assi1bandc

-1

September 3,

2024

```
[ ]: #B]
     import pandas as pd
     data = pd.read_csv("/content/RegularSeasonCompactResults
     (1).csv") data
```

```
[ ]:            Season     DayNum      WTea  WScor     LTeamID      LScore WLoc    Num
     0          1985            20   1228      81           1328     64          N0
     1          1985            25   1106      77           1354     70          H0
     2          1985            25   1112      63           1223     56          H0
     3          1985            25   1165      70           1432     54          H0
     4          1985            25   1192      86           1447     74          H0
     …          …       …        …      …     …         …      …     …
     16154    2019           132   1153      69           1222     57          N0
     16154    2019           132   1209      73           1426     64          N0
     16154    2019           132   1277      65           1276     60          N0
     16155    2019           132   1387      55           1382     53          N0
     16155    2019           132   1463      97           1217     85          H0

[161552 rows x 8 columns]
```

```
[ ]: data.head()
```

```
[ ]:            Season  DayNum WTeamI  WScore  LTeamID   LScore WLoc    NumOT
          0       1985      20   1228      81     1328      64     N      0
          1       1985      25   1106      77     1354      70     H      0
          2       1985      25   1112      63     1223      56     H      0
          3       1985      25   1165      70     1432      54     H      0
          4       1985      25   1192      86     1447      74     H      0
```

```
[ ]: data.tail()
```

```
[ ]:            Season  DayNum WTeamI  WScore  LTeamID   LScore WLoc    NumOT
     161547     2019     132   1153      69     1222      57     N 0
     161548     2019     132   1209      73     1426      64     N 0
     161549     2019     132   1277      65     1276      60     N 0
     161550     2019     132   1387      55     1382      53     N 0
     161551     2019     132   1463      97     1217      85     H 0
```

```
[ ]: #to show number of records
     data.value_counts()
```

```
[ ]:       DayNum  WTeamID  WScore  LTeamID  LScore  WLoc  NumOT
  1985     20      1228     81      1328     64      N     0       1
  2009     120     1454     77      1260     68      H     0       1
                   1404     93      1399     75      H     0       1
                   1410     63      1312     49      A     0       1
                   1421     76      1457     68      H     0       1
                                                                 ..
  1998     73      1275     72      1185     64      H     0       1
                   1277     63      1458     40      H     0       1
                   1282     88      1302     62      A     0       1
                   1286     86      1169     77      A     0       1
  2019     132     1463     97      1217     85      H     0       1
     Name: count, Length: 161552, dtype: int64
```

```
[ ]: #to find missing values in each column
     data.isnull().sum()
```

```
[ ]: Season        0
     DayNum        0
     WTeamID       0
     WScore        0
     LTeamID       0
     LScore        0
     WLoc          0
     NumOT         0
     dtype: int64
```

```
[ ]: data.max()
```

```
[ ]: Season     2019
     DayNum      132
     WTeamID    1466
     WScore      186
     LTeamID    1466
     LScore      150
     WLoc          N
     NumOT         6
     dtype: object
```

```
[ ]: #unique values in each column
     data['Season'].unique()
```

```
[ ]: array([1985, 1986, 1987, 1988, 1989, 1990, 1991, 1992, 1993, 1994, 1995,
            1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, 2005,
        2007, 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016,
        2017, 2018, 2019])
```

```
[ ]: data['WTeamID'].unique()
```

```
[ ]: array([1228, 1106, 1112, 1165, 1192, 1218, 1242, 1260, 1305, 1307, 1344,
        1374,  1412,  1417,  1116,  1120,  1135,  1143,  1153,  1171,  1173,
        1177,  1193,  1196,  1206,  1210,  1211,  1234,  1243,  1257,  1277,
        1278,  1286,  1301,  1325,  1326,  1328,  1339,  1365,  1375,  1397,
        1403,  1423,  1429,  1437,  1438,  1439,  1443,  1455,  1241,  1268,
        1314,  1323,  1458,  1104,  1152,  1160,  1181,  1186,  1247,  1249,
        1276,  1338,  1368,  1376,  1378,  1391,  1396,  1398,  1399,  1428,
        1448,  1464,  1124,  1130,  1200,  1204,  1233,  1246,  1280,  1308,
        1337,  1348,  1373,  1379,  1409,  1444,  1451,  1139,  1154,  1229,
        1259,  1261,  1272,  1273,  1318,  1330,  1360,  1388,  1393,  1419,
        1447,  1161,  1298,  1332,  1347,  1353,  1359,  1361,  1386,  1400,
        1433,  1102,  1111,  1119,  1149,  1151,  1155,  1185,  1201,  1212,
        1267,  1292,  1319,  1320,  1371,  1382,  1385,  1390,  1402,  1425,
        1435,  1441,  1449,  1103,  1122,  1123,  1129,  1140,  1141,  1156,
        1174,  1183,  1202,  1203,  1222,  1227,  1231,  1235,  1238,  1239,
        1248,  1256,  1258,  1264,  1266,  1269,  1285,  1290,  1310,  1329,
        1333,  1345,  1349,  1352,  1380,  1405,  1408,  1410,  1418,  1424,
        1456,  1462,  1168,  1179,  1223,  1281,  1304,  1356,  1426,  1431,
        1114,  1132,  1163,  1180,  1217,  1225,  1296,  1389,  1395,  1133,
        1178,  1199,  1216,  1232,  1245,  1250,  1265,  1275,  1293,  1336,
        1384,  1401,  1416,  1427,  1452,  1134,  1322,  1440,  1450,  1461,
        1113,  1190,  1226,  1254,  1334,  1110,  1137,  1172,  1191,  1207,
        1221,  1224,  1270,  1279,  1299,  1309,  1311,  1351,  1387,  1414,
        1463,  1284,  1108,  1287,  1350,  1109,  1131,  1208,  1220,  1263,
        1331,  1432,  1187,  1145,  1150,  1184,  1321,  1364,  1434,  1209,
        1166,  1197,  1117,  1215,  1327,  1363,  1411,  1446,  1453,  1182,
        1343,  1162,  1335,  1406,  1121,  1147,  1306,  1354,  1283,  1317,
        1126,  1341,  1271,  1436,  1442,  1383,  1159,  1144,  1175,  1288,
        1253,  1274,  1362,  1164,  1457,  1148,  1420,  1157,  1358,  1372,
        1118,  1421,  1460,  1198,  1381,  1251,  1291,  1282,  1454,  1169,
        1302,  1158,  1422,  1369,  1138,  1170,  1404,  1407,  1194,  1214,
        1240,  1142,  1459,  1313,  1176,  1346,  1340,  1115,  1237,  1125,
        1107,  1219,  1357,  1189,  1324,  1392,  1105,  1127,  1415,  1289,
        1394,  1236,  1128,  1252,  1205,  1366,  1430,  1413,  1255,  1294,
        1355,  1344,  1205,  1316,  1312,  1146,  1445,  1195,  1167,  1300,
```

```
[ ]: data['LScore'].unique()
```

```
[ ]:      70,     56,     54,     74,     78,     44,     80,     89,     71,     72,     65,    58,
    62,  50,     60,     53,     48,     40,     52,     76,     55,     77,     66,     59,    47,
    49,  84,     67,     61,     68,     42,     92,     57,     63,     46,     73,     75,    88,
    85,  45,     69,     81,     82,     96,     51,     43,     83,     27,     37,     79,    41,
    93,    86, 105,         38,     87,     91,     32,       90, 140,         35,     95,     39,    98,
    97,    34,     36,       94, 100,       99,     31,  30, 101, 108, 107, 110, 104,
          109, 103, 114, 111,   25, 102, 115, 113, 119, 112, 106, 121, 133,
   127,               33, 126, 122, 125, 144, 150, 136, 116, 137, 141, 123,            28,
      118, 128,          29, 132, 131, 117,            21,     26,     23,     24,     22,   20,
          124])
```

```
[ ]: data['DayNum'].unique()
```

```
[ ]:      25,    26,    27,    28,    29,    30,    31,    32,    33,    34,    35,   36,
     37,  38,    39,    40,    41,    42,    43,    44,    45,    46,    47,    48,   49,
     50,  51,    52,    53,    54,    55,    56,    57,    58,    59,    60,    61,   62,
     63,  65,    66,    67,    68,    69,    70,    71,    72,    73,    74,    75,   76,
     77,  78,    79,    80,    81,    82,    83,    84,    85,    86,    87,    88,   89,
     90,  91,    92,    93,    94,    95,    96,    97,    98,   99, 100, 101, 102,
          103, 104, 105, 106, 107, 108, 109, 110, 111, 112, 113, 114, 115, 116,
          117, 118, 119, 120, 121, 122, 123, 124, 125, 126, 127, 128,
      129, 130, 131, 132,    24,    64,    18,    19,    21,    23,    16,    17,   22,
     11,  15,    12,    13,    14,    10,     8,     9,     0,     1,     2,     3,    4,
      5,   6,    7])
```

```
[ ]: data['WScore'].unique()
```

```
[ ]:      77,    63,    70,    86,    79,    64,    58,    98,     97, 103,     75,   91,
     87,  65,    92,    50,    47,    55,    76,     59, 106,     95,    66,    72,   80,
    109,  94,    85,    73,     68, 104,      93,     69, 102,     67,    60,    62,   90,
     89,  71,    99,    78,    74,    82,    84,    61,    56,    96,    57,    46,   54,
    112,  51,    49,     83, 111,      44, 128,      88,     53, 100, 115,     52, 101,
         45, 110, 108, 107, 105, 121,     48, 142, 114, 117,       42,    37, 127,
    113,  43,          41, 116, 132, 129, 119, 122, 151,         35, 138, 126, 118,
          123, 133, 120, 124, 130, 152, 139, 144, 125, 134, 141,            40, 135,
          136, 131, 162, 181, 147, 145, 173, 166, 150,        36,    39, 148, 157,
          137, 149, 172, 186, 140, 159, 146, 143, 155, 156,            38, 153,     34])
```

```
[ ]: data['LTeamID'].unique()
```

```
[ ]: array([1328, 1354, 1223, 1432, 1447, 1337, 1226, 1268, 1133, 1424, 1288,
            1438, 1411, 1397, 1225, 1368, 1391, 1306, 1388, 1184, 1159,
            1216, 1134, 1296, 1265, 1416, 1137, 1149, 1102, 1114, 1332,
            1317, 1231, 1145, 1453, 1186, 1144, 1384, 1248, 1287, 1334,
            1417, 1126, 1152, 1228, 1347, 1428, 1436, 1172, 1330, 1121,
            1249, 1440, 1200, 1264, 1242, 1455, 1414, 1387, 1183, 1211,
            1284, 1402, 1178, 1364, 1290, 1202, 1180, 1155, 1258, 1192,
            1341, 1135, 1254, 1151, 1220, 1405, 1238, 1431, 1253, 1363,
            1270, 1331, 1434, 1365, 1442, 1456, 1227, 1406, 1260, 1117,
            1163, 1305, 1247, 1165, 1418, 1281, 1109, 1309, 1321, 1285,
            1271, 1351, 1267, 1373, 1322, 1187, 1245, 1150, 1409, 1141,
            1412, 1221, 1122, 1199, 1399, 1197, 1140, 1444, 1398, 1162,
            1441, 1292, 1209, 1375, 1336, 1344, 1343, 1386, 1232, 1241,
            1110, 1335, 1283, 1325, 1206, 1166, 1311, 1147, 1433, 1250,
            1452, 1425, 1208, 1119, 1286, 1307, 1116, 1246, 1263, 1446,
            1382, 1427, 1113, 1161, 1396, 1350, 1201, 1429, 1450, 1338,
            1320, 1448, 1131, 1190, 1129, 1400, 1224, 1154, 1215, 1403,
            1185, 1378, 1234, 1212, 1461, 1451, 1160, 1229, 1111, 1139,
            1280, 1279, 1371, 1156, 1345, 1464, 1191, 1299, 1298, 1310,
            1168, 1352, 1269, 1360, 1426, 1380, 1175, 1132, 1257, 1339,
            1327, 1203, 1319, 1462, 1273, 1275, 1323, 1261, 1308, 1179,
            1443, 1104, 1173, 1359, 1318, 1143, 1348, 1376, 1408, 1193,
            1410, 1395, 1356, 1124, 1171, 1353, 1389, 1419, 1210, 1235,
            1278, 1326, 1103, 1458, 1182, 1204, 1177, 1301, 1266, 1385,
            1329, 1439, 1108, 1112, 1153, 1390, 1174, 1333, 1401, 1120,
            1449, 1243, 1435, 1233, 1304, 1293, 1277, 1361, 1437, 1276,
            1196, 1314, 1256, 1393, 1217, 1130, 1272, 1181, 1207, 1374,
            1164, 1362, 1274, 1118, 1420, 1372, 1157, 1421, 1358, 1148,
            1457, 1198, 1460, 1291, 1381, 1251, 1282, 1169, 1302, 1454,
            1170, 1138, 1422, 1369, 1158, 1404, 1194, 1407, 1142, 1214,
            1459, 1240, 1313, 1115, 1176, 1346, 1237, 1340, 1107, 1357,
            1324, 1189, 1105, 1392, 1125, 1219, 1289, 1415, 1127, 1236,
            1205, 1128, 1252, 1366, 1394, 1294, 1255, 1430, 1413, 1355,
            1244, 1295, 1316, 1445, 1146, 1312, 1167, 1300, 1367, 1342,
```

```
[ ]: data['WLoc'].unique()
```

```
[ ]: array(['N', 'H', 'A'], dtype=object)
```

```
[ ]: data['NumOT'].unique()
```

```
[ ]: array([0, 3, 1, 2, 4, 5, 6])
```

```
[ ]: #number of unique
       values
```

```
[ ]: 35
```

```
[ ]: data['NumOT'].nunique()
```

```
[ ]: 7
```

```
[ ]: data['WLoc'].nunique()
```

```
[ ]: 3
```

```
[ ]: data['WScore'].nunique()
```

```
[ ]: 130
[ ]: data['LTeamID'].nunique()
```

```
[ ]: 366
```

```
[ ]: data['DayNum'].nunique()
```

```
[ ]: 133
```

```
[ ]: data['WTeamID'].nunique()
```

```
[ ]: 366
```

```
[ ]: data['LScore'].nunique()
```

```
[ ]: 118
```

```
[ ]: data['Season'].value_counts()
```

```
[ ]: Season 2019
        2018                546
        2017                3
        2016                540
        2014                5
        2015                539
        2013                5
        2010                536
        2012                9
        2009                536
        2011                2
        2008                535
        2007                4
        2006                532
        2005                0
        2003                526
        2004                3
        2002                525
        2000                3
        2001                524
        1999                9
        1998                524
        1997                6
        1992                516
        1991                3
        1996                504
        1995                3
        1994                475
                            7
```

```
1990                                4045
1989                                4037
1993                                3982
1988                                3955
1987                                3915
1986                                3783
1985                                3737
Name: count, dtype: int64
```

```
[ ]: data['Season'].value_counts()
```

```
[ ]: Season 2019
       2018                         5
       2017                         4
       2016                         6
       2014                         3
       2015
       2013
       2010                         5
       2012                         4
       2009                         0
       2011                         5
       2008
       2007
       2006                         5
       2005                         3
       2003                         9
       2004                         5
       2002
       2000
       2001                         5
       1999                         3
       1998                         6
       1997                         9
       1992
       1991
       1996                         5
       1995                         3
       1994                         6
       1990                         2
       1989
       1993
       1988                         5
       1987                         3
       1986                         5
       1985                         4
```

```
Name: count, dtype: int64
```

```
[ ]: data['DayNum'].value_counts()
```

```
[ ]: DayNum
     75     4720
     96     4698
     89     4689
     103    4642
     110    4641
     ...
     6       124
     1        98
     2        15
     3        12
     0         2
     Name: count, Length: 133, dtype: int64
```

```
[ ]: data['WTeamID'].value_counts()
```

```
[ ]: WTeamID
     1181    901
     1242    884
     1246    845
     1314    839
     1112    820
     ...
     1465     12
     1466      7
     1289      6
     1118      6
     1327      3
     Name: count, Length: 366, dtype: int64
```

```
[ ]: data['WScore'].value_counts()
```

```
[ ]: WScore 75
       73                    5629
       76
       74                    5611
       77
                             5606
       172
       149                   5583
       157
       145                   5436
                             ...
```

```
34                                                        1
Name: count, Length: 130, dtype: int64
```

```
[ ]: data['LTeamID'].value_counts()
```

```
[ ]: LTeamID
                                               1152     744
                                               1271     730
                                               1341     695
                                               1363     681
                                               1306     676
               …
                                               1446      34
                                               1466      22
                                               1118      21
                                               1327      19
                                               1465      14
     Name: count, Length: 366, dtype: int64
```

```
[ ]: data['WLoc'].value_counts()
```

```
[ ]: WLoc
                                                  H     95878
                                                  A     49260
                                                  N     16414
     Name: count, dtype: int64
```

```
[ ]: data['NumOT'].value_counts()
```

```
[ ]: NumOT
                                                  0     155529
                                                  1       4989
                                                  2        844
                                                  3        152
                                                  4         32
                                                  5          5
                                                  6          1
     Name: count, dtype: int64
```

```
[ ]: data['LScore'].value_counts()
```

```
[ ]: LScore 64
     63                                                 591
     62                                                 3
     65                                                 590
     66                                                 7
                                                        588
```

```
          ...
137                                                 1
136                                                 1
150                                                 1
144                                                 1
124                                                 1
Name: count, Length: 118, dtype: int64
```

```
[ ]: #C] Basic function of numpy
       # Create a 1D NumPy array with values ranging from 10
to 49.
 import numpy as np
 array = np.arange(10, 50)
```

```
Array is:                                      [10 11 12 13 14 15 16 17 18 19 20 21 22 23
33
 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49]
```

```
[ ]: #Reshape the array into a 3x5 matrix.
       var = np.random.randint(10, 49,
       size=15) print(var)
       matrix = var.reshape(3, 5)
       print("Matrix: ",matrix)
```

```
[11 36 25 31 19 48 33 27 10 20 45 39 35 10 34]
Matrix:                                         [[11 36 25 31 19]
 [48 33 27 10 20]
 [45 39 35 10 34]]
```

```
[ ]: #Extract the elements that are divisible by 3 from the original array. div =
      array[array % 3 == 0]
      print("Divisible by 3: ",div)
```

```
Divisible by 3:                            [12 15 18 21 24 27 30 33 36 39 42 45 48]
```

```
[ ]: #Create two 3x3 NumPy arrays with random
      integers array1 = np.random.randint(0,
      10,size=(3, 3)) array2 = np.random.randint(0,
      10,size=(3, 3)) print("Array 1: ",array1)
      print("Array 2: ",array2)
```

```
                               Array 1:   [[2 0 9]
 [0 9 7]
 [1 6 5]]
                               Array 2:   [[5 2 1]
 [0 7 5]
 [2 3 0]]
```

```
[ ]: # Perform element-wise addition, subtraction, multiplication, and division
     addition = array1 + array2
     print("Addition of two array: ",addition)

     substraction = array1 - array2
     print("Substraction of two array: ",substraction)

     multiplication = array1 * array2
     print("Multiplication of two array: ",multiplication)

     division = array1/array2
     print("Division of two array: ",division)
```

```
Addition of two array:          [[ 7        2 10]
 [ 0 16 12]
   [ 3      9     5]]
Substraction of two array:                [[-3 -2           8]
   [ 0      2     2]
   [-1      3     5]]
Multiplication of two array:               [[10          0          9]
 [ 0 63 35]
      [ 2 18      0]]
Division of two array:          [[0.4                               0.         9.        ]
 [                  nan 1.28571429 1.4                            ]
 [0.5                     2.                      inf]]

 <ipython-input-37-584bebde1f12>:11: RuntimeWarning: divide by zero
 encountered in divide
   division = array1/array2
 <ipython-input-37-584bebde1f12>:11: RuntimeWarning: invalid value
 encountered in divide
   division = array1/array2
```

```
[ ]:
```