```
import pandas as pd
import numpy as np
import tensorflow as tf


def main():
    ###############################################################
    # Problem #1
    ###############################################################
    team_list = [
        "Golden State",
        "49ers",
        "Giants",
        "Cavaliers",
        "Lakers",
        "Rams",
        "Yankees"
    ]
    print("Problem 1: ", team_list)

    team_series1 = pd.Series(team_list)
    print("Problem 1 - team_series 1:\n", team_series1)

    ###############################################################
    # Problem #2
    ###############################################################
    team_dict = {
        'Oakland-Basketball': 'Golden State',
        'SF-Football': '49ers',
        'SF-Baseball': 'Giants',
        'Cleveland-Basketball': 'Cavaliers',
        'LA-Basketball': 'Lakers',
        'LA-Football': 'Rams',
        'NY-Baseball': 'Yankees',
    }

    print("Problem 2: ", team_dict)
    team_series2 = pd.Series(team_dict)

    print("Problem 2 - team_series 2:\n", team_series2)
    ###############################################################
    # Problem #3
    ###############################################################
    team_series1.index = [
        'one',
        'two',
        'three',
        'four',
        'five',
        'six',
```

```
        'seven'
    ]
    print("Problem 3 - 5th for teams_series1 via loc: ",
          team_series1.loc['five'])
    print("Problem 3 - 5th for teams_series1 via iloc: ",
team_series1.iloc[4])
    print("Problem 3 - 7th for teams_series1 via loc: ",
          team_series1.loc['seven'])
    print("Problem 3 - 7th for teams_series1 via iloc: ",
team_series1.iloc[6])
    print("Problem 3 - 5th for teams_series2 via loc: ",
          team_series2.loc[[False, False, False, False, True, False,
False]].iloc[0])
    print("Problem 3 - 5th for teams_series2 via iloc: ",
team_series2.iloc[4])
    print("Problem 3 - 7th for teams_series2 via loc: ",
          team_series2.loc[[False, False, False, False, False, False,
True]].iloc[0])
    print("Problem 3 - 7th for teams_series2 via iloc: ",
team_series2.iloc[6])
    ###############################################################
    # Problem #4
    ###############################################################
    num = pd.Series(range(1, 101, 1))
    series_sum = 0
    for series_int in num:
        series_sum = series_sum + series_int

    print("Problem 4 - sum by loop: ", series_sum)
    print("Problem 4 - sum by np: ", np.sum(series_sum))
    ###############################################################
    # Problem #5
    ###############################################################
    print("Problem 5 - 5 added to series num:\n", (num+5))
    ###############################################################
    # Problem #6
    ###############################################################
    idx = range(1, 10)
    media_data = {"TV": pd.Series([230.1, 44.5, 17.2, 151.5, 180.8,
8.7, 57.5, 120.2, 8.6],
                                  index=idx),
                  "Radio": pd.Series([37.8, 39.3, 45.9, 41.3, 10.8,
48.9, 32.8, 19.6, 2.1],
                                     index=idx),
                  "NewsPaper": pd.Series([69.2, 45.1, 69.3, 58.5,
58.4, 75, 23.5, 11.6, 1],
                                         index=idx),
                  "Sales": pd.Series([22.1, 10.4, 9.3, 18.5, 12.9,
7.2, 11.8, 13.2, 4.8],
                                     index=idx)}
```

```
    print("Problem 6 - dataframe:\n", pd.DataFrame(media_data))
    ###############################################################
    # Problem #7
    ###############################################################
    house_data = pd.read_csv("00 kc_house_data.csv")
    print("Problem 7 - Number of observations in dataframe is ",
          (house_data.shape[0])*house_data.shape[1])
    print("Problem 7 - Average house price is ",
house_data['price'].mean())
    house_gt500k = house_data['price'].gt(500000)
    print("Problem 7 - Number of houses which are priced greater than
$500,000 is ",
          house_data[house_gt500k].shape[0])
    ###############################################################
    # Problem #8
    ###############################################################
    print("Problem 8 - tensorflow version is ", tf.__version__)


if __name__ == "__main__":
    main()


#########################
# Start of Results
#########################

Problem 1:  ['Golden State', '49ers', 'Giants', 'Cavaliers', 'Lakers',
'Rams', 'Yankees']
Problem 1 - team_series 1:
 0    Golden State
1           49ers
2          Giants
3       Cavaliers
4          Lakers
5            Rams
6         Yankees
dtype: object
Problem 2:  {'Cleveland-Basketball': 'Cavaliers', 'NY-Baseball':
'Yankees', 'SF-Football': '49ers', 'LA-Basketball': 'Lakers', 'SF-
Baseball': 'Giants', 'Oakland-Basketball': 'Golden State', 'LA-
Football': 'Rams'}
Problem 2 - team_series 2:
 Cleveland-Basketball       Cavaliers
LA-Basketball                 Lakers
LA-Football                     Rams
NY-Baseball                  Yankees
Oakland-Basketball      Golden State
SF-Baseball                   Giants
SF-Football                    49ers
```

```
dtype: object
Problem 3 - 5th for teams_series1 via loc:  Lakers
Problem 3 - 5th for teams_series1 via iloc:  Lakers
Problem 3 - 7th for teams_series1 via loc:  Yankees
Problem 3 - 7th for teams_series1 via iloc:  Yankees
Problem 3 - 5th for teams_series2 via loc:  Golden State
Problem 3 - 5th for teams_series2 via iloc:  Golden State
Problem 3 - 7th for teams_series2 via loc:  49ers
Problem 3 - 7th for teams_series2 via iloc:  49ers
Problem 4 - sum by loop:  5050
Problem 4 - sum by np:  5050
Problem 5 - 5 added to series num:
 0       6
1        7
2        8
3        9
4       10
5       11
6       12
7       13
8       14
9       15
10      16
11      17
12      18
13      19
14      20
15      21
16      22
17      23
18      24
19      25
20      26
21      27
22      28
23      29
24      30
25      31
26      32
27      33
28      34
29      35
     ...
70      76
71      77
72      78
73      79
74      80
75      81
76      82
```

```
77     83
78     84
79     85
80     86
81     87
82     88
83     89
84     90
85     91
86     92
87     93
88     94
89     95
90     96
91     97
92     98
93     99
94    100
95    101
96    102
97    103
98    104
99    105
Length: 100, dtype: int64
Problem 6 - dataframe:
    NewsPaper  Radio  Sales     TV
1        69.2   37.8   22.1  230.1
2        45.1   39.3   10.4   44.5
3        69.3   45.9    9.3   17.2
4        58.5   41.3   18.5  151.5
5        58.4   10.8   12.9  180.8
6        75.0   48.9    7.2    8.7
7        23.5   32.8   11.8   57.5
8        11.6   19.6   13.2  120.2
9         1.0    2.1    4.8    8.6
Problem 7 - Number of observations in dataframe is  453873
Problem 7 - Average house price is  540088.1417665294
Problem 7 - Number of houses which are priced greater than $500,000 is
9053
Problem 8 - tensorflow version is  1.12.0
```