```
import pandas as pd

import numpy as np

import matplotlib.pylab as plt

import seaborn as sns

plt.style.use('ggplot')

pd.set_option('max_columns', 200)


# %% [code] {"execution":{"iopub.status.busy":"2021-12-
31T18:56:46.639134Z","iopub.execute_input":"2021-12-
31T18:56:46.639719Z","iopub.status.idle":"2021-12-
31T18:56:46.667382Z","shell.execute_reply.started":"2021-12-
31T18:56:46.639684Z","shell.execute_reply":"2021-12-31T18:56:46.666476Z"}}

df = pd.read_csv('../input/rollercoaster-database/coaster_db.csv')



# %% [code] {"execution":{"iopub.status.busy":"2021-12-
31T18:56:47.269667Z","iopub.execute_input":"2021-12-
31T18:56:47.270011Z","iopub.status.idle":"2021-12-
31T18:56:47.276513Z","shell.execute_reply.started":"2021-12-
31T18:56:47.269977Z","shell.execute_reply":"2021-12-31T18:56:47.275687Z"}}

df.shape


# %% [code] {"execution":{"iopub.status.busy":"2021-12-
31T18:56:47.613142Z","iopub.execute_input":"2021-12-
31T18:56:47.613539Z","iopub.status.idle":"2021-12-
31T18:56:47.658922Z","shell.execute_reply.started":"2021-12-
31T18:56:47.613509Z","shell.execute_reply":"2021-12-31T18:56:47.658309Z"}}

df.head(5)


# %% [code] {"execution":{"iopub.status.busy":"2021-12-
31T18:56:47.948319Z","iopub.execute_input":"2021-12-
```

31T18:56:47.948602Z","iopub.status.idle":"2021-12-
31T18:56:47.954587Z","shell.execute_reply.started":"2021-12-
31T18:56:47.948571Z","shell.execute_reply":"2021-12-31T18:56:47.953845Z"}}

```
df.columns
```

```
# %% [code] {"execution":{"iopub.status.busy":"2021-12-
31T18:56:48.282973Z","iopub.execute_input":"2021-12-
31T18:56:48.283386Z","iopub.status.idle":"2021-12-
31T18:56:48.290983Z","shell.execute_reply.started":"2021-12-
31T18:56:48.283347Z","shell.execute_reply":"2021-12-31T18:56:48.290188Z"}}
```

```
df.dtypes
```

```
# %% [code] {"execution":{"iopub.status.busy":"2021-12-
31T18:56:48.647734Z","iopub.execute_input":"2021-12-
31T18:56:48.648041Z","iopub.status.idle":"2021-12-
31T18:56:48.687364Z","shell.execute_reply.started":"2021-12-
31T18:56:48.648007Z","shell.execute_reply":"2021-12-31T18:56:48.686367Z"}}
```

```
df.describe()
```

```
df = df[['coaster_name',
   # 'Length', 'Speed',
   'Location', 'Status',
   # 'Opening date',
   #   'Type',
   'Manufacturer',
#    'Height restriction', 'Model', 'Height',
#       'Inversions', 'Lift/launch system', 'Cost', 'Trains', 'Park section',
#       'Duration', 'Capacity', 'G-force', 'Designer', 'Max vertical angle',
#       'Drop', 'Soft opening date', 'Fast Lane available', 'Replaced',
#       'Track layout', 'Fastrack available', 'Soft opening date.1',
```

```
#       'Closing date',
#    'Opened',
   # 'Replaced by', 'Website',
#       'Flash Pass Available', 'Must transfer from wheelchair', 'Theme',
#       'Single rider line available', 'Restraint Style',
#       'Flash Pass available', 'Acceleration', 'Restraints', 'Name',
     'year_introduced',
      'latitude', 'longitude',
   'Type_Main',
     'opening_date_clean',
   #'speed1', 'speed2', 'speed1_value', 'speed1_unit',
     'speed_mph',
   #'height_value', 'height_unit',
   'height_ft',
     'Inversions_clean', 'Gforce_clean']].copy()


# %% [code] {"execution":{"iopub.status.busy":"2021-12-
31T19:03:34.100419Z","iopub.execute_input":"2021-12-
31T19:03:34.10083Z","iopub.status.idle":"2021-12-
31T19:03:34.108123Z","shell.execute_reply.started":"2021-12-
31T19:03:34.100799Z","shell.execute_reply":"2021-12-31T19:03:34.107322Z"}}
df['opening_date_clean'] = pd.to_datetime(df['opening_date_clean'])


# %% [code] {"execution":{"iopub.status.busy":"2021-12-
31T19:07:00.935467Z","iopub.execute_input":"2021-12-
31T19:07:00.936143Z","iopub.status.idle":"2021-12-
31T19:07:00.943112Z","shell.execute_reply.started":"2021-12-
31T19:07:00.936091Z","shell.execute_reply":"2021-12-31T19:07:00.942155Z"}}
# Rename our columns
df = df.rename(columns={'coaster_name':'Coaster_Name',
```

```
            'year_introduced':'Year_Introduced',
            'opening_date_clean':'Opening_Date',
            'speed_mph':'Speed_mph',
            'height_ft':'Height_ft',
            'Inversions_clean':'Inversions',
            'Gforce_clean':'Gforce'})


# %% [code] {"execution":{"iopub.status.busy":"2021-12-
31T19:07:47.743843Z","iopub.execute_input":"2021-12-
31T19:07:47.744794Z","iopub.status.idle":"2021-12-
31T19:07:47.754112Z","shell.execute_reply.started":"2021-12-
31T19:07:47.744749Z","shell.execute_reply":"2021-12-31T19:07:47.753179Z"}}
df.isna().sum()


# %% [code] {"execution":{"iopub.status.busy":"2021-12-
31T19:09:09.229944Z","iopub.execute_input":"2021-12-
31T19:09:09.23084Z","iopub.status.idle":"2021-12-
31T19:09:09.2479Z","shell.execute_reply.started":"2021-12-
31T19:09:09.230776Z","shell.execute_reply":"2021-12-31T19:09:09.247222Z"}}
df.loc[df.duplicated()]


# %% [code] {"execution":{"iopub.status.busy":"2021-12-
31T19:10:13.801128Z","iopub.execute_input":"2021-12-
31T19:10:13.801563Z","iopub.status.idle":"2021-12-
31T19:10:13.824876Z","shell.execute_reply.started":"2021-12-
31T19:10:13.801525Z","shell.execute_reply":"2021-12-31T19:10:13.824228Z"}}
# Check for duplicate coaster name
df.loc[df.duplicated(subset=['Coaster_Name'])].head(5)


# %% [code] {"execution":{"iopub.status.busy":"2021-12-
31T19:11:30.91631Z","iopub.execute_input":"2021-12-
31T19:11:30.916627Z","iopub.status.idle":"2021-12-
```

31T19:11:30.934924Z","shell.execute_reply.started":"2021-12-31T19:11:30.916597Z","shell.execute_reply":"2021-12-31T19:11:30.934358Z"}}

Checking an example duplicate

df.query('Coaster_Name == "Crystal Beach Cyclone"')

%% [code] {"execution":{"iopub.status.busy":"2021-12-31T19:11:46.197705Z","iopub.execute_input":"2021-12-31T19:11:46.198033Z","iopub.status.idle":"2021-12-31T19:11:46.204172Z","shell.execute_reply.started":"2021-12-31T19:11:46.198003Z","shell.execute_reply":"2021-12-31T19:11:46.203259Z"}}

df.columns

%% [code] {"execution":{"iopub.status.busy":"2021-12-31T19:13:58.914303Z","iopub.execute_input":"2021-12-31T19:13:58.914698Z","iopub.status.idle":"2021-12-31T19:13:58.9217Z","shell.execute_reply.started":"2021-12-31T19:13:58.914669Z","shell.execute_reply":"2021-12-31T19:13:58.92102Z"}}

df = df.loc[~df.duplicated(subset=['Coaster_Name','Location','Opening_Date'])] \

  .reset_index(drop=True).copy()

%% [markdown]

Step 3: Feature Understanding

(Univariate analysis)

#

- Plotting Feature Distributions

- Histogram

- KDE

- Boxplot

%% [code] {"execution":{"iopub.status.busy":"2021-12-31T19:15:42.180128Z","iopub.execute_input":"2021-12-

31T19:15:42.180536Z","iopub.status.idle":"2021-12-
31T19:15:42.188209Z","shell.execute_reply.started":"2021-12-
31T19:15:42.180508Z","shell.execute_reply":"2021-12-31T19:15:42.187385Z"}}

```
df['Year_Introduced'].value_counts()
```

%% [code] {"execution":{"iopub.status.busy":"2021-12-
31T19:18:37.875007Z","iopub.execute_input":"2021-12-
31T19:18:37.875663Z","iopub.status.idle":"2021-12-
31T19:18:38.080836Z","shell.execute_reply.started":"2021-12-
31T19:18:37.875615Z","shell.execute_reply":"2021-12-31T19:18:38.07998Z"}}

```
ax = df['Year_Introduced'].value_counts() \
   .head(10) \
   .plot(kind='bar', title='Top 10 Years Coasters Introduced')
ax.set_xlabel('Year Introduced')
ax.set_ylabel('Count')
```

%% [code] {"execution":{"iopub.status.busy":"2021-12-
31T19:23:10.801878Z","iopub.execute_input":"2021-12-
31T19:23:10.802756Z","iopub.status.idle":"2021-12-
31T19:23:11.040812Z","shell.execute_reply.started":"2021-12-
31T19:23:10.802718Z","shell.execute_reply":"2021-12-31T19:23:11.039768Z"}}

```
ax = df['Speed_mph'].plot(kind='hist',
                          bins=20,
                          title='Coaster Speed (mph)')
ax.set_xlabel('Speed (mph)')
```

%% [code] {"execution":{"iopub.status.busy":"2021-12-
31T19:23:50.580456Z","iopub.execute_input":"2021-12-
31T19:23:50.580721Z","iopub.status.idle":"2021-12-
31T19:23:50.777702Z","shell.execute_reply.started":"2021-12-
31T19:23:50.580693Z","shell.execute_reply":"2021-12-31T19:23:50.776696Z"}}

```
ax = df['Speed_mph'].plot(kind='kde',
```

```
                title='Coaster Speed (mph)')
ax.set_xlabel('Speed (mph)')


# %% [code] {"execution":{"iopub.status.busy":"2021-12-
31T19:24:46.445487Z","iopub.execute_input":"2021-12-
31T19:24:46.445746Z","iopub.status.idle":"2021-12-
31T19:24:46.454475Z","shell.execute_reply.started":"2021-12-
31T19:24:46.445719Z","shell.execute_reply":"2021-12-31T19:24:46.453787Z"}}
df['Type_Main'].value_counts()


# %% [markdown]
# # Step 4: Feature Relationships
# - Scatterplot
# - Heatmap Correlation
# - Pairplot
# - Groupby comparisons


# %% [code] {"execution":{"iopub.status.busy":"2021-12-
31T19:28:27.768444Z","iopub.execute_input":"2021-12-
31T19:28:27.768762Z","iopub.status.idle":"2021-12-
31T19:28:27.997695Z","shell.execute_reply.started":"2021-12-
31T19:28:27.768721Z","shell.execute_reply":"2021-12-31T19:28:27.996745Z"}}
df.plot(kind='scatter',
    x='Speed_mph',
    y='Height_ft',
    title='Coaster Speed vs. Height')
plt.show()


# %% [code] {"execution":{"iopub.status.busy":"2021-12-
31T19:33:28.681335Z","iopub.execute_input":"2021-12-
31T19:33:28.681767Z","iopub.status.idle":"2021-12-
```

31T19:33:28.987093Z","shell.execute_reply.started":"2021-12-31T19:33:28.681729Z","shell.execute_reply":"2021-12-31T19:33:28.986081Z"}}

```
ax = sns.scatterplot(x='Speed_mph',
            y='Height_ft',
            hue='Year_Introduced',
            data=df)
ax.set_title('Coaster Speed vs. Height')
plt.show()
```

%% [code] {"execution":{"iopub.status.busy":"2021-12-31T19:44:18.59614Z","iopub.execute_input":"2021-12-31T19:44:18.596449Z","iopub.status.idle":"2021-12-31T19:44:26.264213Z","shell.execute_reply.started":"2021-12-31T19:44:18.59642Z","shell.execute_reply":"2021-12-31T19:44:26.263222Z"}}

```
sns.pairplot(df,
          vars=['Year_Introduced','Speed_mph',
              'Height_ft','Inversions','Gforce'],
         hue='Type_Main')
plt.show()
```

%% [code] {"execution":{"iopub.status.busy":"2021-12-31T19:46:48.88683Z","iopub.execute_input":"2021-12-31T19:46:48.887658Z","iopub.status.idle":"2021-12-31T19:46:48.905496Z","shell.execute_reply.started":"2021-12-31T19:46:48.887613Z","shell.execute_reply":"2021-12-31T19:46:48.904692Z"}}

```
df_corr = df[['Year_Introduced','Speed_mph',
   'Height_ft','Inversions','Gforce']].dropna().corr()
df_corr
```

%% [code] {"execution":{"iopub.status.busy":"2021-12-31T19:47:08.677068Z","iopub.execute_input":"2021-12-

31T19:47:08.677335Z","iopub.status.idle":"2021-12-31T19:47:08.996226Z","shell.execute_reply.started":"2021-12-31T19:47:08.677308Z","shell.execute_reply":"2021-12-31T19:47:08.995238Z"}}

```
sns.heatmap(df_corr, annot=True)
```

%% [markdown] {"execution":{"iopub.status.busy":"2021-12-31T20:08:11.029825Z","iopub.execute_input":"2021-12-31T20:08:11.030274Z","iopub.status.idle":"2021-12-31T20:08:11.035592Z","shell.execute_reply.started":"2021-12-31T20:08:11.030244Z","shell.execute_reply":"2021-12-31T20:08:11.03458Z"}}

```
# # Step 5: Ask a Question about the data

# - Try to answer a question you have about the data using a plot or statistic.

#

# What are the locations with the fastest roller coasters (minimum of 10)?
```

%% [code] {"execution":{"iopub.status.busy":"2021-12-31T20:12:48.582882Z","iopub.execute_input":"2021-12-31T20:12:48.583615Z","iopub.status.idle":"2021-12-31T20:12:48.841186Z","shell.execute_reply.started":"2021-12-31T20:12:48.583584Z","shell.execute_reply":"2021-12-31T20:12:48.840345Z"}}

```
ax = df.query('Location != "Other"') \
   .groupby('Location')['Speed_mph'] \
   .agg(['mean','count']) \
   .query('count >= 10') \
   .sort_values('mean')['mean'] \
   .plot(kind='barh', figsize=(12, 5), title='Average Coast Speed by Location')
ax.set_xlabel('Average Coaster Speed')
plt.show()
```