay with host Cozy private room Senju area ...	5686404	Yukiko ...		Adachi Ku	35.744731	139.797384	Private room	7500	2	120	2024-06-15	0.91	1	36	26

In [ ]:
```python
# Overview of df_listings
df_listings.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 16518 entries, 0 to 16517
Data columns (total 18 columns):
 #   Column                            Non-Null Count  Dtype
---  ------                            --------------  -----
 0   id                                16518 non-null  int64
 1   name                              16518 non-null  object
 2   host_id                           16518 non-null  int64
 3   host_name                         16518 non-null  object
 4   neighbourhood_group               16518 non-null  object
 5   neighbourhood                     16518 non-null  object
 6   latitude                          16518 non-null  float64
 7   longitude                         16518 non-null  float64
 8   room_type                         16518 non-null  object
 9   price                             16518 non-null  object
 10  minimum_nights                    16518 non-null  int64
 11  number_of_reviews                 16518 non-null  int64
 12  last_review                       16518 non-null  object
 13  reviews_per_month                 16518 non-null  object
 14  calculated_host_listings_count    16518 non-null  int64
 15  availability_365                  16518 non-null  int64
 16  number_of_reviews_ltm             16518 non-null  int64
 17  license                           16508 non-null  object
dtypes: float64(2), int64(7), object(9)
memory usage: 2.3+ MB
```

In [ ]:
```python
df_listings.describe()
```

Out[ ]:

	id	host_id	latitude	longitude	minimum_nights	number_of_reviews	calculated_host_listings_count	availability_365	number_of_reviews_ltm
count	1.651800e+04	1.651800e+04	16518.000000	16518.000000	16518.000000	16518.000000	16518.000000	16518.000000	16518.000000
mean	5.816491e+17	2.929799e+08	35.698748	139.736294	3.648868	37.783691	16.109335	154.140756	15.071013
std	4.970662e+17	1.846269e+08	0.041095	0.072995	8.920520	64.014216	19.674860	99.525334	21.505364
min	1.976770e+05	3.222340e+05	35.520940	139.081322	1.000000	0.000000	1.000000	0.000000	0.000000
25%	4.144522e+07	1.290963e+08	35.688305	139.699183	1.000000	4.000000	3.000000	75.000000	2.000000
50%	8.301174e+17	2.721294e+08	35.703990	139.728136	2.000000	17.000000	9.000000	144.000000	10.000000
75%	1.040939e+18	4.923137e+08	35.722587	139.790456	2.000000	45.000000	19.000000	234.000000	22.000000
max	1.189054e+18	5.859819e+08	35.840764	139.914020	365.000000	2660.000000	106.000000	365.000000	815.000000

In [ ]:
```python
# Overview of df_neighbourhoods
df_neighbourhoods.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 62 entries, 0 to 61
Data columns (total 2 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   neighbourhood_group  0 non-null      float64
 1   neighbourhood        62 non-null     object
dtypes: float64(1), object(1)
memory usage: 1.1+ KB
```

In [ ]:
```python
# Cleaning df_listings
listings_clean = df_listings.copy()

print(listings_clean.columns)
```

```
Index(['id                  ',
       'name                                                                      ',
       'host_id   ', 'host_name                                  ',
       'neighbourhood_group ', 'neighbourhood       ', 'latitude          ',
       'longitude          ', 'room_type       ', 'price  ', 'minimum_nights ',
       'number_of_reviews ', 'last_review ', 'reviews_per_month ',
       'calculated_host_listings_count ', 'availability_365 ',
       'number_of_reviews_ltm ', 'license'],
      dtype='object')
```

In [ ]:
```python
listings_clean.columns = listings_clean.columns.str.strip().str.lower()
listings_clean = (
    listings_clean.drop(
        columns=[
            "neighbourhood_group",
            "minimum_nights",
            "number_of_reviews",
            "last_review",
            "reviews_per_month",
            "calculated_host_listings_count",
            "availability_365",
            "number_of_reviews_ltm",
            "license",
        ]
    )
    .drop_duplicates()
    .dropna()
)
```

In [ ]:
```python
listings_clean.head()
```

Out[ ]:

	id	name	host_id	host_name	neighbourhood	latitude	longitude	room_type	price
0	197677	Oshiage Holiday Apartment ...	964081	Yoshimi & Marek ...	Sumida Ku	35.717070	139.826080	Entire home/apt	12000
1	776070	Kero-kero house room 1 ...	801494	Kei ...	Kita Ku	35.738440	139.769170	Private room	9652
2	905944	4F - Near Shinjuku & Shibuya ...	4847803	Best Stay In Tokyo! ...	Shibuya Ku	35.678780	139.678470	Entire home/apt	25738
3	1016831	5 mins Shibuya Cat modern sunny Shimokita ...	5596383	Wakana ...	Setagaya Ku	35.658000	139.671340	Private room	23286
4	1196177	Stay with host Cozy private room Senju area ...	5686404	Yukiko ...	Adachi Ku	35.744731	139.797384	Private room	7500

In [ ]:
```python
# Replacing non-ASCII characters with blank spaces.
listings_clean["name"] = listings_clean["name"].apply(
    lambda x: " " if any(ord(char) > 127 for char in x) else x
)
listings_clean["host_name"] = listings_clean["host_name"].apply(
    lambda x: " " if any(ord(char) > 127 for char in x) else x
)
```

In [ ]:
```python
listings_clean
```

Out[ ]:

	id	name	host_id	host_name	neighbourhood	latitude	longitude	room_type	price
0	197677	Oshiage Holiday Apartment ...	964081	Yoshimi & Marek ...	Sumida Ku	35.717070	139.826080	Entire home/apt	12000
1	776070	Kero-kero house room 1 ...	801494	Kei ...	Kita Ku	35.738440	139.769170	Private room	9652
2	905944	4F - Near Shinjuku & Shibuya ...	4847803	Best Stay In Tokyo! ...	Shibuya Ku	35.678780	139.678470	Entire home/apt	25738
3	1016831	5 mins Shibuya Cat modern sunny Shimokita ...	5596383	Wakana ...	Setagaya Ku	35.658000	139.671340	Private room	23286
4	1196177	Stay with host Cozy private room Senju area ...	5686404	Yukiko ...	Adachi Ku	35.744731	139.797384	Private room	7500
...	...	...	...	...	...	...	...	...	...
16513	1188807027935790805	quiet/morden/shibuya/shinjyuku ...	29209062	Kevin ...	Shibuya Ku	35.677552	139.683639	Entire home/apt	16000
16514	1188807658642978418		585909521	Yi ...	Itabashi Ku	35.755566	139.708938	Entire home/apt	4714
16515	1188862130812517749		325747621	Sakura ...	Ota Ku	35.578456	139.737537	Entire home/apt	13186
16516	1189007613727558111	The FLAT 103 ...	152574527	Takaaki ...	Sumida Ku	35.721761	139.814004	Entire home/apt	10400
16517	1189054292409642178		385187074	Bo ...	Sumida Ku	35.725495	139.829714	Entire home/apt	31314

16518 rows × 9 columns

In [ ]:
```python
df_listings_cleaned = listings_clean.copy()
```

In [ ]:
```python
# Cleaning df_neighbourhoods
neighbourhoods_clean = df_neighbourhoods.copy()
df_neighbourhoods_cleaned = neighbourhoods_clean.drop(columns="neighbourhood_group")
```

In [ ]:
```python
df_neighbourhoods_cleaned
```

Out[ ]:

	neighbourhood
0	Adachi Ku
1	Akiruno Shi
2	Akishima Shi
3	Aogashima Mura
4	Arakawa Ku
...	...
57	Tachikawa Shi
58	Taito Ku
59	Tama Shi
60	Toshima Ku
61	Toshima Mura

62 rows × 1 columns

In [ ]:
```python
df_listings_cleaned
```

Out[ ]:

	id	name	host_id	host_name	neighbourhood	latitude	longitude	room_type	price
0	197677	Oshiage Holiday Apartment ...	964081	Yoshimi & Marek ...	Sumida Ku	35.717070	139.826080	Entire home/apt	12000
1	776070	Kero-kero house room 1 ...	801494	Kei ...	Kita Ku	35.738440	139.769170	Private room	9652
2	905944	4F - Near Shinjuku & Shibuya ...	4847803	Best Stay In Tokyo! ...	Shibuya Ku	35.678780	139.678470	Entire home/apt	25738
3	1016831	5 mins Shibuya Cat modern sunny Shimokita ...	5596383	Wakana ...	Setagaya Ku	35.658000	139.671340	Private room	23286
4	1196177	Stay with host Cozy private room Senju area ...	5686404	Yukiko ...	Adachi Ku	35.744731	139.797384	Private room	7500
...	...	...	...	...	...	...	...	...	...
16513	1188807027935790805	quiet/morden/shibuya/shinjyuku ...	29209062	Kevin ...	Shibuya Ku	35.677552	139.683639	Entire home/apt	16000
16514	1188807658642978418		585909521	Yi ...	Itabashi Ku	35.755566	139.708938	Entire home/apt	4714
16515	1188862130812517749		325747621	Sakura ...	Ota Ku	35.578456	139.737537	Entire home/apt	13186
16516	1189007613727558111	The FLAT 103 ...	152574527	Takaaki ...	Sumida Ku	35.721761	139.814004	Entire home/apt	10400
16517	1189054292409642178		385187074	Bo ...	Sumida Ku	35.725495	139.829714	Entire home/apt	31314

16518 rows × 9 columns

In [ ]:
```python
# Creating directory path for export of cleaned data.
clean_data_dir = os.path.join("..", "..", "data", "clean")

cleaned_listings_export_path = os.path.abspath(
    os.path.join(script_dir, clean_data_dir, "cleaned_listings.csv")
)

cleaned_neighbourhoods_export_path = os.path.abspath(
    os.path.join(script_dir, clean_data_dir, "cleaned_neighbourhoods.csv")
)
```

In [ ]:
```python
cleaned_listings_export_path
```

Out[ ]:
```
'd:\\Admin Files\\Desktop\\Data Proj\\personal-proj\\tokyo-airbnb-proj\\data\\clean\\cleaned_listings.csv'
```

In [ ]:
```python
cleaned_neighbourhoods_export_path
```

Out[ ]:
```
'd:\\Admin Files\\Desktop\\Data Proj\\personal-proj\\tokyo-airbnb-proj\\data\\clean\\cleaned_neighbourhoods.csv'
```

In [ ]:
```python
# Exporting cleaned data to directory.
df_listings_cleaned.to_csv(cleaned_listings_export_path, index=False)
df_neighbourhoods_cleaned.to_csv(cleaned_neighbourhoods_export_path, index=False)
```