hw2

July 17, 2025

[129]:

```
import numpy as np
import pandas as pd
```

[130]:

```
# NUMBER 1
data1 = np.arange(100,113)

# NUMBER 1A
print("shape:", data1.shape)

# NUMBER 1B
print("type:", type(data1))
```

```
shape: (13,)
type: <class 'numpy.ndarray'>
```

[131]:

```
# NUMBER 1C

boolean1 = (data1 > 105) & (data1 <= 110)
print(boolean1)
```

```
[False False False False False False  True  True  True  True  True False
 False]
```

[132]:

```
# NUMBER 1D

data1[boolean1] = 0
print(data1)
```

```
[100 101 102 103 104 105   0   0   0   0   0 111 112]
```

[133]:

```
# NUMBER 1E
data2 = data1[5:11].copy()
data2[:] = 0
print(data2)
```

```
[0 0 0 0 0 0]
```

[161]:

```
# NUMBER 2

random1 = np.random.randint(0, 100, size=(4,3))
```

```
df_random1 = pd.DataFrame(random1)
df_random1
```

[161]:

```
    0   1   2
0  17  77  93
1  66  96  81
2  57  83  10
3  39   4  94
```

[162]:

```
ds1 = df_random1.iloc[0].max()
ds2 = df_random1.iloc[1].max()
ds3 = df_random1.iloc[2].max()
ds4 = df_random1.iloc[3].max()

ds_max = pd.Series((ds1, ds2, ds3, ds4), index=['Row 1 Max',
                                                 'Row 2 Max', 'Row 3 Max', 'Row␣
 ↪4 Max'])
ds_max
```

[162]:

```
Row 1 Max    93
Row 2 Max    96
Row 3 Max    83
Row 4 Max    94
dtype: int64
```

[136]:

```
max_random1 = np.max(df_random1, axis=1)
max_random1
```

[136]:

```
0    66
1    45
2    95
3    59
dtype: int64
```

[137]:

```
# NUMBER 3
year = pd.Series([1991, 1992, 1993, 1994, 1995,
                  1996, 1997, 1998, 1999, 2000])
year
```

[137]:

```
0    1991
1    1992
2    1993
3    1994
4    1995
5    1996
6    1997
```

```
7    1998
8    1999
9    2000
dtype: int64
```

[138]:

```
# NUMBER 3A
year.size
```

[138]:

```
10
```

[139]:

```
# NUMBER 3B
rainfall = pd.Series([12.09, 12.35, 12.51, 10.25,
10.18, 10.59, 10.26, 10.48, 8.67, 10.23])
rainfall
```

[139]:

```
0    12.09
1    12.35
2    12.51
3    10.25
4    10.18
5    10.59
6    10.26
7    10.48
8     8.67
9    10.23
dtype: float64
```

[140]:

```
# NUMBER 3C
rainfall_normalized = (rainfall - rainfall.mean())/rainfall.std()
rainfall_normalized
```

[140]:

```
0    1.107402
1    1.324050
2    1.457371
3   -0.425796
4   -0.484124
5   -0.142487
6   -0.417463
7   -0.234146
8   -1.742346
9   -0.442461
dtype: float64
```

[141]:

```
# NUMBER 3D
df_rainfall_values = pd.DataFrame({
    'year': year,
    'rainfall': rainfall
})
```

```
df_rainfall_values
```

[141]:

```
   year  rainfall
0  1991     12.09
1  1992     12.35
2  1993     12.51
3  1994     10.25
4  1995     10.18
5  1996     10.59
6  1997     10.26
7  1998     10.48
8  1999      8.67
9  2000     10.23
```

[142]:

```
df_rainfall_values[df_rainfall_values['rainfall'] < 11]['year']
```

[142]:

```
3    1994
4    1995
5    1996
6    1997
7    1998
8    1999
9    2000
Name: year, dtype: int64
```

[143]:

```
# NUMBER 3E
df_rainfall_values.loc[df_rainfall_values['year'].between(1996, 2000),␣
  ↪'rainfall'] = np.nan
df_rainfall_values['rainfall']
```

[143]:

```
0    12.09
1    12.35
2    12.51
3    10.25
4    10.18
5      NaN
6      NaN
7      NaN
8      NaN
9      NaN
Name: rainfall, dtype: float64
```

[144]:

```
# NUMBER 3F

df_rainfall_values['rainfall'] = df_rainfall_values['rainfall'].fillna(0)
df_rainfall_values
```

[144]:

```
   year  rainfall
0  1991     12.09
1  1992     12.35
2  1993     12.51
3  1994     10.25
4  1995     10.18
5  1996      0.00
6  1997      0.00
7  1998      0.00
8  1999      0.00
9  2000      0.00
```

[145]:

```
# NUMBER 4

df_cars = pd.read_csv('../../data/cars.csv')
df_cars
```

[145]:

```
      MPG  CYL    ENG   WGT
0    18.0    8  307.0  3504
1    15.0    8  350.0  3693
2    18.0    8  318.0  3436
3    16.0    8  304.0  3433
4    17.0    8  302.0  3449
..    …    …     …     …
387  27.0    4  140.0  2790
388  44.0    4   97.0  2130
389  32.0    4  135.0  2295
390  28.0    4  120.0  2625
391  31.0    4  119.0  2720

[392 rows x 4 columns]
```

[146]:

```
# NUMBER 4A
correlation = np.corrcoef(df_cars['WGT'], df_cars['MPG'])
correlation = correlation[0,1]
print(f'Correlation between WGT and MPGs is: {correlation}. Since the number is␣
  ↪negative, we can conclude that a heavier vehicle will produce a less␣
  ↪efficient MPG rating.')
```

```
Correlation between WGT and MPGs is: -0.8322442135675991. Since the number is
negative, we can conclude that a heavier vehicle will produce a less efficient
MPG rating.
```

[147]:

```
# NUMBER 4B
cyl = df_cars['CYL'].value_counts()
cyl
```

```
[147]: CYL
       4    199
       8    103
       6     83
       3      4
       5      3
       Name: count, dtype: int64
```

```
[148]: # NUMBER 4C
       df_cars['ENG2WGT'] = df_cars['ENG'] / df_cars['WGT']
       df_cars[['ENG', 'WGT', 'ENG2WGT']].head()
```

```
[148]:      ENG   WGT   ENG2WGT
       0  307.0  3504  0.087614
       1  350.0  3693  0.094774
       2  318.0  3436  0.092549
       3  304.0  3433  0.088552
       4  302.0  3449  0.087562
```

```
[149]: # NUMBER 5

       df_kaggle = pd.read_csv('../../data/kaggle-uber-other-federal.csv')
       df_kaggle.head()
```

```
[149]:          Date      Time                                   PU_Address  \
       0  07/01/2014  07:15 AM  Brooklyn Museum, 200 Eastern Pkwy., BK NY;
       1  07/01/2014  07:30 AM              33 Robert Dr., Short Hills NJ;
       2  07/01/2014  08:00 AM                    60 Glenmore Ave., BK NY;
       3  07/01/2014  09:00 AM                     128 East 31 St., BK NY;
       4  07/01/2014  09:30 AM                 139-39 35 Ave., Flushing NY;

                                                 DO_Address  \
       0                           1 Brookdale Plaza, BK NY;
       1  John F Kennedy International Airport, vitona A…
       2                            2171 Nostrand Ave., BK NY;
       3                                  369 93rd St., BK NY;
       4                                   La Guardia Airport;

                                            Routing Details  \
       0  PU: Brooklyn Museum, 200 Eastern Pkwy., BK NY;…
       1  PU: 33 Robert Dr., Short Hills NJ; DO: John F …
       2  PU: 60 Glenmore Ave., BK NY; DO: 2171 Nostrand…
       3  PU: 128 East 31 St., BK NY; DO: 369 93rd St., …
       4  PU: 139-39 35 Ave., Flushing NY; DO: La Guardi…

                                               PU_Address.1     Status
       0  Brooklyn Museum, 200 Eastern Pkwy., BK NY; DO:…  Cancelled
```

```
1  33 Robert Dr., Short Hills NJ; DO: John F Kenn…    Arrived
2  60 Glenmore Ave., BK NY; DO: 2171 Nostrand Ave…   Assigned
3  128 East 31 St., BK NY; DO: 369 93rd St., BK NY;    Assigned
4  139-39 35 Ave., Flushing NY; DO: La Guardia Ai…   Assigned
```

[150]:
```
# NUMBER 5A
df_new_kaggle = df_kaggle[['Date', 'Time', 'Status', 'PU_Address']].copy()
df_new_kaggle['Datetime'] = pd.to_datetime(df_new_kaggle['Date'] + ' ' +␣
 ↪df_new_kaggle['Time'])
df_new_kaggle
```

[150]:
```
          Date      Time     Status  \
0   07/01/2014  07:15 AM  Cancelled
1   07/01/2014  07:30 AM    Arrived
2   07/01/2014  08:00 AM   Assigned
3   07/01/2014  09:00 AM   Assigned
4   07/01/2014  09:30 AM   Assigned
..         …         …        …
94  07/21/2014  06:00 AM   Assigned
95  07/21/2014  08:30 AM  Cancelled
96  07/21/2014  12:00 PM    Arrived
97  07/21/2014  04:45 PM   Assigned
98  07/22/2014  01:30 PM    Arrived

                                           PU_Address            Datetime
0          Brooklyn Museum, 200 Eastern Pkwy., BK NY; 2014-07-01 07:15:00
1                        33 Robert Dr., Short Hills NJ; 2014-07-01 07:30:00
2                              60 Glenmore Ave., BK NY; 2014-07-01 08:00:00
3                               128 East 31 St., BK NY; 2014-07-01 09:00:00
4                        139-39 35 Ave., Flushing NY; 2014-07-01 09:30:00
..                                                  …                   …
94                266 prospect park west, brooklyn NY; 2014-07-21 06:00:00
95                                       42 St., BK NY; 2014-07-21 08:30:00
96                              663 51st Street, BK NY; 2014-07-21 12:00:00
97                    255 Fieldston Terrace, Bronx NY; 2014-07-21 16:45:00
98  Columbia University, 630 W 168 St., NY NY; ST:… 2014-07-22 13:30:00

[99 rows x 5 columns]
```

[151]:
```
# NUMBER 5B
df_new_kaggle.dtypes
```

[151]:
```
Date                      object
Time                      object
Status                    object
PU_Address                object
Datetime          datetime64[ns]
```

```
dtype: object
```

[152]:

```
df_new_kaggle['Date'] = pd.to_datetime(df_new_kaggle['Date']).dt.date
df_new_kaggle['Time'] = pd.to_timedelta(df_new_kaggle['Time'] + ':00')
df_new_kaggle['Status'] = df_new_kaggle['Status'].astype('category')
df_new_kaggle['PU_Address'] = df_new_kaggle['PU_Address'].astype('category')

df_new_kaggle.dtypes
```

[152]:

```
Date                   object
Time          timedelta64[ns]
Status               category
PU_Address           category
Datetime       datetime64[ns]
dtype: object
```

[153]:

```
# NUMBER 5C
df_new_kaggle['Hour'] = df_new_kaggle['Datetime'].dt.hour
df_new_kaggle.head()
```

[153]:

```
         Date            Time     Status  \
0  2014-07-01 0 days 07:15:00  Cancelled
1  2014-07-01 0 days 07:30:00    Arrived
2  2014-07-01 0 days 08:00:00   Assigned
3  2014-07-01 0 days 09:00:00   Assigned
4  2014-07-01 0 days 09:30:00   Assigned

                                   PU_Address            Datetime  Hour
0  Brooklyn Museum, 200 Eastern Pkwy., BK NY; 2014-07-01 07:15:00     7
1              33 Robert Dr., Short Hills NJ; 2014-07-01 07:30:00     7
2                    60 Glenmore Ave., BK NY; 2014-07-01 08:00:00     8
3                     128 East 31 St., BK NY; 2014-07-01 09:00:00     9
4                139-39 35 Ave., Flushing NY; 2014-07-01 09:30:00     9
```

[154]:

```
# NUMBER 5D
df_new_kaggle['Date'] = df_new_kaggle['Datetime'].dt.date
df_new_kaggle.set_index('Date', inplace=True)
df_new_kaggle
```

[154]:

```
                      Time     Status  \
Date
2014-07-01 0 days 07:15:00  Cancelled
2014-07-01 0 days 07:30:00    Arrived
2014-07-01 0 days 08:00:00   Assigned
2014-07-01 0 days 09:00:00   Assigned
2014-07-01 0 days 09:30:00   Assigned
…                    …        …
```

```
2014-07-21 0 days 06:00:00   Assigned
2014-07-21 0 days 08:30:00  Cancelled
2014-07-21 0 days 12:00:00    Arrived
2014-07-21 0 days 04:45:00   Assigned
2014-07-22 0 days 01:30:00    Arrived

                                                   PU_Address  \
Date
2014-07-01         Brooklyn Museum, 200 Eastern Pkwy., BK NY;
2014-07-01                     33 Robert Dr., Short Hills NJ;
2014-07-01                           60 Glenmore Ave., BK NY;
2014-07-01                            128 East 31 St., BK NY;
2014-07-01                       139-39 35 Ave., Flushing NY;
…                                                    …
2014-07-21               266 prospect park west, brooklyn NY;
2014-07-21                                     42 St., BK NY;
2014-07-21                            663 51st Street, BK NY;
2014-07-21                   255 Fieldston Terrace, Bronx NY;
2014-07-22  Columbia University, 630 W 168 St., NY NY; ST:…

                      Datetime  Hour
Date
2014-07-01 2014-07-01 07:15:00     7
2014-07-01 2014-07-01 07:30:00     7
2014-07-01 2014-07-01 08:00:00     8
2014-07-01 2014-07-01 09:00:00     9
2014-07-01 2014-07-01 09:30:00     9
…                         …   …
2014-07-21 2014-07-21 06:00:00     6
2014-07-21 2014-07-21 08:30:00     8
2014-07-21 2014-07-21 12:00:00    12
2014-07-21 2014-07-21 16:45:00    16
2014-07-22 2014-07-22 13:30:00    13

[99 rows x 5 columns]
```

[155]:

```
# NUMBER 5E
df_new_kaggle.loc[pd.to_datetime('07/02/2014').date()].shape[0]
```

[155]: 4

[156]:

```
# NUMBER 5F
df_new_kaggle.reset_index()
df_new_kaggle.head()
```

[156]:

```
                      Time     Status  \
Date
```

```
2014-07-01 0 days 07:15:00  Cancelled
2014-07-01 0 days 07:30:00    Arrived
2014-07-01 0 days 08:00:00   Assigned
2014-07-01 0 days 09:00:00   Assigned
2014-07-01 0 days 09:30:00   Assigned

                                            PU_Address             Datetime  \
Date
2014-07-01  Brooklyn Museum, 200 Eastern Pkwy., BK NY; 2014-07-01 07:15:00
2014-07-01              33 Robert Dr., Short Hills NJ; 2014-07-01 07:30:00
2014-07-01                    60 Glenmore Ave., BK NY; 2014-07-01 08:00:00
2014-07-01                     128 East 31 St., BK NY; 2014-07-01 09:00:00
2014-07-01                139-39 35 Ave., Flushing NY; 2014-07-01 09:30:00

            Hour
Date
2014-07-01     7
2014-07-01     7
2014-07-01     8
2014-07-01     9
2014-07-01     9
```

[157]:
```
# NUMBER 6A
'''
1) Which manufacturer offers cereals with the highest average rating?
2) Which cereals provide the highest protein content per serving?
3) What are the average sugar levels by manufacturer, which ones are lower?
4) Which cereal provides the most vitamins and fiber?
'''
```

[157]:
```
'\n1) Which manufacturer offers cereals with the highest average rating?\n2)
Which cereals provide the highest protein content per serving?\n3) What are the
average sugar levels by manufacturer, which ones are lower?\n4) Which cereal
provides the most vitamins and fiber? \n'
```

[158]:
```
# NUMBER 6B
'''
COLUMNS I WOULD CHOOSE TO ANSWER THE QUESTIONS ABOVE:

Name
mfr
protein
fiber
sugars
vitamins
rating

```

```
Probably approach the problem accessing the dataframe so:

df_cereal[['protein', 'sugars', 'fiber', 'rating', 'vitamins', 'mfr']].
 ↪describe()

to check sum:

df_cereal.isna().sum()
'''
```

[158]:

```
"\nCOLUMNS I WOULD CHOOSE TO ANSWER THE QUESTIONS
ABOVE:\n\nName\nmfr\nprotein\nfiber\nsugars\nvitamins\nrating\n\nProbably
approach the problem accessing the dataframe so:\n\ndf_cereal[['protein',
'sugars', 'fiber', 'rating', 'vitamins', 'mfr']].describe()\n\nto check
sum:\n\ndf_cereal.isna().sum()\n"
```

[159]:

```
# NUMBER 6C
'''
1) Bar plot: Average rating per manufacturer to visualize the brands
2) Scatter Plot: Fiber vs Ratings, this can showcase the fiber content in high␣
 ↪ratings cereals
3) Box Plot: Sugar content by manufacturer, comparing sugar levels
4) Histogram: Distribution of protein content from each cereal manufacturer
'''
```

[159]:

```
'\n1) Bar plot: Average rating per manufacturer to visualize the brands\n2)
Scatter Plot: Fiber vs Ratings, this can showcase the fiber content in high
ratings cereals\n3) Box Plot: Sugar content by manufacturer, comparing sugar
levels\n4) Histogram: Distribution of protein content from each cereal
manufacturer\n'
```

[160]:

```
# NUMBER 6D

'''
OTHER INFORMATION OR DATA TO THAT MAY BE HELPFUL?

1) Cost per Serving - only thing left out from that columns list I can
think of as an important factor to include. To compare affordability
between the different cereal products.

'''
```

[160]:

```
'\nOTHER INFORMATION OR DATA TO THAT MAY BE HELPFUL?\n\n1) Cost per Serving -
only thing left out from that columns list I can\nthink of as an important
factor to include. To compare affordability \nbetween the different cereal
products.\n\n'
```