NAME: GALAPIA, XANDER SAM E.

SECTION: CPE22S3

Add text cell

```
import pandas as pd

weather = pd.read_csv('data/nyc_weather_2018.csv')
weather.head()
```

	attributes	datatype	date	station	value
0	,,N,	PRCP	2018-01-01T00:00:00	GHCND:US1CTFR0039	0.0
1	,,N,	PRCP	2018-01-01T00:00:00	GHCND:US1NJBG0015	0.0
2	,,N,	SNOW	2018-01-01T00:00:00	GHCND:US1NJBG0015	0.0
3	,,N,	PRCP	2018-01-01T00:00:00	GHCND:US1NJBG0017	0.0
4	,,N,	SNOW	2018-01-01T00:00:00	GHCND:US1NJBG0017	0.0

```
snow_data = weather.query('datatype == "SNOW" and value > 0')
snow_data.head()
```

	attributes	datatype	date	station	value
124	,,N,	SNOW	2018-01-01T00:00:00	GHCND:US1NYWC0019	25.0
723	,,N,	SNOW	2018-01-04T00:00:00	GHCND:US1NJBG0015	229.0
726	,,N,	SNOW	2018-01-04T00:00:00	GHCND:US1NJBG0017	10.0
730	,,N,	SNOW	2018-01-04T00:00:00	GHCND:US1NJBG0018	46.0
737	,,N,	SNOW	2018-01-04T00:00:00	GHCND:US1NJES0018	10.0

```
import sqlite3
with sqlite3.connect('/content/data/weather.db') as connection:
  snow_data_from_db = pd.read_sql('SELECT * FROM weather WHERE datatype == "SNOW" AND value > 0' ,connection)


snow_data.reset_index().drop(columns='index').equals(snow_data_from_db)
```

```
True
```

```
weather[(weather.datatype == 'SNOW') & (weather.value > 0)].equals(snow_data)
```

```
True
```

```
station_info = pd.read_csv('data/weather_stations.csv')
station_info.head()
```

	id		de	longitude	elevation
0	GHCND:US1CTFR0022	STAMFORD 2.6 SSW, CT US	41.0641	-73.5770	36.6
1	GHCND:US1CTFR0039	STAMFORD 4.2 S, CT US	41.0378	-73.5682	6.4
2	GHCND:US1NJBG0001	BERGENFIELD 0.3 SW, NJ US	40.9213	-74.0020	20.1
3	GHCND:US1NJBG0002	SADDLE BROOK TWP 0.6 E, NJ US	40.9027	-74.0834	16.8
4	GHCND:US1NJBG0003	TENAFLY 1.3 W, NJ US	40.9147	-73.9775	21.6

Add text cell

```
weather.head()
```

	attributes	datatype	date	station	value
0	„N,	PRCP	2018-01-01T00:00:00	GHCND:US1CTFR0039	0.0
1	„N,	PRCP	2018-01-01T00:00:00	GHCND:US1NJBG0015	0.0
2	„N,	SNOW	2018-01-01T00:00:00	GHCND:US1NJBG0015	0.0
3	„N,	PRCP	2018-01-01T00:00:00	GHCND:US1NJBG0017	0.0
4	„N,	SNOW	2018-01-01T00:00:00	GHCND:US1NJBG0017	0.0

```
station_info.id.describe()
```

```
count                   262
unique                  262
top       GHCND:US1CTFR0022
freq                      1
Name: id, dtype: object
```

```
weather.station.describe()
```

```
count                 80256
unique                  109
top       GHCND:USW00094789
freq                   4270
Name: station, dtype: object
```

```
station_info.shape[0], weather.shape[0]
```

```
(262, 80256)
```

```
def get_row_count(*dfs):
  return [df.shape[0] for df in dfs]
get_row_count(station_info, weather)
```

```
[262, 80256]
```

Add text cell

```
def get_info(attr, *dfs):
  return list(map(lambda x: getattr(x, attr), dfs))
get_info('shape', station_info, weather)
```

```
[(262, 5), (80256, 5)]
```

```
inner_join = weather.merge(station_info, left_on ='station', right_on='id')
inner_join.sample(5, random_state=0)
```

	attributes	datatype	date	station	value	id	name	latitude	longitude	elevation
27422	„N,	PRCP	2018-01-23T00:00:00	GHCND:US1NYSF0061	2.3	GHCND:US1NYSF0061	CENTERPORT 0.9 SW, NY US	40.8917	-73.3831	53.6
19317	T„N,	PRCP	2018-08-10T00:00:00	GHCND:US1NJUN0014	0.0	GHCND:US1NJUN0014	WESTFIELD 0.6 NE, NJ US	40.6588	-74.3358	36.3
13778	„N,	WESF	2018-02-18T00:00:00	GHCND:US1NJMS0089	19.6	GHCND:US1NJMS0089	PARSIPPANY TROY HILLS TWP 1.3, NJ US	40.8716	-74.4055	103.6
39633	„7,0700	PRCP	2018-04-06T00:00:00	GHCND:USC00301309	0.0	GHCND:USC00301309	CENTERPORT, NY US	40.8838	-73.3722	9.1
51025	„W,2400	SNWD	2018-12-14T00:00:00	GHCND:USW00014734	0.0	GHCND:USW00014734	NEWARK LIBERTY INTERNATIONAL AIRPORT, NJ US	40.6825	-74.1694	2.1

```
weather.merge(station_info.rename(dict(id='station'), axis = 1), on='station').sample(5, random_state=0)
```

	attributes	datatype	date	station	value	name	latitude	longitude	elevation
27422	„N,	PRCP	2018-01-23T00:00:00	GHCND:US1NYSF0061	2.3	CENTERPORT 0.9 SW, NY US	40.8917	-73.3831	53.6
19317	T„N,	PRCP	2018-08-10T00:00:00	GHCND:US1NJUN0014	0.0	WESTFIELD 0.6 NE, NJ US	40.6588	-74.3358	36.3
13778	„N,	WESF	2018-02-18T00:00:00	GHCND:US1NJMS0089	19.6	PARSIPPANY TROY HILLS TWP 1.3, NJ US	40.8716	-74.4055	103.6
39633	„7,0700	PRCP	2018-04-06T00:00:00	GHCND:USC00301309	0.0	CENTERPORT, NY US	40.8838	-73.3722	9.1
51025	„W,2400	SNWD	2018-12-14T00:00:00	GHCND:USW00014734	0.0	NEWARK LIBERTY INTERNATIONAL AIRPORT, NJ US	40.6825	-74.1694	2.1

```
left_join = station_info.merge(weather, left_on='id', right_on='station', how ='left')
right_join = weather.merge(station_info, left_on = 'station', right_on = 'id', how= 'right')

right_join.tail()
```

	attributes	datatype	date	station	value	id	name	latitude	longitude	elevation
80404	„W,	WDF5	2018-12-31T00:00:00	GHCND:USW00094789	130.0	GHCND:USW00094789	JFK INTERNATIONAL AIRPORT, NY US	40.6386	-73.7622	3.4
80405	„W,	WSF2	2018-12-31T00:00:00	GHCND:USW00094789	9.8	GHCND:USW00094789	JFK INTERNATIONAL AIRPORT, NY US	40.6386	-73.7622	3.4
80406	„W,	WSF5	2018-12-31T00:00:00	GHCND:USW00094789	12.5	GHCND:USW00094789	JFK INTERNATIONAL AIRPORT, NY US	40.6386	-73.7622	3.4
80407	„W,	WT01	2018-12-31T00:00:00	GHCND:USW00094789	1.0	GHCND:USW00094789	JFK INTERNATIONAL AIRPORT, NY US	40.6386	-73.7622	3.4
80408	„W,	WT02	2018-12-31T00:00:00	GHCND:USW00094789	1.0	GHCND:USW00094789	JFK INTERNATIONAL AIRPORT, NY US	40.6386	-73.7622	3.4

Add text cell

```
left_join.sort_index(axis=1).sort_values(['date', 'station']).reset_index().drop(columns = 'index').equals(
    right_join.sort_index(axis=1).sort_values(['date', 'station']).reset_index().drop(columns ='index')
)
```

```
True
```

```
get_info('shape', inner_join, left_join, right_join)
```

```
[(80256, 10), (80409, 10), (80409, 10)]
```

```
outer_join = weather.merge(
    station_info[station_info.name.str.contains('NY')],
    left_on = 'station', right_on = 'id', how ='outer' , indicator = True
)

outer_join.sample(4, random_state=0).append(outer_join[outer_join.station.isna()].head(2))
```

```
<ipython-input-18-78c2db34de9a>:6: FutureWarning: The frame.append method is deprecated and will be removed from pandas in a future version. Use pandas.concat instead.
  outer_join.sample(4, random_state=0).append(outer_join[outer_join.station.isna()].head(2))
```

	attributes	datatype	date	station	value	id	name	latitude	longitude	elevation	_merge
17259	„N,	PRCP	2018-05-15T00:00:00	GHCND:US1NJPS0022	0.3	NaN	NaN	NaN	NaN	NaN	left_only
76178	„N,	PRCP	2018-05-19T00:00:00	GHCND:US1NJPS0015	8.1	NaN	NaN	NaN	NaN	NaN	left_only
73410	„N,	MDPR	2018-08-05T00:00:00	GHCND:US1NYNS0018	12.2	GHCND:US1NYNS0018	HICKSVILLE 1.3 ENE, NY US	40.7687	-73.5017	45.7	both
74822	„N,	SNOW	2018-04-02T00:00:00	GHCND:US1NJMS0016	178.0	NaN	NaN	NaN	NaN	NaN	left_only
80256	NaN	NaN	NaN	NaN	NaN	GHCND:US1NJMS0036	PARSIPPANY TROY HILLS TWP 2.1, NJ US	40.8656	-74.3851	64.3	right_only
80257	NaN	NaN	NaN	NaN	NaN	GHCND:US1NJMS0039	PARSIPPANY TROY HILLS TWP 1.3, NJ US	40.8533	-74.4470	94.2	right_only

Add text cell

```
import sqlite3

with sqlite3.connect('/content/data/weather.db') as connection:
  inner_join_from_db = pd.read_sql(
      'SELECT * FROM weather JOIN stations ON weather.station == stations.id',
      connection
      )
inner_join_from_db.shape == inner_join.shape
```

```
True
```

```
dirty_data = pd.read_csv(
    'data/dirty_data.csv' , index_col = 'date'
).drop_duplicates().drop(columns='SNWD')
dirty_data.head()
```

```
---------------------------------------------------------------------------
FileNotFoundError                         Traceback (most recent call last)
<ipython-input-20-f4e53e2576b5> in <cell line: 1>()
----> 1 dirty_data = pd.read_csv(
      2     'data/dirty_data.csv' , index_col = 'date'
      3 ).drop_duplicates().drop(columns='SNWD')
      4 dirty_data.head()
```

Add text cell

6 frames

```
/usr/local/lib/python3.10/dist-packages/pandas/io/common.py in get_handle(path_or_buf, mode, encoding, compression, memory_map, is_text, errors, storage_options)
    854         if ioargs.encoding and "b" not in ioargs.mode:
    855             # Encoding
--> 856             handle = open(
    857                 handle,
    858                 ioargs.mode,

FileNotFoundError: [Errno 2] No such file or directory: 'data/dirty_data.csv'
```

```
valid_station = dirty_data.query('station != "?"').copy().drop(columns=['WESF', 'station'])
station_with_weaf = dirty_data.query('station == "?"').copy().drop(columns = ['station', 'TOBS', 'TMIN', 'TMAX'])
```

```
valid_station.merge(
    station_with_weaf, left_index=True, right_index=True
).query('WESF > 0').head()
```

date	PRCP_x	SNOW_x	TMAX	TMIN	TOBS	inclement_weather_x	PRCP_y	SNOW_y	WESF	inclement_weather_y
2018-01-30T00:00:00	0.0	0.0	6.7	-1.7	-0.6	False	1.5	13.0	1.8	True
2018-03-08T00:00:00	48.8	NaN	1.1	-0.6	1.1	False	28.4	NaN	28.7	NaN
2018-03-13T00:00:00	4.1	51.0	5.6	-3.9	0.0	True	3.0	13.0	3.0	True
2018-03-21T00:00:00	0.0	0.0	2.8	-2.8	0.6	False	6.6	114.0	8.6	True
2018-04-02T00:00:00	9.1	127.0	12.8	-1.1	-1.1	True	14.0	152.0	15.2	True

```
valid_station.merge(
    station_with_weaf, left_index=True, right_index = True, suffixes =('', '_?')
).query('WESF > 0').head()
```

date	PRCP	SNOW	TMAX	TMIN	TOBS	inclement_weather	PRCP_?	SNOW_?	WESF	inclement_weather_?
2018-01-30T00:00:00	0.0	0.0	6.7	-1.7	-0.6	False	1.5	13.0	1.8	True
2018-03-08T00:00:00	48.8	NaN	1.1	-0.6	1.1	False	28.4	NaN	28.7	NaN
2018-03-13T00:00:00	4.1	51.0	5.6	-3.9	0.0	True	3.0	13.0	3.0	True
2018-03-21T00:00:00	0.0	0.0	2.8	-2.8	0.6	False	6.6	114.0	8.6	True
2018-04-02T00:00:00	9.1	127.0	12.8	-1.1	-1.1	True	14.0	152.0	15.2	True

Add text cell

```
valid_station.join(station_with_weaf, rsuffix='_?').query('WESF > 0').head()
```

date	PRCP	SNOW	TMAX	TMIN	TOBS	inclement_weather	PRCP_?	SNOW_?	WESF	inclement_weather_?
2018-01-30T00:00:00	0.0	0.0	6.7	-1.7	-0.6	False	1.5	13.0	1.8	True
2018-03-08T00:00:00	48.8	NaN	1.1	-0.6	1.1	False	28.4	NaN	28.7	NaN
2018-03-13T00:00:00	4.1	51.0	5.6	-3.9	0.0	True	3.0	13.0	3.0	True
2018-03-21T00:00:00	0.0	0.0	2.8	-2.8	0.6	False	6.6	114.0	8.6	True
2018-04-02T00:00:00	9.1	127.0	12.8	-1.1	-1.1	True	14.0	152.0	15.2	True

```
weather.set_index('station', inplace= True)
station_info.set_index('id', inplace= True)
```

```
weather.index.intersection(station_info.index)
```

```
Index(['GHCND:US1CTFR0039', 'GHCND:US1NJBG0015', 'GHCND:US1NJBG0017',
       'GHCND:US1NJBG0018', 'GHCND:US1NJBG0023', 'GHCND:US1NJBG0030',
       'GHCND:US1NJBG0039', 'GHCND:US1NJBG0044', 'GHCND:US1NJES0018',
       'GHCND:US1NJES0024',
       ...
       'GHCND:US1NJMS0047', 'GHCND:US1NYSF0083', 'GHCND:US1NYNY0074',
       'GHCND:US1NJPS0018', 'GHCND:US1NJBG0037', 'GHCND:USC00284987',
```

```
           'GHCND:US1NJES0031', 'GHCND:US1NJMD0086', 'GHCND:US1NJMS0097',
           'GHCND:US1NJMN0081'],
          dtype='object', length=109)
```

```
weather.index.difference(station_info.index)
```

```
Index([], dtype='object')
```

Add text cell

```
station_info.index.difference(weather.index)
```

```
Index(['GHCND:US1CTFR0022', 'GHCND:US1NJBG0001', 'GHCND:US1NJBG0002',
       'GHCND:US1NJBG0005', 'GHCND:US1NJBG0006', 'GHCND:US1NJBG0008',
       'GHCND:US1NJBG0011', 'GHCND:US1NJBG0012', 'GHCND:US1NJBG0013',
       'GHCND:US1NJBG0020',
       ...
       'GHCND:USC00308322', 'GHCND:USC00308749', 'GHCND:USC00308946',
       'GHCND:USC00309117', 'GHCND:USC00309270', 'GHCND:USC00309400',
       'GHCND:USC00309466', 'GHCND:USC00309576', 'GHCND:USW00014708',
       'GHCND:USW00014786'],
      dtype='object', length=153)
```

```
ny_in_name = station_info[station_info.name.str.contains('NY')]

ny_in_name.index.difference(weather.index).shape[0]\
+ weather.index.difference(ny_in_name.index).shape[0]\
== weather.index.symmetric_difference(ny_in_name.index).shape[0]
```

```
True
```

```
weather.index.unique().union(station_info.index)
```

```
Index(['GHCND:US1CTFR0022', 'GHCND:US1CTFR0039', 'GHCND:US1NJBG0001',
       'GHCND:US1NJBG0002', 'GHCND:US1NJBG0003', 'GHCND:US1NJBG0005',
       'GHCND:US1NJBG0006', 'GHCND:US1NJBG0008', 'GHCND:US1NJBG0010',
       'GHCND:US1NJBG0011',
       ...
       'GHCND:USW00014708', 'GHCND:USW00014732', 'GHCND:USW00014734',
       'GHCND:USW00014786', 'GHCND:USW00054743', 'GHCND:USW00054787',
       'GHCND:USW00094728', 'GHCND:USW00094741', 'GHCND:USW00094745',
       'GHCND:USW00094789'],
      dtype='object', length=262)
```

```
ny_in_name = station_info[station_info.name.str.contains('NY')]

ny_in_name.index.difference(weather.index).union(weather.index.difference(ny_in_name.index)).equals(
    weather.index.symmetric_difference(ny_in_name.index)
)
```