

You can use the following command to **merge the two DataFrames** above:

```
dataframe_1.merge(dataframe_2, on = ['column_1', 'column_2'], how = '___')
```

The attribute "**how**" in the code above specifies the type of merge that is to be performed. Merges are of the following different types:

**left:** This will select the entries only in the first dataframe.

**right:** This will consider the entries only in the second dataframe.

**outer:** This takes the union of all the entries in the dataframes.

**inner:** This will result in the intersection of the keys from both frames.

Depending on the situation, you can use an appropriate method to merge the two DataFrames.

## Dataframes Merge

Description - Perform an inner merge on two data frames df\_1 and df\_2 on 'unique\_id' and print the combined dataframe.

In [2]:

```
import pandas as pd
df_1 = pd.read_csv('https://cdn.upgrad.com/uploads/production/1ed2840b-f083-44fe-9eb4-acda009ac620/restaurant-1.csv')
df_2 = pd.read_csv('https://cdn.upgrad.com/uploads/production/c87b99b4-467c-4128-8729-85bff0b0acbe/restaurant-2.csv')
df_3 = df_1.merge(df_2,on='unique_id',how='inner')
print(df_3.head(20))
```

	name	address	city \
0	arnie morton's of chicago	"435 s. la cienega blvd."	"los angeles"
1	art's deli	"12224 ventura blvd."	"studio city"
2	bel-air hotel	"701 stone canyon rd."	"bel air"
3	cafe bizou	"14016 ventura blvd."	"sherman oaks"
4	campanile	"624 s. la brea ave."	"los angeles"
5	chinois on main	"2709 main st."	"santa monica"
6	citrus	"6703 melrose ave."	"los angeles"
7	fenix at the argyle	"8358 sunset blvd."	"w. hollywood"
8	granita	"23725 w. malibu rd."	"malibu"
9	grill the	"9560 dayton way"	"beverly hills"
10	katsu	"1972 hillhurst ave."	"los feliz"
11	l'orangerie	"903 n. la cienega blvd."	"w. hollywood"
12	le chardonnay (los angeles)	"8284 melrose ave."	"los angeles"
13	locanda veneta	"8638 w. third st."	"los angeles"
14	matsuhisa	"129 n. la cienega blvd."	"beverly hills"

15	palm the (los angeles)	"9001 santa monica blvd."	"w. hollywood"
16	patina	"5955 melrose ave."	"los angeles"
17	philippe the original	"1001 n. alameda st."	"chinatown"
18	pinot bistro	"12969 ventura blvd."	"studio city"
19	rex il ristorante	"617 s. olive st."	"los angeles"

	cuisine	unique_id	name_2 \
0	"steakhouses"	'0'	arnie morton's of chicago
1	"delis"	'1'	art's delicatessen
2	"californian"	'2'	hotel bel-air
3	"french bistro"	'3'	cafe bizou
4	"californian"	'4'	campanile
5	"pacific new wave"	'5'	chinois on main
6	"californian"	'6'	citrus
7	"french (new)"	'7'	fenix
8	"californian"	'8'	granita
9	"american (traditional)"	'9'	grill on the alley
10	"japanese"	'10'	restaurant katsu
11	"french (classic)"	'11'	l'orangerie
12	"french bistro"	'12'	le chardonnay
13	"italian"	'13'	locanda veneta
14	"seafood"	'14'	matsuhisa
15	"steakhouses"	'15'	the palm
16	"californian"	'16'	patina
17	"cafeterias"	'17'	philippe's the original
18	"french bistro"	'18'	pinot bistro
19	"nuova cucina italian"	'19'	rex il ristorante

	address_2	city_2	cuisine_2
0	"435 s. la cienega blv."	"los angeles"	"american"
1	"12224 ventura blvd."	"studio city"	"american"
2	"701 stone canyon rd."	"bel air"	"californian"
3	"14016 ventura blvd."	"sherman oaks"	"french"
4	"624 s. la brea ave."	"los angeles"	"american"
5	"2709 main st."	"santa monica"	"french"
6	"6703 melrose ave."	"los angeles"	"californian"
7	"8358 sunset blvd. west"	"hollywood"	"american"
8	"23725 w. malibu rd."	"malibu"	"californian"
9	"9560 dayton way"	"los angeles"	"american"
10	"1972 n. hillhurst ave."	"los angeles"	"asian"
11	"903 n. la cienega blvd."	"los angeles"	"french"
12	"8284 melrose ave."	"los angeles"	"french"
13	"3rd st."	"los angeles"	"italian"
14	"129 n. la cienega blvd."	"beverly hills"	"asian"

15	"9001 santa monica blvd."	"los angeles"	"american"
16	"5955 melrose ave."	"los angeles"	"californian"
17	"1001 n. alameda st."	"los angeles"	"american"
18	"12969 ventura blvd."	"los angeles"	"french"
19	"617 s. olive st."	"los angeles"	"italian"

---

## Dataframe Append

Description - Append two datasets df\_1 and df\_2, and print the combined dataframe.

In [4]:

```
# Suppressing warnings
import warnings
warnings.simplefilter("ignore")

import pandas as pd
df_1 = pd.read_csv('https://cdn.upgrad.com/uploads/production/1ed2840b-f083-44fe-9eb4-acda009ac620/restaurant-1.csv')
df_2 = pd.read_csv('https://cdn.upgrad.com/uploads/production/c87b99b4-467c-4128-8729-85bff0b0acbe/restaurant-2.csv')
df_3 = df_1.append(df_2)

print(df_3.head())
```

	name	address	city \
0	arnie morton's of chicago	"435 s. la cienega blvd."	"los angeles"
1	art's deli	"12224 ventura blvd."	"studio city"
2	bel-air hotel	"701 stone canyon rd."	"bel air"
3	cafe bizou	"14016 ventura blvd."	"sherman oaks"
4	campanile	"624 s. la brea ave."	"los angeles"

	cuisine	unique_id	name_2	address_2	city_2	cuisine_2
0	"steakhouses"	'0'	NaN	NaN	NaN	NaN
1	"delis"	'1'	NaN	NaN	NaN	NaN
2	"californian"	'2'	NaN	NaN	NaN	NaN
3	"french bistro"	'3'	NaN	NaN	NaN	NaN
4	"californian"	'4'	NaN	NaN	NaN	NaN

---

## Operations on multiple dataframes

Description -

Given three data frames containing the number of gold, silver, and bronze Olympic medals won by some countries, determine the total number of medals won by each country.

Note: All three data frames don't have all the same countries. So, ensure you use the 'fill\_value' argument (set it to zero), to avoid getting NaN values. Also, ensure you sort the final data frame, according to the total medal count in descending order. Make sure that the results are in integers.

In [5]:

```
import numpy as np
import pandas as pd

# Defining the three dataframes indicating the gold, silver, and bronze medal counts of different countries
gold = pd.DataFrame({'Country': ['USA', 'France', 'Russia'],
                     'Medals': [15, 13, 9]}
)
silver = pd.DataFrame({'Country': ['USA', 'Germany', 'Russia'],
                      'Medals': [29, 20, 16]}
)
bronze = pd.DataFrame({'Country': ['France', 'USA', 'UK'],
                       'Medals': [40, 28, 27]}
)
df_1 = pd.DataFrame(gold).append(pd.DataFrame(silver)).append(pd.DataFrame(bronze))
grp_df1 = df_1.groupby("Country").Medals.agg(sum).to_frame()
grp_df1.columns = ["Medals"]
grp_df1 = grp_df1.sort_values(by=["Medals"], ascending=False)
print(grp_df1)
```

	Medals
Country	
USA	72
France	53
UK	27
Russia	25
Germany	20

---

---