Week 2

May 13, 2020

---

*You are currently looking at **version 1.0** of this notebook. To download notebooks and datafiles, as well as get help on Jupyter notebooks in the Coursera platform, visit the Jupyter Notebook FAQ course resource.*

---

1 The Series Data Structure

```
In [1]: import pandas as pd
        pd.Series?

In [2]: animals = ['Tiger', 'Bear', 'Moose']
        pd.Series(animals)

Out[2]: 0    Tiger
        1     Bear
        2    Moose
        dtype: object

In [3]: numbers = [1, 2, 3]
        pd.Series(numbers)

Out[3]: 0    1
        1    2
        2    3
        dtype: int64

In [4]: animals = ['Tiger', 'Bear', None]
        pd.Series(animals)

Out[4]: 0    Tiger
        1     Bear
        2     None
        dtype: object

In [5]: numbers = [1, 2, None]
        pd.Series(numbers)
```

```
Out[5]: 0    1.0
        1    2.0
        2    NaN
        dtype: float64
```

```
In [6]: import numpy as np
        np.nan == None
```

```
Out[6]: False
```

```
In [7]: np.nan == np.nan
```

```
Out[7]: False
```

```
In [8]: np.isnan(np.nan)
```

```
Out[8]: True
```

```
In [9]: sports = {'Archery': 'Bhutan',
                  'Golf': 'Scotland',
                  'Sumo': 'Japan',
                  'Taekwondo': 'South Korea'}
        s = pd.Series(sports)
        s
```

```
Out[9]: Archery           Bhutan
        Golf            Scotland
        Sumo               Japan
        Taekwondo    South Korea
        dtype: object
```

```
In [10]: s.index
```

```
Out[10]: Index(['Archery', 'Golf', 'Sumo', 'Taekwondo'], dtype='object')
```

```
In [13]: s = pd.Series(['Tiger', 'Bear', 'Moose'], index=['India', 'America', 'Canada'])
         s
```

```
Out[13]: India      Tiger
         America     Bear
         Canada     Moose
         dtype: object
```

```
In [14]: sports = {'Archery': 'Bhutan',
                   'Golf': 'Scotland',
                   'Sumo': 'Japan',
                   'Taekwondo': 'South Korea'}
         s = pd.Series(sports, index=['Golf', 'Sumo', 'Hockey'])
         s
```

```
Out[14]: Golf      Scotland
         Sumo         Japan
         Hockey         NaN
         dtype: object
```

2 Querying a Series

```
In [15]: sports = {'Archery': 'Bhutan',
                   'Golf': 'Scotland',
                   'Sumo': 'Japan',
                   'Taekwondo': 'South Korea'}
         s = pd.Series(sports)
         s

Out[15]: Archery           Bhutan
         Golf            Scotland
         Sumo               Japan
         Taekwondo    South Korea
         dtype: object

In [16]: s.iloc[3]

Out[16]: 'South Korea'

In [17]: s.loc['Golf']

Out[17]: 'Scotland'

In [18]: s[3]

Out[18]: 'South Korea'

In [19]: s['Golf']

Out[19]: 'Scotland'

In [24]: sports = {99: 'Bhutan',
                   100: 'Scotland',
                   101: 'Japan',
                   102: 'South Korea'}
         s = pd.Series(sports)

In [21]: s[0] #This won't call s.iloc[0] as one might expect, it generates an error instead


         ---------------------------------------------------------------------------

         KeyError                                  Traceback (most recent call last)

         <ipython-input-21-a5f43d492595> in <module>()
    ----> 1 s[0] #This won't call s.iloc[0] as one might expect, it generates an error instead


         /opt/conda/lib/python3.6/site-packages/pandas/core/series.py in __getitem__(self, key)
         601         key = com._apply_if_callable(key, self)
```

```
    602         try:
--> 603             result = self.index.get_value(self, key)
    604
    605             if not is_scalar(result):


/opt/conda/lib/python3.6/site-packages/pandas/indexes/base.py in get_value(self, series,
   2167         try:
   2168             return self._engine.get_value(s, k,
-> 2169                                           tz=getattr(series.dtype, 'tz', None))
   2170         except KeyError as e1:
   2171             if len(self) > 0 and self.inferred_type in ['integer', 'boolean']:


pandas/index.pyx in pandas.index.IndexEngine.get_value (pandas/index.c:3557)()


pandas/index.pyx in pandas.index.IndexEngine.get_value (pandas/index.c:3240)()


pandas/index.pyx in pandas.index.IndexEngine.get_loc (pandas/index.c:4279)()


pandas/src/hashtable_class_helper.pxi in pandas.hashtable.Int64HashTable.get_item (panda


pandas/src/hashtable_class_helper.pxi in pandas.hashtable.Int64HashTable.get_item (panda


KeyError: 0
```

```
In [23]: s = pd.Series([100.00, 120.00, 101.00, 3.00])
         s

Out[23]: 0    100.0
         1    120.0
         2    101.0
         3      3.0
         dtype: float64

In [27]: total = 0
         for item in s:
             total+=item
         print(str(total))


         ---------------------------------------------------------------------------
```

```
TypeError                                 Traceback (most recent call last)

<ipython-input-27-fbfd83f3ab8d> in <module>()
  1 total = 0
  2 for item in s:
----> 3     total+=item
  4 print(str(total))


TypeError: unsupported operand type(s) for +=: 'int' and 'str'
```

In [28]:
```
import numpy as np

total = np.sum(s)
print(total)
```

```
BhutanScotlandJapanSouth Korea
```

In [32]:
```
#this creates a big series of random numbers
s = pd.Series(np.random.randint(0,10,1000))
s.head()
```

Out[32]:
```
0    7
1    3
2    8
3    0
4    4
dtype: int64
```

In [33]:
```
len(s)
```

Out[33]:
```
1000
```

In [34]:
```
%%timeit -n 100
summary = 0
for item in s:
    summary+=item
```

```
The slowest run took 7.00 times longer than the fastest. This could mean that an intermediate re
213 ţs ś 189 ţs per loop (mean ś std. dev. of 7 runs, 100 loops each)
```

In [35]:
```
%%timeit -n 100
summary = np.sum(s)
```

```
The slowest run took 9.30 times longer than the fastest. This could mean that an intermediate re
171 ţs ś 193 ţs per loop (mean ś std. dev. of 7 runs, 100 loops each)
```

```
In [36]: s+=2 #adds two to each item in s using broadcasting
         s.head()

Out[36]: 0     9
         1     5
         2    10
         3     2
         4     6
         dtype: int64

In [37]: for label, value in s.iteritems():
             s.set_value(label, value+2)
         s.head()

Out[37]: 0    11
         1     7
         2    12
         3     4
         4     8
         dtype: int64

In [1]: %%timeit -n 10
        s = pd.Series(np.random.randint(0,1000,10000))
        for label, value in s.iteritems():
            s.loc[label]= value+2


        ---------------------------------------------------------------------------

        NameError                                 Traceback (most recent call last)

        <ipython-input-1-28c01e28f9f7> in <module>()
    ----> 1 get_ipython().run_cell_magic('timeit', '-n 10', 's = pd.Series(np.random.randint(0,1


        /opt/conda/lib/python3.6/site-packages/IPython/core/interactiveshell.py in run_cell_magi
       2101             magic_arg_s = self.var_expand(line, stack_depth)
       2102             with self.builtin_trap:
    -> 2103                 result = fn(magic_arg_s, cell)
       2104             return result
       2105


        <decorator-gen-61> in timeit(self, line, cell)


        /opt/conda/lib/python3.6/site-packages/IPython/core/magic.py in <lambda>(f, *a, **k)
        185     # but it's overkill for just that one bit of state.
        186     def magic_deco(arg):
```

```
--> 187         call = lambda f, *a, **k: f(*a, **k)
    188
    189         if callable(arg):


    /opt/conda/lib/python3.6/site-packages/IPython/core/magics/execution.py in timeit(self,
   1082                     break
   1083
-> 1084         all_runs = timer.repeat(repeat, number)
   1085         best = min(all_runs) / number
   1086         worst = max(all_runs) / number


    /opt/conda/lib/python3.6/timeit.py in repeat(self, repeat, number)
    204         r = []
    205         for i in range(repeat):
--> 206             t = self.timeit(number)
    207             r.append(t)
    208         return r


    /opt/conda/lib/python3.6/site-packages/IPython/core/magics/execution.py in timeit(self,
    158         gc.disable()
    159         try:
--> 160             timing = self.inner(it, self.timer)
    161         finally:
    162             if gcold:


    <magic-timeit> in inner(_it, _timer)


    NameError: name 'pd' is not defined
```

```
In [2]: %%timeit -n 10
        s = pd.Series(np.random.randint(0,1000,10000))
        s+=2


        ---------------------------------------------------------------------------

        NameError                                 Traceback (most recent call last)

        <ipython-input-2-eaf0097ef77c> in <module>()
    ----> 1 get_ipython().run_cell_magic('timeit', '-n 10', 's = pd.Series(np.random.randint(0,1
```

```
    /opt/conda/lib/python3.6/site-packages/IPython/core/interactiveshell.py in run_cell_magi
   2101             magic_arg_s = self.var_expand(line, stack_depth)
   2102             with self.builtin_trap:
-> 2103                 result = fn(magic_arg_s, cell)
   2104             return result
   2105


    <decorator-gen-61> in timeit(self, line, cell)


    /opt/conda/lib/python3.6/site-packages/IPython/core/magic.py in <lambda>(f, *a, **k)
    185     # but it's overkill for just that one bit of state.
    186     def magic_deco(arg):
--> 187         call = lambda f, *a, **k: f(*a, **k)
    188
    189         if callable(arg):


    /opt/conda/lib/python3.6/site-packages/IPython/core/magics/execution.py in timeit(self,
   1082                     break
   1083
-> 1084         all_runs = timer.repeat(repeat, number)
   1085         best = min(all_runs) / number
   1086         worst = max(all_runs) / number


    /opt/conda/lib/python3.6/timeit.py in repeat(self, repeat, number)
    204         r = []
    205         for i in range(repeat):
--> 206             t = self.timeit(number)
    207             r.append(t)
    208         return r


    /opt/conda/lib/python3.6/site-packages/IPython/core/magics/execution.py in timeit(self,
    158         gc.disable()
    159         try:
--> 160             timing = self.inner(it, self.timer)
    161         finally:
    162             if gcold:


    <magic-timeit> in inner(_it, _timer)


    NameError: name 'pd' is not defined
```

```
In [3]: s = pd.Series([1, 2, 3])
        s.loc['Animal'] = 'Bears'
        s
```

```
        ---------------------------------------------------------------------------
        NameError                                 Traceback (most recent call last)

        <ipython-input-3-a9ee31708421> in <module>()
        ----> 1 s = pd.Series([1, 2, 3])
              2 s.loc['Animal'] = 'Bears'
              3 s


        NameError: name 'pd' is not defined
```

```
In [4]: original_sports = pd.Series({'Archery': 'Bhutan',
                                     'Golf': 'Scotland',
                                     'Sumo': 'Japan',
                                     'Taekwondo': 'South Korea'})
        cricket_loving_countries = pd.Series(['Australia',
                                              'Barbados',
                                              'Pakistan',
                                              'England'],
                                             index=['Cricket',
                                                    'Cricket',
                                                    'Cricket',
                                                    'Cricket'])
        all_countries = original_sports.append(cricket_loving_countries)
```

```
        ---------------------------------------------------------------------------
        NameError                                 Traceback (most recent call last)

        <ipython-input-4-bdb4ab47443f> in <module>()
        ----> 1 original_sports = pd.Series({'Archery': 'Bhutan',
              2                              'Golf': 'Scotland',
              3                              'Sumo': 'Japan',
              4                              'Taekwondo': 'South Korea'})
              5 cricket_loving_countries = pd.Series(['Australia',


        NameError: name 'pd' is not defined
```

```
In [5]: original_sports
```

```
        ---------------------------------------------------------------------------

        NameError                                 Traceback (most recent call last)

        <ipython-input-5-6057be8bfdd4> in <module>()
    ----> 1 original_sports


        NameError: name 'original_sports' is not defined


In [6]: cricket_loving_countries


        ---------------------------------------------------------------------------

        NameError                                 Traceback (most recent call last)

        <ipython-input-6-72bda9ef54df> in <module>()
    ----> 1 cricket_loving_countries


        NameError: name 'cricket_loving_countries' is not defined


In [7]: all_countries


        ---------------------------------------------------------------------------

        NameError                                 Traceback (most recent call last)

        <ipython-input-7-8692c1c3d2e5> in <module>()
    ----> 1 all_countries


        NameError: name 'all_countries' is not defined


In [8]: all_countries.loc['Cricket']


        ---------------------------------------------------------------------------

        NameError                                 Traceback (most recent call last)

        <ipython-input-8-87582a65f040> in <module>()
    ----> 1 all_countries.loc['Cricket']
```

```
NameError: name 'all_countries' is not defined
```

3 The DataFrame Data Structure

```
In [11]: import pandas as pd
         purchase_1 = pd.Series({'Name': 'Chris',
                                 'Item Purchased': 'Dog Food',
                                 'Cost': 22.50})
         purchase_2 = pd.Series({'Name': 'Kevyn',
                                 'Item Purchased': 'Kitty Litter',
                                 'Cost': 2.50})
         purchase_3 = pd.Series({'Name': 'Vinod',
                                 'Item Purchased': 'Bird Seed',
                                 'Cost': 5.00})
         df = pd.DataFrame([purchase_1, purchase_2, purchase_3], index=['Store 1', 'Store 2', 'S
         df.head()
```

```
Out[11]:          Cost Item Purchased   Name
         Store 1  22.5       Dog Food  Chris
         Store 2   2.5   Kitty Litter  Kevyn
         Store 3   5.0      Bird Seed  Vinod
```

```
In [12]: df.loc['Store 2']
```

```
Out[12]: Cost                         2.5
         Item Purchased    Kitty Litter
         Name                      Kevyn
         Name: Store 2, dtype: object
```

```
In [13]: type(df.loc['Store 2'])
```

```
Out[13]: pandas.core.series.Series
```

```
In [14]: df.loc['Store 1']
```

```
Out[14]: Cost                   22.5
         Item Purchased    Dog Food
         Name                 Chris
         Name: Store 1, dtype: object
```

```
In [15]: df.loc['Store 1', 'Cost']
```

```
Out[15]: 22.5
```

```
In [16]: df.T
```

```
Out[16]:                   Store 1       Store 2    Store 3
         Cost                 22.5           2.5          5
         Item Purchased  Dog Food  Kitty Litter  Bird Seed
         Name               Chris         Kevyn      Vinod

In [17]: df.T.loc['Cost']

Out[17]: Store 1    22.5
         Store 2     2.5
         Store 3       5
         Name: Cost, dtype: object

In [18]: df['Cost']

Out[18]: Store 1    22.5
         Store 2     2.5
         Store 3     5.0
         Name: Cost, dtype: float64

In [19]: df.loc['Store 1']['Cost']

Out[19]: 22.5

In [20]: df.loc[:,['Name', 'Cost']]

Out[20]:           Name  Cost
         Store 1  Chris  22.5
         Store 2  Kevyn   2.5
         Store 3  Vinod   5.0

In [21]: df.drop('Store 1')

Out[21]:          Cost Item Purchased   Name
         Store 2   2.5   Kitty Litter  Kevyn
         Store 3   5.0      Bird Seed  Vinod

In [22]: df

Out[22]:          Cost Item Purchased   Name
         Store 1  22.5       Dog Food  Chris
         Store 2   2.5   Kitty Litter  Kevyn
         Store 3   5.0      Bird Seed  Vinod

In [23]: copy_df = df.copy()
         copy_df = copy_df.drop('Store 1')
         copy_df

Out[23]:          Cost Item Purchased   Name
         Store 2   2.5   Kitty Litter  Kevyn
         Store 3   5.0      Bird Seed  Vinod
```

```
In [24]: copy_df.drop?

In [25]: del copy_df['Name']
         copy_df
```

Out[25]:

	Cost	Item Purchased
Store 2	2.5	Kitty Litter
Store 3	5.0	Bird Seed

```
In [26]: df['Location'] = None
         df
```

Out[26]:

	Cost	Item Purchased	Name	Location
Store 1	22.5	Dog Food	Chris	None
Store 2	2.5	Kitty Litter	Kevyn	None
Store 3	5.0	Bird Seed	Vinod	None

4 Dataframe Indexing and Loading

```
In [27]: costs = df['Cost']
         costs
```

```
Out[27]: Store 1    22.5
         Store 2     2.5
         Store 3     5.0
         Name: Cost, dtype: float64

In [28]: costs+=2
         costs

Out[28]: Store 1    24.5
         Store 2     4.5
         Store 3     7.0
         Name: Cost, dtype: float64

In [29]: df
```

Out[29]:

	Cost	Item Purchased	Name	Location
Store 1	24.5	Dog Food	Chris	None
Store 2	4.5	Kitty Litter	Kevyn	None
Store 3	7.0	Bird Seed	Vinod	None

```
In [30]: !cat olympics.csv

0,1,2,3,4,5,6,7,8,9,10,11,12,13,14,15
, Summer,01 !,02 !,03 !,Total, Winter,01 !,02 !,03 !,Total, Games,01 !,02 !,03 !,Combined total
Afghanistană(AFG),13,0,0,2,2,0,0,0,0,0,13,0,0,2,2
Algeriaă(ALG),12,5,2,8,15,3,0,0,0,0,15,5,2,8,15
Argentinaă(ARG),23,18,24,28,70,18,0,0,0,0,41,18,24,28,70
```

Armeniaă(ARM),5,1,2,9,12,6,0,0,0,0,11,1,2,9,12
Australasiaă(ANZ) [ANZ],2,3,4,5,12,0,0,0,0,0,2,3,4,5,12
Australiaă(AUS) [AUS] [Z],25,139,152,177,468,18,5,3,4,12,43,144,155,181,480
Austriaă(AUT),26,18,33,35,86,22,59,78,81,218,48,77,111,116,304
Azerbaijană(AZE),5,6,5,15,26,5,0,0,0,0,10,6,5,15,26
Bahamasă(BAH),15,5,2,5,12,0,0,0,0,0,15,5,2,5,12
Bahrainăă(BRN),8,0,0,1,1,0,0,0,0,0,8,0,0,1,1
Barbadosă(BAR) [BAR],11,0,0,1,1,0,0,0,0,0,11,0,0,1,1
Belarusă(BLR),5,12,24,39,75,6,6,4,5,15,11,18,28,44,90
Belgiumă(BEL),25,37,52,53,142,20,1,1,3,5,45,38,53,56,147
Bermudaă(BER),17,0,0,1,1,7,0,0,0,0,24,0,0,1,1
Bohemiaă(BOH) [BOH] [Z],3,0,1,3,4,0,0,0,0,0,3,0,1,3,4
Botswanaă(BOT),9,0,1,0,1,0,0,0,0,0,9,0,1,0,1
Brazilă(BRA),21,23,30,55,108,7,0,0,0,0,28,23,30,55,108
British West Indiesă(BWI) [BWI],1,0,0,2,2,0,0,0,0,0,1,0,0,2,2
Bulgariaă(BUL) [H],19,51,85,78,214,19,1,2,3,6,38,52,87,81,220
Burundiă(BDI),5,1,0,0,1,0,0,0,0,0,5,1,0,0,1
Cameroonă(CMR),13,3,1,1,5,1,0,0,0,0,14,3,1,1,5
Canadaă(CAN),25,59,99,121,279,22,62,56,52,170,47,121,155,173,449
Chileă(CHI) [I],22,2,7,4,13,16,0,0,0,0,38,2,7,4,13
Chinaă(CHN) [CHN],9,201,146,126,473,10,12,22,19,53,19,213,168,145,526
Colombiaă(COL),18,2,6,11,19,1,0,0,0,0,19,2,6,11,19
Costa Ricaă(CRC),14,1,1,2,4,6,0,0,0,0,20,1,1,2,4
Ivory Coastă(CIV) [CIV],12,0,1,0,1,0,0,0,0,0,12,0,1,0,1
Croatiaă(CRO),6,6,7,10,23,7,4,6,1,11,13,10,13,11,34
Cubaă(CUB) [Z],19,72,67,70,209,0,0,0,0,0,19,72,67,70,209
Cyprusă(CYP),9,0,1,0,1,10,0,0,0,0,19,0,1,0,1
Czech Republicăă(CZE) [CZE],5,14,15,15,44,6,7,9,8,24,11,21,24,23,68
Czechoslovakiaă(TCH) [TCH],16,49,49,45,143,16,2,8,15,25,32,51,57,60,168
Denmarkă(DEN) [Z],26,43,68,68,179,13,0,1,0,1,39,43,69,68,180
Djiboutiă(DJI) [B],7,0,0,1,1,0,0,0,0,0,7,0,0,1,1
Dominican Republicăă(DOM),13,3,2,1,6,0,0,0,0,0,13,3,2,1,6
Ecuadorăă(ECU),13,1,1,0,2,0,0,0,0,0,13,1,1,0,2
Egyptă(EGY) [EGY] [Z],21,7,9,10,26,1,0,0,0,0,22,7,9,10,26
Eritreaă(ERI),4,0,0,1,1,0,0,0,0,0,4,0,0,1,1
Estoniaă(EST),11,9,9,15,33,9,4,2,1,7,20,13,11,16,40
Ethiopiaă(ETH),12,21,7,17,45,2,0,0,0,0,14,21,7,17,45
Finlandă(FIN),24,101,84,117,302,22,42,62,57,161,46,143,146,174,463
Franceă(FRA) [O] [P] [Z],27,202,223,246,671,22,31,31,47,109,49,233,254,293,780
Gabonă(GAB),9,0,1,0,1,0,0,0,0,0,9,0,1,0,1
Georgiaă(GEO),5,6,5,14,25,6,0,0,0,0,11,6,5,14,25
Germanyă(GER) [GER] [Z],15,174,182,217,573,11,78,78,53,209,26,252,260,270,782
United Team of Germanyă(EUA) [EUA],3,28,54,36,118,3,8,6,5,19,6,36,60,41,137
East Germanyă(GDR) [GDR],5,153,129,127,409,6,39,36,35,110,11,192,165,162,519
West Germanyă(FRG) [FRG],5,56,67,81,204,6,11,15,13,39,11,67,82,94,243
Ghanaă(GHA) [GHA],13,0,1,3,4,1,0,0,0,0,14,0,1,3,4
Great Britainăă(GBR) [GBR] [Z],27,236,272,272,780,22,10,4,12,26,49,246,276,284,806
Greeceă(GRE) [Z],27,30,42,39,111,18,0,0,0,0,45,30,42,39,111

Grenadaă(GRN),8,1,0,0,1,0,0,0,0,0,8,1,0,0,1
Guatemalaă(GUA),13,0,1,0,1,1,0,0,0,0,14,0,1,0,1
Guyanaă(GUY) [GUY],16,0,0,1,1,0,0,0,0,0,16,0,0,1,1
Haitiă(HAI) [J],14,0,1,1,2,0,0,0,0,0,14,0,1,1,2
Hong Kongă(HKG) [HKG],15,1,1,1,3,4,0,0,0,0,19,1,1,1,3
Hungaryă(HUN),25,167,144,165,476,22,0,2,4,6,47,167,146,169,482
Icelandă(ISL),19,0,2,2,4,17,0,0,0,0,36,0,2,2,4
Indiaă(IND) [F],23,9,6,11,26,9,0,0,0,0,32,9,6,11,26
Indonesiaă(INA),14,6,10,11,27,0,0,0,0,0,14,6,10,11,27
Irană(IRI) [K],15,15,20,25,60,10,0,0,0,0,25,15,20,25,60
Iraqă(IRQ),13,0,0,1,1,0,0,0,0,0,13,0,0,1,1
Irelandă(IRL),20,9,8,12,29,6,0,0,0,0,26,9,8,12,29
Israelă(ISR),15,1,1,5,7,6,0,0,0,0,21,1,1,5,7
Italyă(ITA) [M] [S],26,198,166,185,549,22,37,34,43,114,48,235,200,228,663
Jamaicaă(JAM) [JAM],16,17,30,20,67,7,0,0,0,0,23,17,30,20,67
Japană(JPN),21,130,126,142,398,20,10,17,18,45,41,140,143,160,443
Kazakhstană(KAZ),5,16,17,19,52,6,1,3,3,7,11,17,20,22,59
Kenyaă(KEN),13,25,32,29,86,3,0,0,0,0,16,25,32,29,86
North Koreaă(PRK),9,14,12,21,47,8,0,1,1,2,17,14,13,22,49
South Koreaă(KOR),16,81,82,80,243,17,26,17,10,53,33,107,99,90,296
Kuwaită(KUW),12,0,0,2,2,0,0,0,0,0,12,0,0,2,2
Kyrgyzstană(KGZ),5,0,1,2,3,6,0,0,0,0,11,0,1,2,3
Latviaă(LAT),10,3,11,5,19,10,0,4,3,7,20,3,15,8,26
Lebanonă(LIB),16,0,2,2,4,16,0,0,0,0,32,0,2,2,4
Liechtensteină(LIE),16,0,0,0,0,18,2,2,5,9,34,2,2,5,9
Lithuaniaă(LTU),8,6,5,10,21,8,0,0,0,0,16,6,5,10,21
Luxembourgă(LUX) [O],22,1,1,0,2,8,0,2,0,2,30,1,3,0,4
Macedoniaă(MKD),5,0,0,1,1,5,0,0,0,0,10,0,0,1,1
Malaysiaă(MAS) [MAS],12,0,3,3,6,0,0,0,0,0,12,0,3,3,6
Mauritiusă(MRI),8,0,0,1,1,0,0,0,0,0,8,0,0,1,1
Mexicoă(MEX),22,13,21,28,62,8,0,0,0,0,30,13,21,28,62
Moldovaă(MDA),5,0,2,5,7,6,0,0,0,0,11,0,2,5,7
Mongoliaă(MGL),12,2,9,13,24,13,0,0,0,0,25,2,9,13,24
Montenegroă(MNE),2,0,1,0,1,2,0,0,0,0,4,0,1,0,1
Moroccoă(MAR),13,6,5,11,22,6,0,0,0,0,19,6,5,11,22
Mozambiqueă(MOZ),9,1,0,1,2,0,0,0,0,0,9,1,0,1,2
Namibiaă(NAM),6,0,4,0,4,0,0,0,0,0,6,0,4,0,4
Netherlandsă(NED) [Z],25,77,85,104,266,20,37,38,35,110,45,114,123,139,376
Netherlands Antillesă(AHO) [AHO] [I],13,0,1,0,1,2,0,0,0,0,15,0,1,0,1
New Zealandă(NZL) [NZL],22,42,18,39,99,15,0,1,0,1,37,42,19,39,100
Nigeră(NIG),11,0,0,1,1,0,0,0,0,0,11,0,0,1,1
Nigeriaă(NGR),15,3,8,12,23,0,0,0,0,0,15,3,8,12,23
Norwayă(NOR) [Q],24,56,49,43,148,22,118,111,100,329,46,174,160,143,477
Pakistană(PAK),16,3,3,4,10,2,0,0,0,0,18,3,3,4,10
Panamaă(PAN),16,1,0,2,3,0,0,0,0,0,16,1,0,2,3
Paraguayă(PAR),11,0,1,0,1,1,0,0,0,0,12,0,1,0,1
Peruă(PER) [L],17,1,3,0,4,2,0,0,0,0,19,1,3,0,4
Philippinesă(PHI),20,0,2,7,9,4,0,0,0,0,24,0,2,7,9

```
Polandă(POL),20,64,82,125,271,22,6,7,7,20,42,70,89,132,291
Portugală(POR),23,4,8,11,23,7,0,0,0,0,30,4,8,11,23
Puerto Ricoă(PUR),17,0,2,6,8,6,0,0,0,0,23,0,2,6,8
Qatară(QAT),8,0,0,4,4,0,0,0,0,0,8,0,0,4,4
Romaniaă(ROU),20,88,94,119,301,20,0,0,1,1,40,88,94,120,302
Russiaă(RUS) [RUS],5,132,121,142,395,6,49,40,35,124,11,181,161,177,519
Russian Empireă(RU1) [RU1],3,1,4,3,8,0,0,0,0,0,3,1,4,3,8
Soviet Unionă(URS) [URS],9,395,319,296,1010,9,78,57,59,194,18,473,376,355,1204
Unified Teamă(EUN) [EUN],1,45,38,29,112,1,9,6,8,23,2,54,44,37,135
Saudi Arabiaă(KSA),10,0,1,2,3,0,0,0,0,0,10,0,1,2,3
Senegală(SEN),13,0,1,0,1,5,0,0,0,0,18,0,1,0,1
Serbiaă(SRB) [SRB],3,1,2,4,7,2,0,0,0,0,5,1,2,4,7
Serbia and Montenegroă(SCG) [SCG],3,2,4,3,9,3,0,0,0,0,6,2,4,3,9
Singaporeă(SIN),15,0,2,2,4,0,0,0,0,0,15,0,2,2,4
Slovakiaă(SVK) [SVK],5,7,9,8,24,6,2,2,1,5,11,9,11,9,29
Sloveniaă(SLO),6,4,6,9,19,7,2,4,9,15,13,6,10,18,34
South Africaă(RSA),18,23,26,27,76,6,0,0,0,0,24,23,26,27,76
Spainăă(ESP) [Z],22,37,59,35,131,19,1,0,1,2,41,38,59,36,133
Sri Lankaă(SRI) [SRI],16,0,2,0,2,0,0,0,0,0,16,0,2,0,2
Sudană(SUD),11,0,1,0,1,0,0,0,0,0,11,0,1,0,1
Surinameă(SUR) [E],11,1,0,1,2,0,0,0,0,0,11,1,0,1,2
Swedenă(SWE) [Z],26,143,164,176,483,22,50,40,54,144,48,193,204,230,627
Switzerlandă(SUI),27,47,73,65,185,22,50,40,48,138,49,97,113,113,323
Syriaă(SYR),12,1,1,1,3,0,0,0,0,0,12,1,1,1,3
Chinese Taipeiă(TPE) [TPE] [TPE2],13,2,7,12,21,11,0,0,0,0,24,2,7,12,21
Tajikistană(TJK),5,0,1,2,3,4,0,0,0,0,9,0,1,2,3
Tanzaniaă(TAN) [TAN],12,0,2,0,2,0,0,0,0,0,12,0,2,0,2
Thailandă(THA),15,7,6,11,24,3,0,0,0,0,18,7,6,11,24
Togoă(TOG),9,0,0,1,1,1,0,0,0,0,10,0,0,1,1
Tongaă(TGA),8,0,1,0,1,1,0,0,0,0,9,0,1,0,1
Trinidad and Tobagoă(TRI) [TRI],16,2,5,11,18,3,0,0,0,0,19,2,5,11,18
Tunisiaă(TUN),13,3,3,4,10,0,0,0,0,0,13,3,3,4,10
Turkeyă(TUR),21,39,25,24,88,16,0,0,0,0,37,39,25,24,88
Ugandaă(UGA),14,2,3,2,7,0,0,0,0,0,14,2,3,2,7
Ukraineă(UKR),5,33,27,55,115,6,2,1,4,7,11,35,28,59,122
United Arab Emiratesă(UAE),8,1,0,0,1,0,0,0,0,0,8,1,0,0,1
United Statesă(USA) [P] [Q] [R] [Z],26,976,757,666,2399,22,96,102,84,282,48,1072,859,750,2681
Uruguayă(URU),20,2,2,6,10,1,0,0,0,0,21,2,2,6,10
Uzbekistană(UZB),5,5,5,10,20,6,1,0,0,1,11,6,5,10,21
Venezuelaă(VEN),17,2,2,8,12,4,0,0,0,0,21,2,2,8,12
Vietnamă(VIE),14,0,2,0,2,0,0,0,0,0,14,0,2,0,2
Virgin Islandsă(ISV),11,0,1,0,1,7,0,0,0,0,18,0,1,0,1
Yugoslaviaă(YUG) [YUG],16,26,29,28,83,14,0,3,1,4,30,26,32,29,87
Independent Olympic Participantsă(IOP) [IOP],1,0,1,2,3,0,0,0,0,0,1,0,1,2,3
Zambiaă(ZAM) [ZAM],12,0,1,1,2,0,0,0,0,0,12,0,1,1,2
Zimbabweă(ZIM) [ZIM],12,3,4,1,8,1,0,0,0,0,13,3,4,1,8
Mixed teamă(ZZX) [ZZX],3,8,5,4,17,0,0,0,0,0,3,8,5,4,17
Totals,27,4809,4775,5130,14714,22,959,958,948,2865,49,5768,5733,6078,17579
```

In [31]:
```
df = pd.read_csv('olympics.csv')
df.head()
```

Out[31]:

	0	1	2	3	4	5	6	7	8
0	NaN	Summer	01 !	02 !	03 !	Total	Winter	01 !	02 !
1	Afghanistană(AFG)	13	0	0	2	2	0	0	0
2	Algeriaă(ALG)	12	5	2	8	15	3	0	0
3	Argentinaă(ARG)	23	18	24	28	70	18	0	0
4	Armeniaă(ARM)	5	1	2	9	12	6	0	0

	9	10	11	12	13	14	15
0	03 !	Total	Games	01 !	02 !	03 !	Combined total
1	0	0	13	0	0	2	2
2	0	0	15	5	2	8	15
3	0	0	41	18	24	28	70
4	0	0	11	1	2	9	12

In [32]:
```
df = pd.read_csv('olympics.csv', index_col = 0, skiprows=1)
df.head()
```

Out[32]:

	Summer	01 !	02 !	03 !	Total	Winter	01 !.1
Afghanistană(AFG)	13	0	0	2	2	0	0
Algeriaă(ALG)	12	5	2	8	15	3	0
Argentinaă(ARG)	23	18	24	28	70	18	0
Armeniaă(ARM)	5	1	2	9	12	6	0
Australasiaă(ANZ) [ANZ]	2	3	4	5	12	0	0

	02 !.1	03 !.1	Total.1	Games	01 !.2	02 !.2
Afghanistană(AFG)	0	0	0	13	0	0
Algeriaă(ALG)	0	0	0	15	5	2
Argentinaă(ARG)	0	0	0	41	18	24
Armeniaă(ARM)	0	0	0	11	1	2
Australasiaă(ANZ) [ANZ]	0	0	0	2	3	4

	03 !.2	Combined total
Afghanistană(AFG)	2	2
Algeriaă(ALG)	8	15
Argentinaă(ARG)	28	70
Armeniaă(ARM)	9	12
Australasiaă(ANZ) [ANZ]	5	12

In [33]:
```
df.columns
```

Out[33]:
```
Index([' Summer', '01 !', '02 !', '03 !', 'Total', ' Winter', '01 !.1',
       '02 !.1', '03 !.1', 'Total.1', ' Games', '01 !.2', '02 !.2', '03 !.2',
       'Combined total'],
      dtype='object')
```

```
In [34]: for col in df.columns:
             if col[:2]=='01':
                 df.rename(columns={col:'Gold' + col[4:]}, inplace=True)
             if col[:2]=='02':
                 df.rename(columns={col:'Silver' + col[4:]}, inplace=True)
             if col[:2]=='03':
                 df.rename(columns={col:'Bronze' + col[4:]}, inplace=True)
             if col[:1]=='':
                 df.rename(columns={col:'#' + col[1:]}, inplace=True)

         df.head()
```

Out[34]:

	# Summer	Gold	Silver	Bronze	Total	# Winter	\
Afghanistană(AFG)	13	0	0	2	2	0	
Algeriaă(ALG)	12	5	2	8	15	3	
Argentinaă(ARG)	23	18	24	28	70	18	
Armeniaă(ARM)	5	1	2	9	12	6	
Australasiaă(ANZ) [ANZ]	2	3	4	5	12	0	

	Gold.1	Silver.1	Bronze.1	Total.1	# Games	Gold.2	\
Afghanistană(AFG)	0	0	0	0	13	0	
Algeriaă(ALG)	0	0	0	0	15	5	
Argentinaă(ARG)	0	0	0	0	41	18	
Armeniaă(ARM)	0	0	0	0	11	1	
Australasiaă(ANZ) [ANZ]	0	0	0	0	2	3	

	Silver.2	Bronze.2	Combined total
Afghanistană(AFG)	0	2	2
Algeriaă(ALG)	2	8	15
Argentinaă(ARG)	24	28	70
Armeniaă(ARM)	2	9	12
Australasiaă(ANZ) [ANZ]	4	5	12

5 Querying a DataFrame

```
In [35]: df['Gold'] > 0
```

Out[35]:

```
Afghanistană(AFG)                 False
Algeriaă(ALG)                      True
Argentinaă(ARG)                    True
Armeniaă(ARM)                      True
Australasiaă(ANZ) [ANZ]            True
Australiaă(AUS) [AUS] [Z]          True
Austriaă(AUT)                      True
Azerbaijană(AZE)                   True
Bahamasă(BAH)                      True
Bahrainăă(BRN)                    False
Barbadosă(BAR) [BAR]              False
```

Belarusă(BLR)	True
Belgiumă(BEL)	True
Bermudaă(BER)	False
Bohemiaă(BOH) [BOH] [Z]	False
Botswanaă(BOT)	False
Brazilă(BRA)	True
British West Indiesă(BWI) [BWI]	False
Bulgariaă(BUL) [H]	True
Burundiă(BDI)	True
Cameroonă(CMR)	True
Canadaă(CAN)	True
Chileă(CHI) [I]	True
Chinaă(CHN) [CHN]	True
Colombiaă(COL)	True
Costa Ricaă(CRC)	True
Ivory Coastă(CIV) [CIV]	False
Croatiaă(CRO)	True
Cubaă(CUB) [Z]	True
Cyprusă(CYP)	False
	...
Sri Lankaă(SRI) [SRI]	False
Sudană(SUD)	False
Surinameă(SUR) [E]	True
Swedenă(SWE) [Z]	True
Switzerlandă(SUI)	True
Syriaă(SYR)	True
Chinese Taipeiă(TPE) [TPE] [TPE2]	True
Tajikistană(TJK)	False
Tanzaniaă(TAN) [TAN]	False
Thailandă(THA)	True
Togoă(TOG)	False
Tongaă(TGA)	False
Trinidad and Tobagoă(TRI) [TRI]	True
Tunisiaă(TUN)	True
Turkeyă(TUR)	True
Ugandaă(UGA)	True
Ukraineă(UKR)	True
United Arab Emiratesă(UAE)	True
United Statesă(USA) [P] [Q] [R] [Z]	True
Uruguayă(URU)	True
Uzbekistană(UZB)	True
Venezuelaă(VEN)	True
Vietnamă(VIE)	False
Virgin Islandsă(ISV)	False
Yugoslaviaă(YUG) [YUG]	True
Independent Olympic Participantsă(IOP) [IOP]	False
Zambiaă(ZAM) [ZAM]	False
Zimbabweă(ZIM) [ZIM]	True

```
Mixed teamă(ZZX) [ZZX]                          True
Totals                                          True
Name: Gold, dtype: bool
```

```
In [36]: only_gold = df.where(df['Gold'] > 0)
         only_gold.head()
```

Out[36]:

	# Summer	Gold	Silver	Bronze	Total	# Winter	\
Afghanistană(AFG)	NaN	NaN	NaN	NaN	NaN	NaN	
Algeriaă(ALG)	12.0	5.0	2.0	8.0	15.0	3.0	
Argentinaă(ARG)	23.0	18.0	24.0	28.0	70.0	18.0	
Armeniaă(ARM)	5.0	1.0	2.0	9.0	12.0	6.0	
Australasiaă(ANZ) [ANZ]	2.0	3.0	4.0	5.0	12.0	0.0	

	Gold.1	Silver.1	Bronze.1	Total.1	# Games	Gold.2	\
Afghanistană(AFG)	NaN	NaN	NaN	NaN	NaN	NaN	
Algeriaă(ALG)	0.0	0.0	0.0	0.0	15.0	5.0	
Argentinaă(ARG)	0.0	0.0	0.0	0.0	41.0	18.0	
Armeniaă(ARM)	0.0	0.0	0.0	0.0	11.0	1.0	
Australasiaă(ANZ) [ANZ]	0.0	0.0	0.0	0.0	2.0	3.0	

	Silver.2	Bronze.2	Combined total
Afghanistană(AFG)	NaN	NaN	NaN
Algeriaă(ALG)	2.0	8.0	15.0
Argentinaă(ARG)	24.0	28.0	70.0
Armeniaă(ARM)	2.0	9.0	12.0
Australasiaă(ANZ) [ANZ]	4.0	5.0	12.0

```
In [37]: only_gold['Gold'].count()
```

```
Out[37]: 100
```

```
In [38]: df['Gold'].count()
```

```
Out[38]: 147
```

```
In [39]: only_gold = only_gold.dropna()
         only_gold.head()
```

Out[39]:

	# Summer	Gold	Silver	Bronze	Total	# Winter	\
Algeriaă(ALG)	12.0	5.0	2.0	8.0	15.0	3.0	
Argentinaă(ARG)	23.0	18.0	24.0	28.0	70.0	18.0	
Armeniaă(ARM)	5.0	1.0	2.0	9.0	12.0	6.0	
Australasiaă(ANZ) [ANZ]	2.0	3.0	4.0	5.0	12.0	0.0	
Australiaă(AUS) [AUS] [Z]	25.0	139.0	152.0	177.0	468.0	18.0	

	Gold.1	Silver.1	Bronze.1	Total.1	# Games	\
Algeriaă(ALG)	0.0	0.0	0.0	0.0	15.0	
Argentinaă(ARG)	0.0	0.0	0.0	0.0	41.0	

```
Armeniaă(ARM)                 0.0       0.0       0.0       0.0      11.0
Australasiaă(ANZ) [ANZ]       0.0       0.0       0.0       0.0       2.0
Australiaă(AUS) [AUS] [Z]     5.0       3.0       4.0      12.0      43.0

                           Gold.2  Silver.2  Bronze.2  Combined total
Algeriaă(ALG)                 5.0       2.0       8.0            15.0
Argentinaă(ARG)              18.0      24.0      28.0            70.0
Armeniaă(ARM)                 1.0       2.0       9.0            12.0
Australasiaă(ANZ) [ANZ]       3.0       4.0       5.0            12.0
Australiaă(AUS) [AUS] [Z]   144.0     155.0     181.0           480.0
```

```
In [40]: only_gold = df[df['Gold'] > 0]
         only_gold.head()
```

```
Out[40]:                          # Summer  Gold  Silver  Bronze  Total  # Winter  \
         Algeriaă(ALG)                  12     5       2       8     15         3
         Argentinaă(ARG)                23    18      24      28     70        18
         Armeniaă(ARM)                   5     1       2       9     12         6
         Australasiaă(ANZ) [ANZ]         2     3       4       5     12         0
         Australiaă(AUS) [AUS] [Z]      25   139     152     177    468        18

                           Gold.1  Silver.1  Bronze.1  Total.1  # Games  \
         Algeriaă(ALG)          0         0         0        0       15
         Argentinaă(ARG)        0         0         0        0       41
         Armeniaă(ARM)          0         0         0        0       11
         Australasiaă(ANZ) [ANZ]        0         0         0        0        2
         Australiaă(AUS) [AUS] [Z]      5         3         4       12       43

                           Gold.2  Silver.2  Bronze.2  Combined total
         Algeriaă(ALG)          5         2         8              15
         Argentinaă(ARG)       18        24        28              70
         Armeniaă(ARM)          1         2         9              12
         Australasiaă(ANZ) [ANZ]        3         4         5              12
         Australiaă(AUS) [AUS] [Z]    144       155       181             480
```

```
In [ ]: len(df[(df['Gold'] > 0) | (df['Gold.1'] > 0)])
```

```
In [ ]: df[(df['Gold.1'] > 0) & (df['Gold'] == 0)]
```

6 Indexing Dataframes

```
In [41]: df.head()
```

```
Out[41]:                        # Summer  Gold  Silver  Bronze  Total  # Winter  \
         Afghanistaă(AFG)              13     0       0       2      2         0
         Algeriaă(ALG)                 12     5       2       8     15         3
         Argentinaă(ARG)               23    18      24      28     70        18
         Armeniaă(ARM)                  5     1       2       9     12         6
```

```
Australasiaă(ANZ) [ANZ]         2     3       4       5     12         0

                           Gold.1  Silver.1  Bronze.1  Total.1  # Games  Gold.2  \
Afghanistană(AFG)               0         0         0        0       13       0
Algeriaă(ALG)                   0         0         0        0       15       5
Argentinaă(ARG)                 0         0         0        0       41      18
Armeniaă(ARM)                   0         0         0        0       11       1
Australasiaă(ANZ) [ANZ]         0         0         0        0        2       3

                           Silver.2  Bronze.2  Combined total
Afghanistană(AFG)                 0         2               2
Algeriaă(ALG)                     2         8              15
Argentinaă(ARG)                  24        28              70
Armeniaă(ARM)                     2         9              12
Australasiaă(ANZ) [ANZ]           4         5              12
```

In [42]:
```
df['country'] = df.index
df = df.set_index('Gold')
df.head()
```

Out[42]:
```
      # Summer  Silver  Bronze  Total  # Winter  Gold.1  Silver.1  Bronze.1  \
Gold
0           13       0       2      2         0       0         0         0
5           12       2       8     15         3       0         0         0
18          23      24      28     70        18       0         0         0
1            5       2       9     12         6       0         0         0
3            2       4       5     12         0       0         0         0

      Total.1  # Games  Gold.2  Silver.2  Bronze.2  Combined total  \
Gold
0           0       13       0         0         2               2
5           0       15       5         2         8              15
18          0       41      18        24        28              70
1           0       11       1         2         9              12
3           0        2       3         4         5              12

                       country
Gold
0           Afghanistană(AFG)
5               Algeriaă(ALG)
18            Argentinaă(ARG)
1               Armeniaă(ARM)
3     Australasiaă(ANZ) [ANZ]
```

In [43]:
```
df = df.reset_index()
df.head()
```

Out[43]:
```
   Gold  # Summer  Silver  Bronze  Total  # Winter  Gold.1  Silver.1  \
0     0        13       0       2      2         0       0         0
```

```
1     5        12       2       8     15         3       0         0
2    18        23      24      28     70        18       0         0
3     1         5       2       9     12         6       0         0
4     3         2       4       5     12         0       0         0

   Bronze.1  Total.1  # Games  Gold.2  Silver.2  Bronze.2  Combined total  \
0         0        0       13       0         0         2               2
1         0        0       15       5         2         8              15
2         0        0       41      18        24        28              70
3         0        0       11       1         2         9              12
4         0        0        2       3         4         5              12

                    country
0         Afghanistană(AFG)
1             Algeriaă(ALG)
2           Argentinaă(ARG)
3             Armeniaă(ARM)
4  Australasiaă(ANZ) [ANZ]
```

In [44]:
```
df = pd.read_csv('census.csv')
df.head()
```

Out[44]:
```
   SUMLEV  REGION  DIVISION  STATE  COUNTY   STNAME         CTYNAME  \
0      40       3         6      1       0  Alabama         Alabama
1      50       3         6      1       1  Alabama  Autauga County
2      50       3         6      1       3  Alabama  Baldwin County
3      50       3         6      1       5  Alabama  Barbour County
4      50       3         6      1       7  Alabama     Bibb County

   CENSUS2010POP  ESTIMATESBASE2010  POPESTIMATE2010     ...       \
0        4779736            4780127          4785161     ...
1          54571              54571            54660     ...
2         182265             182265           183193     ...
3          27457              27457            27341     ...
4          22915              22919            22861     ...

   RDOMESTICMIG2011  RDOMESTICMIG2012  RDOMESTICMIG2013  RDOMESTICMIG2014  \
0          0.002295         -0.193196          0.381066          0.582002
1          7.242091         -2.915927         -3.012349          2.265971
2         14.832960         17.647293         21.845705         19.243287
3         -4.728132         -2.500690         -7.056824         -3.904217
4         -5.527043         -5.068871         -6.201001         -0.177537

   RDOMESTICMIG2015  RNETMIG2011  RNETMIG2012  RNETMIG2013  RNETMIG2014  \
0         -0.467369     1.030015     0.826644     1.383282     1.724718
1         -2.530799     7.606016    -2.626146    -2.722002     2.592270
2         17.197872    15.844176    18.559627    22.727626    20.317142
3        -10.543299    -4.874741    -2.758113    -7.167664    -3.978583
```

```
4          0.177258    -5.088389    -4.363636    -5.403729     0.754533

   RNETMIG2015
0     0.712594
1    -2.187333
2    18.293499
3   -10.543299
4     1.107861

[5 rows x 100 columns]
```

In [45]:
```
df['SUMLEV'].unique()
```

Out[45]:
```
array([40, 50])
```

In [46]:
```
df=df[df['SUMLEV'] == 50]
df.head()
```

Out[46]:

	SUMLEV	REGION	DIVISION	STATE	COUNTY	STNAME	CTYNAME	CENSUS2010POP	ESTIMATESBASE2010	POPESTIMATE2010	...	RDOMESTICMIG2011	RDOMESTICMIG2012	RDOMESTICMIG2013	RDOMESTICMIG2014	RDOMESTICMIG2015	RNETMIG2011	RNETMIG2012	RNETMIG2013	RNETMIG2014	RNETMIG2015
1	50	3	6	1	1	Alabama	Autauga County	54571	54571	54660	...	7.242091	-2.915927	-3.012349	2.265971	-2.530799	7.606016	-2.626146	-2.722002	2.592270	-2.187333
2	50	3	6	1	3	Alabama	Baldwin County	182265	182265	183193	...	14.832960	17.647293	21.845705	19.243287	17.197872	15.844176	18.559627	22.727626	20.317142	18.293499
3	50	3	6	1	5	Alabama	Barbour County	27457	27457	27341	...	-4.728132	-2.500690	-7.056824	-3.904217	-10.543299	-4.874741	-2.758113	-7.167664	-3.978583	
4	50	3	6	1	7	Alabama	Bibb County	22915	22919	22861	...	-5.527043	-5.068871	-6.201001	-0.177537	0.177258	-5.088389	-4.363636	-5.403729	0.754533	
5	50	3	6	1	9	Alabama	Blount County	57322	57322	57373	...	1.807375	-1.177622	-1.748766	-2.062535	-1.369970	1.859511	-0.848580	-1.402476	-1.577232	

```
3   -10.543299
4     1.107861
5    -0.884411

[5 rows x 100 columns]
```

```
In [ ]: columns_to_keep = ['STNAME',
                          'CTYNAME',
                          'BIRTHS2010',
                          'BIRTHS2011',
                          'BIRTHS2012',
                          'BIRTHS2013',
                          'BIRTHS2014',
                          'BIRTHS2015',
                          'POPESTIMATE2010',
                          'POPESTIMATE2011',
                          'POPESTIMATE2012',
                          'POPESTIMATE2013',
                          'POPESTIMATE2014',
                          'POPESTIMATE2015']
        df = df[columns_to_keep]
        df.head()
```

```
In [47]: df = df.set_index(['STNAME', 'CTYNAME'])
         df.head()
```

Out[47]:

STNAME	CTYNAME	SUMLEV	REGION	DIVISION	STATE	COUNTY	\
Alabama	Autauga County	50	3	6	1	1	
	Baldwin County	50	3	6	1	3	
	Barbour County	50	3	6	1	5	
	Bibb County	50	3	6	1	7	
	Blount County	50	3	6	1	9	

STNAME	CTYNAME	CENSUS2010POP	ESTIMATESBASE2010	POPESTIMATE2010	\
Alabama	Autauga County	54571	54571	54660	
	Baldwin County	182265	182265	183193	
	Barbour County	27457	27457	27341	
	Bibb County	22915	22919	22861	
	Blount County	57322	57322	57373	

STNAME	CTYNAME	POPESTIMATE2011	POPESTIMATE2012	...	\
Alabama	Autauga County	55253	55175	...	
	Baldwin County	186659	190396	...	
	Barbour County	27226	27159	...	
	Bibb County	22733	22642	...	

```
                Blount County             57711            57776     ...

                                    RDOMESTICMIG2011  RDOMESTICMIG2012  RDOMESTICMIG2013  \
STNAME  CTYNAME
Alabama Autauga County                  7.242091         -2.915927         -3.012349
        Baldwin County                 14.832960         17.647293         21.845705
        Barbour County                 -4.728132         -2.500690         -7.056824
        Bibb County                    -5.527043         -5.068871         -6.201001
        Blount County                   1.807375         -1.177622         -1.748766

                                    RDOMESTICMIG2014  RDOMESTICMIG2015  RNETMIG2011  \
STNAME  CTYNAME
Alabama Autauga County                  2.265971         -2.530799     7.606016
        Baldwin County                 19.243287         17.197872    15.844176
        Barbour County                 -3.904217        -10.543299    -4.874741
        Bibb County                    -0.177537          0.177258    -5.088389
        Blount County                  -2.062535         -1.369970     1.859511

                                    RNETMIG2012  RNETMIG2013  RNETMIG2014  RNETMIG2015
STNAME  CTYNAME
Alabama Autauga County                -2.626146    -2.722002     2.592270    -2.187333
        Baldwin County                18.559627    22.727626    20.317142    18.293499
        Barbour County                -2.758113    -7.167664    -3.978583   -10.543299
        Bibb County                   -4.363636    -5.403729     0.754533     1.107861
        Blount County                 -0.848580    -1.402476    -1.577232    -0.884411

        [5 rows x 98 columns]

In [48]: df.loc['Michigan', 'Washtenaw County']

Out[48]: SUMLEV                     50.000000
         REGION                      2.000000
         DIVISION                    3.000000
         STATE                      26.000000
         COUNTY                    161.000000
         CENSUS2010POP          344791.000000
         ESTIMATESBASE2010      345066.000000
         POPESTIMATE2010        345563.000000
         POPESTIMATE2011        349048.000000
         POPESTIMATE2012        351213.000000
         POPESTIMATE2013        354289.000000
         POPESTIMATE2014        357029.000000
         POPESTIMATE2015        358880.000000
         NPOPCHG_2010              497.000000
         NPOPCHG_2011             3485.000000
         NPOPCHG_2012             2165.000000
         NPOPCHG_2013             3076.000000
         NPOPCHG_2014             2740.000000
```

```
NPOPCHG_2015               1851.000000
BIRTHS2010                  977.000000
BIRTHS2011                 3826.000000
BIRTHS2012                 3780.000000
BIRTHS2013                 3662.000000
BIRTHS2014                 3683.000000
BIRTHS2015                 3709.000000
DEATHS2010                  542.000000
DEATHS2011                 2053.000000
DEATHS2012                 2070.000000
DEATHS2013                 2136.000000
DEATHS2014                 2197.000000
                              ...
RBIRTH2011                   11.016238
RBIRTH2012                   10.795975
RBIRTH2013                   10.381260
RBIRTH2014                   10.355425
RBIRTH2015                   10.361652
RDEATH2011                    5.911222
RDEATH2012                    5.912081
RDEATH2013                    6.055263
RDEATH2014                    6.177265
RDEATH2015                    6.126477
RNATURALINC2011               5.105016
RNATURALINC2012               4.883893
RNATURALINC2013               4.325998
RNATURALINC2014               4.178159
RNATURALINC2015               4.235175
RINTERNATIONALMIG2011         5.061826
RINTERNATIONALMIG2012         5.557928
RINTERNATIONALMIG2013         6.007070
RINTERNATIONALMIG2014         6.756472
RINTERNATIONALMIG2015         6.674033
RDOMESTICMIG2011              0.129569
RDOMESTICMIG2012             -4.309822
RDOMESTICMIG2013             -1.780293
RDOMESTICMIG2014             -2.955078
RDOMESTICMIG2015             -6.078985
RNETMIG2011                   5.191395
RNETMIG2012                   1.248106
RNETMIG2013                   4.226778
RNETMIG2014                   3.801394
RNETMIG2015                   0.595048
Name: (Michigan, Washtenaw County), dtype: float64
```

```
In [49]: df.loc[ [('Michigan', 'Washtenaw County'),
                  ('Michigan', 'Wayne County')] ]
```

```
Out[49]:                          SUMLEV  REGION  DIVISION  STATE  COUNTY  \
```

```
STNAME   CTYNAME
Michigan Washtenaw County      50       2         3     26     161
         Wayne County          50       2         3     26     163

                               CENSUS2010POP  ESTIMATESBASE2010  POPESTIMATE2010  \
STNAME   CTYNAME
Michigan Washtenaw County         344791             345066           345563
         Wayne County            1820584            1820641          1815199

                               POPESTIMATE2011  POPESTIMATE2012     ...       \
STNAME   CTYNAME                                                   ...
Michigan Washtenaw County           349048           351213     ...
         Wayne County              1801273          1792514     ...

                               RDOMESTICMIG2011  RDOMESTICMIG2012  \
STNAME   CTYNAME
Michigan Washtenaw County          0.129569         -4.309822
         Wayne County            -13.340073        -10.271616

                               RDOMESTICMIG2013  RDOMESTICMIG2014  \
STNAME   CTYNAME
Michigan Washtenaw County          -1.780293         -2.955078
         Wayne County            -14.119617        -11.903253

                               RDOMESTICMIG2015  RNETMIG2011  RNETMIG2012  \
STNAME   CTYNAME
Michigan Washtenaw County          -6.078985     5.191395     1.248106
         Wayne County              -8.762835   -11.344758    -8.098421

                               RNETMIG2013  RNETMIG2014  RNETMIG2015
STNAME   CTYNAME
Michigan Washtenaw County     4.226778     3.801394     0.595048
         Wayne County       -11.732437    -9.161648    -6.010195

[2 rows x 98 columns]
```

7 Missing values

```
In [50]: df = pd.read_csv('log.csv')
         df
```

```
Out[50]:          time    user          video  playback position paused  volume
         0   1469974424  cheryl     intro.html                  5  False    10.0
         1   1469974454  cheryl     intro.html                  6    NaN     NaN
         2   1469974544  cheryl     intro.html                  9    NaN     NaN
         3   1469974574  cheryl     intro.html                 10    NaN     NaN
         4   1469977514     bob     intro.html                  1    NaN     NaN
```

```
5   1469977544     bob     intro.html                  1    NaN     NaN
6   1469977574     bob     intro.html                  1    NaN     NaN
7   1469977604     bob     intro.html                  1    NaN     NaN
8   1469974604  cheryl     intro.html                 11    NaN     NaN
9   1469974694  cheryl     intro.html                 14    NaN     NaN
10  1469974724  cheryl     intro.html                 15    NaN     NaN
11  1469974454     sue  advanced.html                 24    NaN     NaN
12  1469974524     sue  advanced.html                 25    NaN     NaN
13  1469974424     sue  advanced.html                 23  False    10.0
14  1469974554     sue  advanced.html                 26    NaN     NaN
15  1469974624     sue  advanced.html                 27    NaN     NaN
16  1469974654     sue  advanced.html                 28    NaN     5.0
17  1469974724     sue  advanced.html                 29    NaN     NaN
18  1469974484  cheryl     intro.html                  7    NaN     NaN
19  1469974514  cheryl     intro.html                  8    NaN     NaN
20  1469974754     sue  advanced.html                 30    NaN     NaN
21  1469974824     sue  advanced.html                 31    NaN     NaN
22  1469974854     sue  advanced.html                 32    NaN     NaN
23  1469974924     sue  advanced.html                 33    NaN     NaN
24  1469977424     bob     intro.html                  1   True    10.0
25  1469977454     bob     intro.html                  1    NaN     NaN
26  1469977484     bob     intro.html                  1    NaN     NaN
27  1469977634     bob     intro.html                  1    NaN     NaN
28  1469977664     bob     intro.html                  1    NaN     NaN
29  1469974634  cheryl     intro.html                 12    NaN     NaN
30  1469974664  cheryl     intro.html                 13    NaN     NaN
31  1469977694     bob     intro.html                  1    NaN     NaN
32  1469977724     bob     intro.html                  1    NaN     NaN
```

In [51]:
```
df.fillna?
```

In [52]:
```
df = df.set_index('time')
df = df.sort_index()
df
```

Out[52]:

time	user	video	playback position	paused	volume
1469974424	cheryl	intro.html	5	False	10.0
1469974424	sue	advanced.html	23	False	10.0
1469974454	cheryl	intro.html	6	NaN	NaN
1469974454	sue	advanced.html	24	NaN	NaN
1469974484	cheryl	intro.html	7	NaN	NaN
1469974514	cheryl	intro.html	8	NaN	NaN
1469974524	sue	advanced.html	25	NaN	NaN
1469974544	cheryl	intro.html	9	NaN	NaN
1469974554	sue	advanced.html	26	NaN	NaN
1469974574	cheryl	intro.html	10	NaN	NaN
1469974604	cheryl	intro.html	11	NaN	NaN

```
1469974624     sue  advanced.html                27    NaN     NaN
1469974634  cheryl     intro.html                12    NaN     NaN
1469974654     sue  advanced.html                28    NaN     5.0
1469974664  cheryl     intro.html                13    NaN     NaN
1469974694  cheryl     intro.html                14    NaN     NaN
1469974724  cheryl     intro.html                15    NaN     NaN
1469974724     sue  advanced.html                29    NaN     NaN
1469974754     sue  advanced.html                30    NaN     NaN
1469974824     sue  advanced.html                31    NaN     NaN
1469974854     sue  advanced.html                32    NaN     NaN
1469974924     sue  advanced.html                33    NaN     NaN
1469977424     bob     intro.html                 1   True    10.0
1469977454     bob     intro.html                 1    NaN     NaN
1469977484     bob     intro.html                 1    NaN     NaN
1469977514     bob     intro.html                 1    NaN     NaN
1469977544     bob     intro.html                 1    NaN     NaN
1469977574     bob     intro.html                 1    NaN     NaN
1469977604     bob     intro.html                 1    NaN     NaN
1469977634     bob     intro.html                 1    NaN     NaN
1469977664     bob     intro.html                 1    NaN     NaN
1469977694     bob     intro.html                 1    NaN     NaN
1469977724     bob     intro.html                 1    NaN     NaN
```

In [53]:
```
df = df.reset_index()
df = df.set_index(['time', 'user'])
df
```

Out[53]:

time	user	video	playback position	paused	volume
1469974424	cheryl	intro.html	5	False	10.0
	sue	advanced.html	23	False	10.0
1469974454	cheryl	intro.html	6	NaN	NaN
	sue	advanced.html	24	NaN	NaN
1469974484	cheryl	intro.html	7	NaN	NaN
1469974514	cheryl	intro.html	8	NaN	NaN
1469974524	sue	advanced.html	25	NaN	NaN
1469974544	cheryl	intro.html	9	NaN	NaN
1469974554	sue	advanced.html	26	NaN	NaN
1469974574	cheryl	intro.html	10	NaN	NaN
1469974604	cheryl	intro.html	11	NaN	NaN
1469974624	sue	advanced.html	27	NaN	NaN
1469974634	cheryl	intro.html	12	NaN	NaN
1469974654	sue	advanced.html	28	NaN	5.0
1469974664	cheryl	intro.html	13	NaN	NaN
1469974694	cheryl	intro.html	14	NaN	NaN
1469974724	cheryl	intro.html	15	NaN	NaN
	sue	advanced.html	29	NaN	NaN
1469974754	sue	advanced.html	30	NaN	NaN

```
1469974824 sue     advanced.html                 31    NaN     NaN
1469974854 sue     advanced.html                 32    NaN     NaN
1469974924 sue     advanced.html                 33    NaN     NaN
1469977424 bob        intro.html                  1   True    10.0
1469977454 bob        intro.html                  1    NaN     NaN
1469977484 bob        intro.html                  1    NaN     NaN
1469977514 bob        intro.html                  1    NaN     NaN
1469977544 bob        intro.html                  1    NaN     NaN
1469977574 bob        intro.html                  1    NaN     NaN
1469977604 bob        intro.html                  1    NaN     NaN
1469977634 bob        intro.html                  1    NaN     NaN
1469977664 bob        intro.html                  1    NaN     NaN
1469977694 bob        intro.html                  1    NaN     NaN
1469977724 bob        intro.html                  1    NaN     NaN
```

In [54]:
```
df = df.fillna(method='ffill')
df.head()
```

Out[54]:

time	user	video	playback position	paused	volume
1469974424	cheryl	intro.html	5	False	10.0
	sue	advanced.html	23	False	10.0
1469974454	cheryl	intro.html	6	False	10.0
	sue	advanced.html	24	False	10.0
1469974484	cheryl	intro.html	7	False	10.0

In [ ]: