

```
In [1]: import numpy as np

In [2]: np.__version__
Out[2]: '2.1.3'

In [3]: #Import numpy
import numpy as np

#Seasons
Seasons = ["2015","2016","2017","2018","2019","2020","2021","2022","2023","2024"]
Sdict = {"2015":0,"2016":1,"2017":2,"2018":3,"2019":4,"2020":5,"2021":6,"2022":7

#Players
Players = ["Sachin","Rahul","Smith","Sami","Pollard","Morris","Samson","Dhoni","P
Pdict = {"Sachin":0,"Rahul":1,"Smith":2,"Sami":3,"Pollard":4,"Morris":5,"Samson":6

#Salaries
Sachin_Salary = [15946875,17718750,19490625,21262500,23034375,24806250,25244493,
Rahul_Salary = [12000000,12744189,13488377,14232567,14976754,16324500,18038573,1
Smith_Salary = [4621800,5828090,13041250,14410581,15779912,14500000,16022500,175
Sami_Salary = [3713640,4694041,13041250,14410581,15779912,17149243,18518574,1945
Pollard_Salary = [4493160,4806720,6061274,13758000,15202590,16647180,18091770,19
Morris_Salary = [3348000,4235220,12455000,14410581,15779912,14500000,16022500,17
Samson_Salary = [3144240,3380160,3615960,4574189,13520500,14940153,16359805,1777
Dhoni_Salary = [0,0,4171200,4484040,4796880,6053663,15506632,16669630,17832627,1
Kohli_Salary = [0,0,0,4822800,5184480,5546160,6993708,16402500,17632688,18862875
Sky_Salary = [3031920,3841443,13041250,14410581,15779912,14200000,15691000,17182
#Matrix
Salary = np.array([Sachin_Salary, Rahul_Salary, Smith_Salary, Sami_Salary, Polla

#Games
Sachin_G = [80,77,82,82,73,82,58,78,6,35]
Rahul_G = [82,57,82,79,76,72,60,72,79,80]
Smith_G = [79,78,75,81,76,79,62,76,77,69]
Sami_G = [80,65,77,66,69,77,55,67,77,40]
Pollard_G = [82,82,82,79,82,78,54,76,71,41]
Morris_G = [70,69,67,77,70,77,57,74,79,44]
Samson_G = [78,64,80,78,45,80,60,70,62,82]
Dhoni_G = [35,35,80,74,82,78,66,81,81,27]
Kohli_G = [40,40,40,81,78,81,39,0,10,51]
Sky_G = [75,51,51,79,77,76,49,69,54,62]
#Matrix
Games = np.array([Sachin_G, Rahul_G, Smith_G, Sami_G, Pollard_G, Morris_G, Samso

#Points
Sachin PTS = [2832,2430,2323,2201,1970,2078,1616,2133,83,782]
Rahul PTS = [1653,1426,1779,1688,1619,1312,1129,1170,1245,1154]
Smith PTS = [2478,2132,2250,2304,2258,2111,1683,2036,2089,1743]
Sami PTS = [2122,1881,1978,1504,1943,1970,1245,1920,2112,966]
Pollard PTS = [1292,1443,1695,1624,1503,1784,1113,1296,1297,646]
Morris PTS = [1572,1561,1496,1746,1678,1438,1025,1232,1281,928]
Samson PTS = [1258,1104,1684,1781,841,1268,1189,1186,1185,1564]
Dhoni PTS = [903,903,1624,1871,2472,2161,1850,2280,2593,686]
Kohli PTS = [597,597,597,1361,1619,2026,852,0,159,904]
Sky PTS = [2040,1397,1254,2386,2045,1941,1082,1463,1028,1331]
```

```
#Matrix  
Points = np.array([Sachin PTS, Rahul PTS, Smith PTS, Sami PTS, Pollard PTS, Morr
```

In [4]: Salary

```
Out[4]: array([[15946875, 17718750, 19490625, 21262500, 23034375, 24806250,  
    25244493, 27849149, 30453805, 23500000],  
    [12000000, 12744189, 13488377, 14232567, 14976754, 16324500,  
    18038573, 19752645, 21466718, 23180790],  
    [ 4621800,  5828090, 13041250, 14410581, 15779912, 14500000,  
    16022500, 17545000, 19067500, 20644400],  
    [ 3713640,  4694041, 13041250, 14410581, 15779912, 17149243,  
    18518574, 19450000, 22407474, 22458000],  
    [ 4493160,  4806720, 6061274, 13758000, 15202590, 16647180,  
    18091770, 19536360, 20513178, 21436271],  
    [ 3348000,  4235220, 12455000, 14410581, 15779912, 14500000,  
    16022500, 17545000, 19067500, 20644400],  
    [ 3144240,  3380160, 3615960, 4574189, 13520500, 14940153,  
    16359805, 17779458, 18668431, 20068563],  
    [      0,        0, 4171200, 4484040, 4796880, 6053663,  
    15506632, 16669630, 17832627, 18995624],  
    [      0,        0,        0, 4822800, 5184480, 5546160,  
    6993708, 16402500, 17632688, 18862875],  
    [ 3031920, 3841443, 13041250, 14410581, 15779912, 14200000,  
    15691000, 17182000, 18673000, 15000000]])
```

In [5]: Games

```
Out[5]: array([[ 80,  77,  82,  82,  73,  82,  58,  78,   6,  35],  
    [ 82,  57,  82,  79,  76,  72,  60,  72,  79,  80],  
    [ 79,  78,  75,  81,  76,  79,  62,  76,  77,  69],  
    [ 80,  65,  77,  66,  69,  77,  55,  67,  77,  40],  
    [ 82,  82,  82,  79,  82,  78,  54,  76,  71,  41],  
    [ 70,  69,  67,  77,  70,  77,  57,  74,  79,  44],  
    [ 78,  64,  80,  78,  45,  80,  60,  70,  62,  82],  
    [ 35,  35,  80,  74,  82,  78,  66,  81,  81,  27],  
    [ 40,  40,  40,  81,  78,  81,  39,   0,  10,  51],  
    [ 75,  51,  51,  79,  77,  76,  49,  69,  54,  62]])
```

In [6]: Points

```
Out[6]: array([[ 2832, 2430, 2323, 2201, 1970, 2078, 1616, 2133,   83,  782],  
    [1653, 1426, 1779, 1688, 1619, 1312, 1129, 1170, 1245, 1154],  
    [2478, 2132, 2250, 2304, 2258, 2111, 1683, 2036, 2089, 1743],  
    [2122, 1881, 1978, 1504, 1943, 1970, 1245, 1920, 2112,  966],  
    [1292, 1443, 1695, 1624, 1503, 1784, 1113, 1296, 1297,  646],  
    [1572, 1561, 1496, 1746, 1678, 1438, 1025, 1232, 1281,  928],  
    [1258, 1104, 1684, 1781,  841, 1268, 1189, 1186, 1185, 1564],  
    [ 903,  903, 1624, 1871, 2472, 2161, 1850, 2280, 2593,  686],  
    [ 597,  597,  597, 1361, 1619, 2026,   852,     0,  159,  904],  
    [2040, 1397, 1254, 2386, 2045, 1941, 1082, 1463, 1028, 1331]])
```

In [7]: mydata=np.arange(0,20)
print(mydata)

```
[ 0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19]
```

In [8]: np.reshape(mydata,(4,5))

```
Out[8]: array([[ 0,  1,  2,  3,  4],
   [ 5,  6,  7,  8,  9],
   [10, 11, 12, 13, 14],
   [15, 16, 17, 18, 19]])
```

```
In [9]: mydata
```

```
Out[9]: array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
   17, 18, 19])
```

```
In [38]: MATR1=mydata.reshape(5,4,order='c')
MATR1
```

```
Out[38]: array([[ 0,  1,  2,  3],
   [ 4,  5,  6,  7],
   [ 8,  9, 10, 11],
   [12, 13, 14, 15],
   [16, 17, 18, 19]])
```

```
In [39]: MATR1
```

```
Out[39]: array([[ 0,  1,  2,  3],
   [ 4,  5,  6,  7],
   [ 8,  9, 10, 11],
   [12, 13, 14, 15],
   [16, 17, 18, 19]])
```

```
In [41]: MATR1[4,3]
```

```
Out[41]: np.int64(19)
```

```
In [42]: MATR1[3,3]
```

```
Out[42]: np.int64(15)
```

```
In [43]: MATR1[-3,-1]
```

```
Out[43]: np.int64(11)
```

```
In [44]: mydata
```

```
Out[44]: array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
   17, 18, 19])
```

```
In [47]: MATR2=np.reshape(mydata,(5,4),order='F')
MATR2
```

```
Out[47]: array([[ 0,  5, 10, 15],
   [ 1,  6, 11, 16],
   [ 2,  7, 12, 17],
   [ 3,  8, 13, 18],
   [ 4,  9, 14, 19]])
```

```
In [48]: MATR2[4,3]
```

```
Out[48]: np.int64(19)
```

```
In [49]: MATR2[0,2]
```

```
Out[49]: np.int64(10)
```

```
In [50]: MATR2[-2,-4]
```

```
Out[50]: np.int64(3)
```

```
In [51]: MATR2[0:4]
```

```
Out[51]: array([[ 0,  5, 10, 15],  
                 [ 1,  6, 11, 16],  
                 [ 2,  7, 12, 17],  
                 [ 3,  8, 13, 18]])
```

```
In [52]: MATR2[3:4]
```

```
Out[52]: array([[ 3,  8, 13, 18]])
```

```
In [53]: MATR2[0:2]
```

```
Out[53]: array([[ 0,  5, 10, 15],  
                 [ 1,  6, 11, 16]])
```

```
In [54]: MATR2[1:2]
```

```
Out[54]: array([[ 1,  6, 11, 16]])
```

```
In [55]: MATR2[1,2]
```

```
Out[55]: np.int64(11)
```

```
In [56]: MATR2[-2,-1]
```

```
Out[56]: np.int64(18)
```

```
In [57]: MATR2[-3,-3]
```

```
Out[57]: np.int64(7)
```

```
In [58]: MATR2
```

```
Out[58]: array([[ 0,  5, 10, 15],  
                 [ 1,  6, 11, 16],  
                 [ 2,  7, 12, 17],  
                 [ 3,  8, 13, 18],  
                 [ 4,  9, 14, 19]])
```

```
In [59]: MATR2[4]
```

```
Out[59]: array([ 4,  9, 14, 19])
```

```
In [60]: mydata
```

```
Out[60]: array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,  
                 17, 18, 19])
```

```
In [61]: matr3=mydata.reshape(5,4,order='A')  
matr3
```

```
Out[61]: array([[ 0,  1,  2,  3],
   [ 4,  5,  6,  7],
   [ 8,  9, 10, 11],
   [12, 13, 14, 15],
   [16, 17, 18, 19]])
```

```
In [62]: MATR1
```

```
Out[62]: array([[ 0,  1,  2,  3],
   [ 4,  5,  6,  7],
   [ 8,  9, 10, 11],
   [12, 13, 14, 15],
   [16, 17, 18, 19]])
```

```
In [63]: MATR2
```

```
Out[63]: array([[ 0,  5, 10, 15],
   [ 1,  6, 11, 16],
   [ 2,  7, 12, 17],
   [ 3,  8, 13, 18],
   [ 4,  9, 14, 19]])
```

```
In [64]: a1=['welcome','to','datascience']
a2=['required','hard','work']
a3=[1,2,3]
```

```
In [65]: [a1,a2,a3]
```

```
Out[65]: [['welcome', 'to', 'datascience'], ['required', 'hard', 'work'], [1, 2, 3]]
```

```
In [66]: np.array([1,2,3])
```

```
Out[66]: array([1, 2, 3])
```

```
In [67]: Games
```

```
Out[67]: array([[80, 77, 82, 82, 73, 82, 58, 78,  6, 35],
   [82, 57, 82, 79, 76, 72, 60, 72, 79, 80],
   [79, 78, 75, 81, 76, 79, 62, 76, 77, 69],
   [80, 65, 77, 66, 69, 77, 55, 67, 77, 40],
   [82, 82, 82, 79, 82, 78, 54, 76, 71, 41],
   [70, 69, 67, 77, 70, 77, 57, 74, 79, 44],
   [78, 64, 80, 78, 45, 80, 60, 70, 62, 82],
   [35, 35, 80, 74, 82, 78, 66, 81, 81, 27],
   [40, 40, 40, 81, 78, 81, 39,  0, 10, 51],
   [75, 51, 51, 79, 77, 76, 49, 69, 54, 62]])
```

```
In [68]: Games[0]
```

```
Out[68]: array([80, 77, 82, 82, 73, 82, 58, 78,  6, 35])
```

```
In [69]: Games[0:5]
```

```
Out[69]: array([[80, 77, 82, 82, 73, 82, 58, 78,  6, 35],
   [82, 57, 82, 79, 76, 72, 60, 72, 79, 80],
   [79, 78, 75, 81, 76, 79, 62, 76, 77, 69],
   [80, 65, 77, 66, 69, 77, 55, 67, 77, 40],
   [82, 82, 82, 79, 82, 78, 54, 76, 71, 41]])
```

```
In [70]: Games[0,5]
```

```
Out[70]: np.int64(82)
```

```
In [71]: Games[0,2]
```

```
Out[71]: np.int64(82)
```

```
In [72]: Games
```

```
Out[72]: array([[80, 77, 82, 82, 73, 82, 58, 78, 6, 35],  
                 [82, 57, 82, 79, 76, 72, 60, 72, 79, 80],  
                 [79, 78, 75, 81, 76, 79, 62, 76, 77, 69],  
                 [80, 65, 77, 66, 69, 77, 55, 67, 77, 40],  
                 [82, 82, 82, 79, 82, 78, 54, 76, 71, 41],  
                 [70, 69, 67, 77, 70, 77, 57, 74, 79, 44],  
                 [78, 64, 80, 78, 45, 80, 60, 70, 62, 82],  
                 [35, 35, 80, 74, 82, 78, 66, 81, 81, 27],  
                 [40, 40, 40, 81, 78, 81, 39, 0, 10, 51],  
                 [75, 51, 51, 79, 77, 76, 49, 69, 54, 62]])
```

```
In [73]: Games[0:2]
```

```
Out[73]: array([[80, 77, 82, 82, 73, 82, 58, 78, 6, 35],  
                 [82, 57, 82, 79, 76, 72, 60, 72, 79, 80]])
```

```
In [74]: Games[-3:-1]
```

```
Out[74]: array([[35, 35, 80, 74, 82, 78, 66, 81, 81, 27],  
                 [40, 40, 40, 81, 78, 81, 39, 0, 10, 51]])
```

```
In [75]: Games[-3,-1]
```

```
Out[75]: np.int64(27)
```

```
In [76]: Points
```

```
Out[76]: array([[2832, 2430, 2323, 2201, 1970, 2078, 1616, 2133, 83, 782],  
                 [1653, 1426, 1779, 1688, 1619, 1312, 1129, 1170, 1245, 1154],  
                 [2478, 2132, 2250, 2304, 2258, 2111, 1683, 2036, 2089, 1743],  
                 [2122, 1881, 1978, 1504, 1943, 1970, 1245, 1920, 2112, 966],  
                 [1292, 1443, 1695, 1624, 1503, 1784, 1113, 1296, 1297, 646],  
                 [1572, 1561, 1496, 1746, 1678, 1438, 1025, 1232, 1281, 928],  
                 [1258, 1104, 1684, 1781, 841, 1268, 1189, 1186, 1185, 1564],  
                 [903, 903, 1624, 1871, 2472, 2161, 1850, 2280, 2593, 686],  
                 [597, 597, 597, 1361, 1619, 2026, 852, 0, 159, 904],  
                 [2040, 1397, 1254, 2386, 2045, 1941, 1082, 1463, 1028, 1331]])
```

```
In [77]: Points[0]
```

```
Out[77]: array([2832, 2430, 2323, 2201, 1970, 2078, 1616, 2133, 83, 782])
```

```
In [78]: Points[6,1]
```

```
Out[78]: np.int64(1104)
```

```
In [79]: Points[3:6]
```

```
Out[79]: array([[2122, 1881, 1978, 1504, 1943, 1970, 1245, 1920, 2112, 966],  
                 [1292, 1443, 1695, 1624, 1503, 1784, 1113, 1296, 1297, 646],  
                 [1572, 1561, 1496, 1746, 1678, 1438, 1025, 1232, 1281, 928]])
```

```
In [80]: Points[-6,-1]
```

```
Out[80]: np.int64(646)
```

```
In [81]: dict1={'key1':'val1','key2':'val2','key3':'val3'}
```

```
In [82]: dict1
```

```
Out[82]: {'key1': 'val1', 'key2': 'val2', 'key3': 'val3'}
```

```
In [83]: dict1['key2']
```

```
Out[83]: 'val2'
```

```
In [84]: dict2={'bang':2,'hyd':'we are hear','pune':True}
```

```
In [85]: dict2
```

```
Out[85]: {'bang': 2, 'hyd': 'we are hear', 'pune': True}
```

```
In [86]: dict3={'Germany':'I have been here','France':2,'Spain':True}
```

```
In [87]: dict3
```

```
Out[87]: {'Germany': 'I have been here', 'France': 2, 'Spain': True}
```

```
In [89]: dict3['Germany']
```

```
Out[89]: 'I have been here'
```

```
In [90]: Games
```

```
Out[90]: array([[80, 77, 82, 82, 73, 82, 58, 78, 6, 35],  
                 [82, 57, 82, 79, 76, 72, 60, 72, 79, 80],  
                 [79, 78, 75, 81, 76, 79, 62, 76, 77, 69],  
                 [80, 65, 77, 66, 69, 77, 55, 67, 77, 40],  
                 [82, 82, 82, 79, 82, 78, 54, 76, 71, 41],  
                 [70, 69, 67, 77, 70, 77, 57, 74, 79, 44],  
                 [78, 64, 80, 78, 45, 80, 60, 70, 62, 82],  
                 [35, 35, 80, 74, 82, 78, 66, 81, 81, 27],  
                 [40, 40, 40, 81, 78, 81, 39, 0, 10, 51],  
                 [75, 51, 51, 79, 77, 76, 49, 69, 54, 62]])
```

```
In [10]: Sdict
```

```
Out[10]: {'2015': 0,
          '2016': 1,
          '2017': 2,
          '2018': 3,
          '2019': 4,
          '2020': 5,
          '2021': 6,
          '2022': 7,
          '2023': 8,
          '2024': 9}
```

```
In [11]: Pdict
```

```
Out[11]: {'Sachin': 0,
           'Rahul': 1,
           'Smith': 2,
           'Sami': 3,
           'Pollard': 4,
           'Morris': 5,
           'Samson': 6,
           'Dhoni': 7,
           'Kohli': 8,
           'Sky': 9}
```

```
In [91]: Pdict['Sachin']
```

```
Out[91]: 0
```

Games

```
In [93]: Games[Pdict['Rahul']]
```

```
Out[93]: array([82, 57, 82, 79, 76, 72, 60, 72, 79, 80])
```

```
In [12]: Games[5]
```

```
Out[12]: array([70, 69, 67, 77, 70, 77, 57, 74, 79, 44])
```

```
In [13]: Games[[5,3]]
```

```
Out[13]: array([[70, 69, 67, 77, 70, 77, 57, 74, 79, 44],
                 [80, 65, 77, 66, 69, 77, 55, 67, 77, 40]])
```

```
In [14]: Salary[0]
```

```
Out[14]: array([15946875, 17718750, 19490625, 21262500, 23034375, 24806250,
                25244493, 27849149, 30453805, 23500000])
```

```
In [15]: Games[0]
```

```
Out[15]: array([80, 77, 82, 82, 73, 82, 58, 78, 6, 35])
```

```
In [16]: Salary[0]/Games[0]
```

```
Out[16]: array([ 199335.9375 , 230113.63636364, 237690.54878049,
                 259298.7804878 , 315539.38356164, 302515.24390244,
                 435249.87931034, 357040.37179487, 5075634.16666667,
                 671428.57142857])
```

```
In [94]: Salary[2,4]
```

```
Out[94]: np.int64(15779912)
```

```
In [95]: Salary[Pdict['Sky']][Sdict['2019']]
```

```
Out[95]: np.int64(15779912)
```

```
In [96]: Salary/Games
```

```
Out[96]: array([[ 199335.9375 , 230113.63636364, 237690.54878049,
   259298.7804878 , 315539.38356164, 302515.24390244,
   435249.87931034, 357040.37179487, 5075634.16666667,
   671428.57142857],
 [ 146341.46341463, 223582.26315789, 164492.40243902,
  180159.07594937, 197062.55263158, 226729.16666667,
  300642.88333333, 274342.29166667, 271730.60759494,
  289759.875 ],
 [ 58503.79746835, 74719.1025641 , 173883.33333333,
  177908.40740741, 207630.42105263, 183544.30379747,
  258427.41935484, 230855.26315789, 247629.87012987,
  299194.20289855],
 [ 46420.5 , 72216.01538462, 169366.88311688,
  218342.13636364, 228694.37681159, 222717.44155844,
  336701.34545455, 290298.50746269, 291006.15584416,
  561450. ],
 [ 54794.63414634, 58618.53658537, 73917.97560976,
  174151.89873418, 185397.43902439, 213425.38461538,
  335032.77777778, 257057.36842105, 288918. ,
  522835.87804878],
 [ 47828.57142857, 61380. , 185895.52238806,
  187150.4025974 , 225427.31428571, 188311.68831169,
  281096.49122807, 237094.59459459, 241360.75949367,
  469190.90909091],
 [ 40310.76923077, 52815. , 45199.5 ,
  58643.44871795, 300455.55555556, 186751.9125 ,
  272663.41666667, 253992.25714286, 301103.72580645,
  244738.57317073],
 [ 0. , 0. , 52140. ,
  60595.13513514, 58498.53658537, 77611.06410256,
  234948.96969697, 205797.90123457, 220155.88888889,
  703541.62962963],
 [ 0. , 0. , 0. ,
  59540.74074074, 66467.69230769, 68471.11111111,
  179325.84615385, inf, 1763268.8 ,
  369860.29411765],
 [ 40425.6 , 75322.41176471, 255710.78431373,
  182412.41772152, 204933.92207792, 186842.10526316,
  320224.48979592, 249014.49275362, 345796.2962963 ,
  241935.48387097]])
```

```
In [17]: np.round(Salary[0]/Games[0])
```

```
Out[17]: array([ 199336., 230114., 237691., 259299., 315539., 302515.,
 435250., 357040., 5075634., 671429.])
```

visualie the data

```
In [18]: import warnings  
warnings.filterwarnings('ignore')  
  
#to ignore unwanted errors from we should write the code as ignore all
```

```
In [19]: import matplotlib.pyplot as plt
```

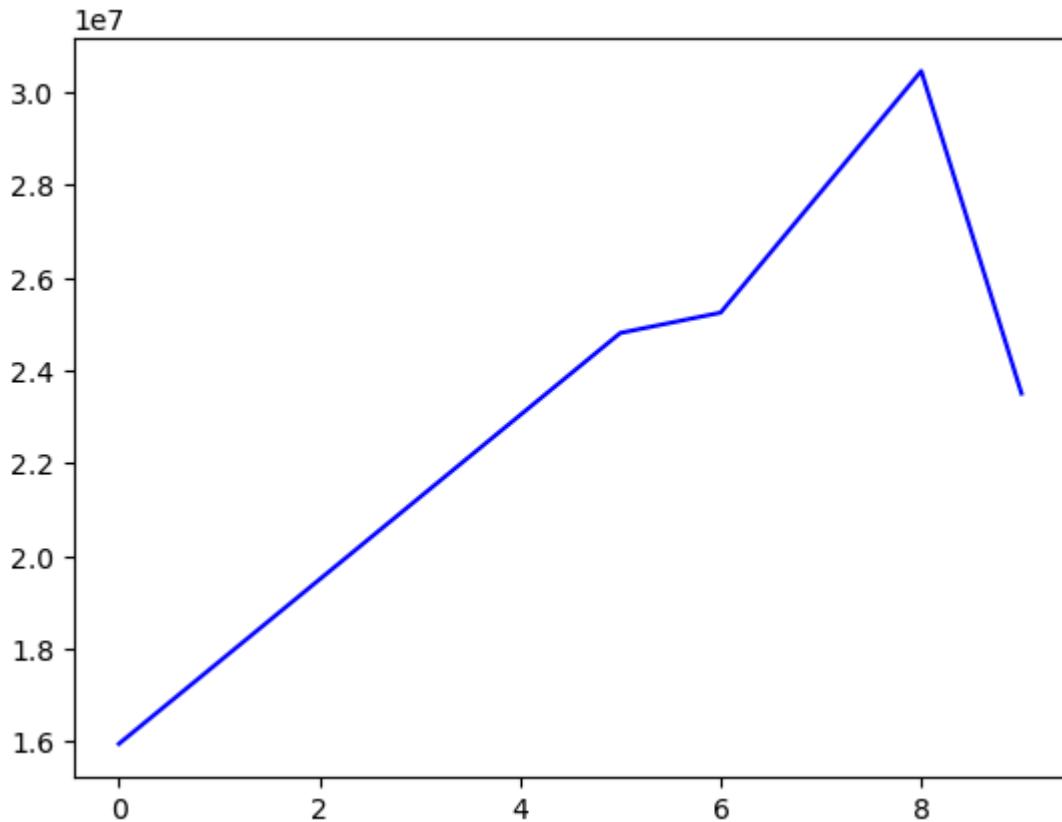
```
In [20]: Salary[0]
```

```
Out[20]: array([15946875, 17718750, 19490625, 21262500, 23034375, 24806250,  
25244493, 27849149, 30453805, 23500000])
```

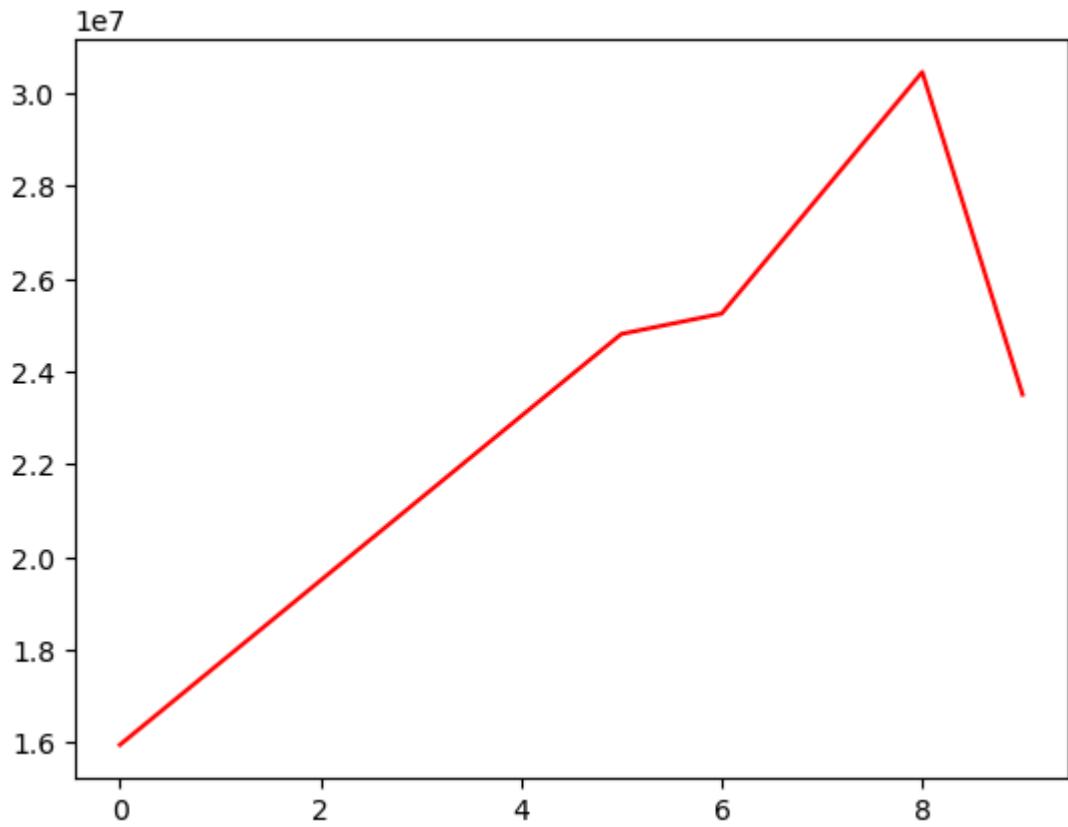
```
In [97]: plt.plot(Salary[0])
```

```
Out[97]: [
```

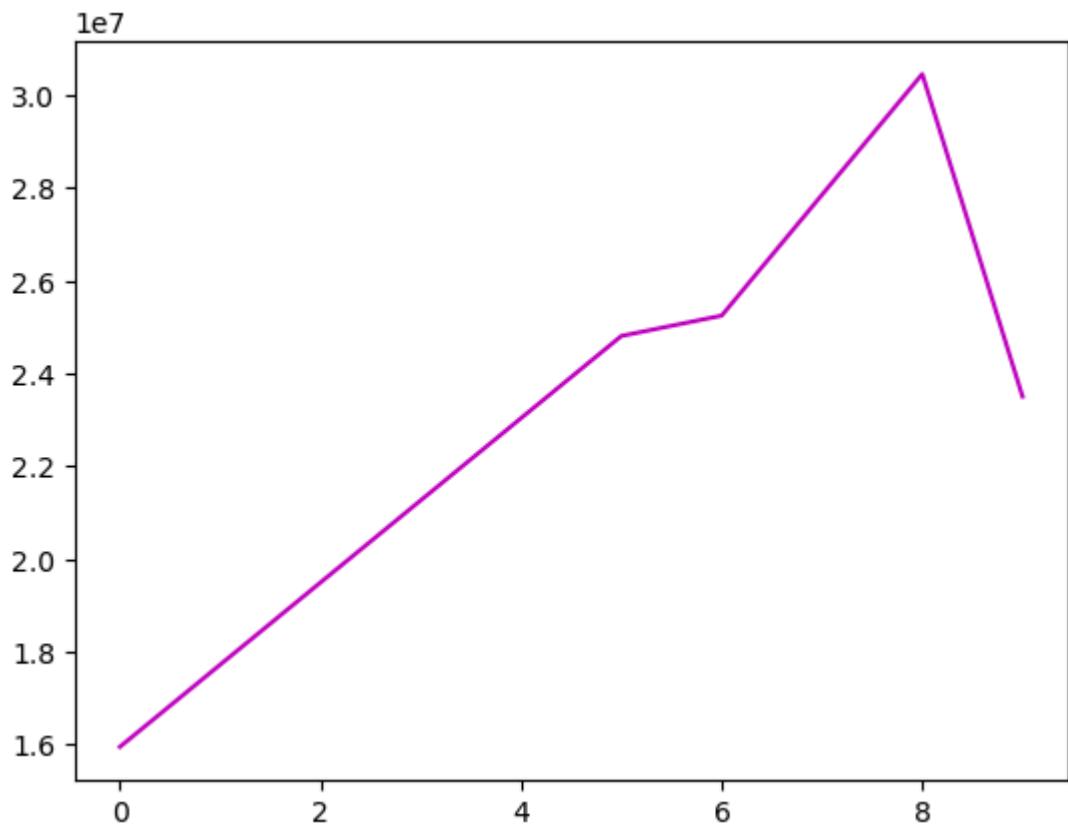
```
In [21]: plt.plot(Salary[0], color="blue")  
plt.show()
```



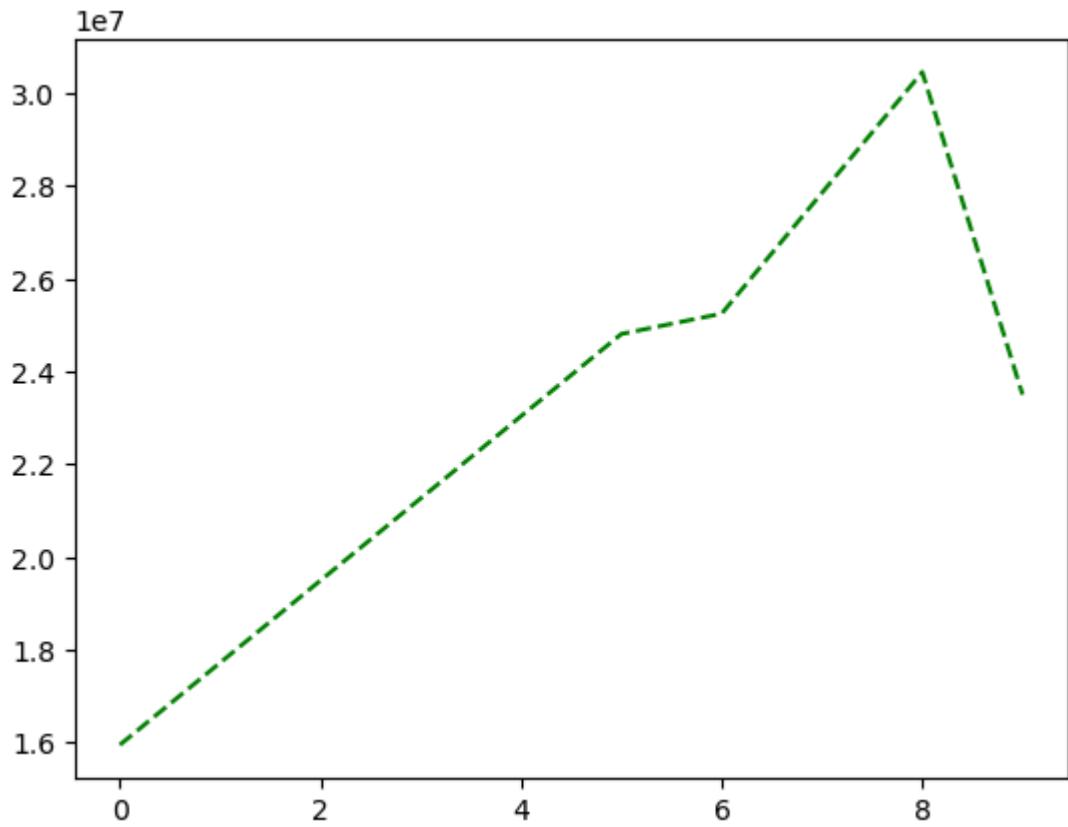
```
In [22]: plt.plot(Salary[0], c="r")  
plt.show()
```



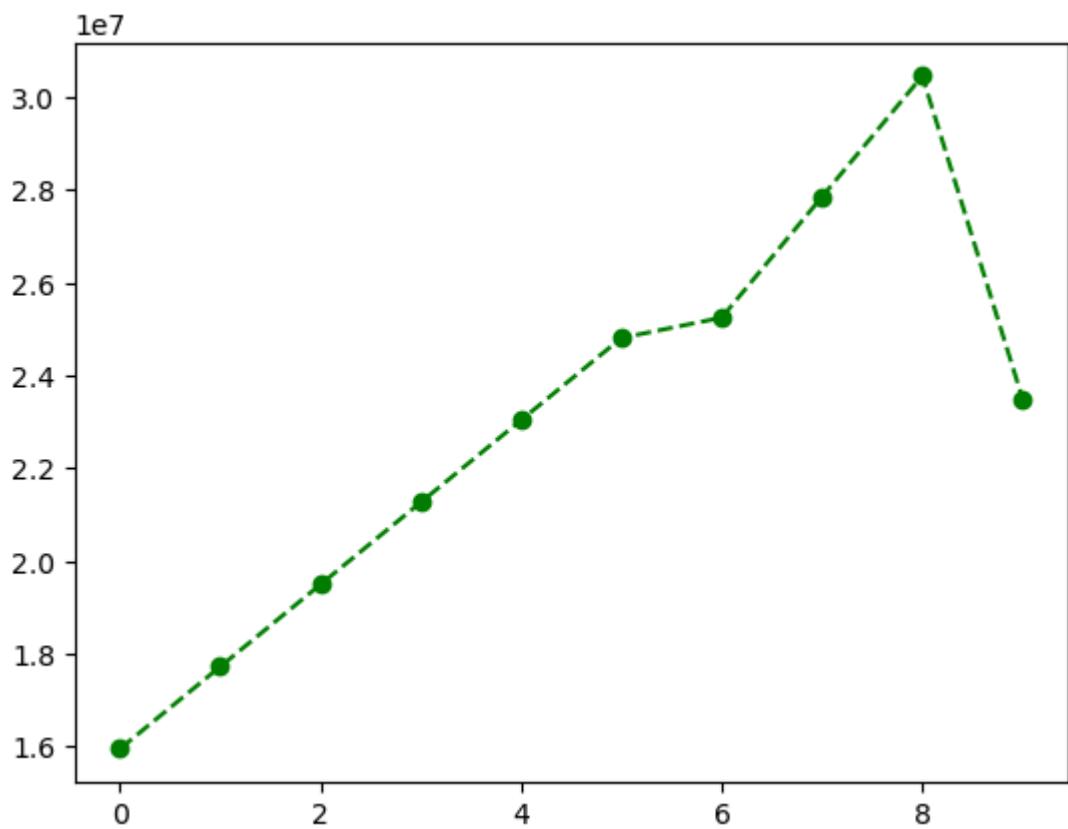
```
In [23]: plt.plot(Salary[0], c="m")
plt.show()
```



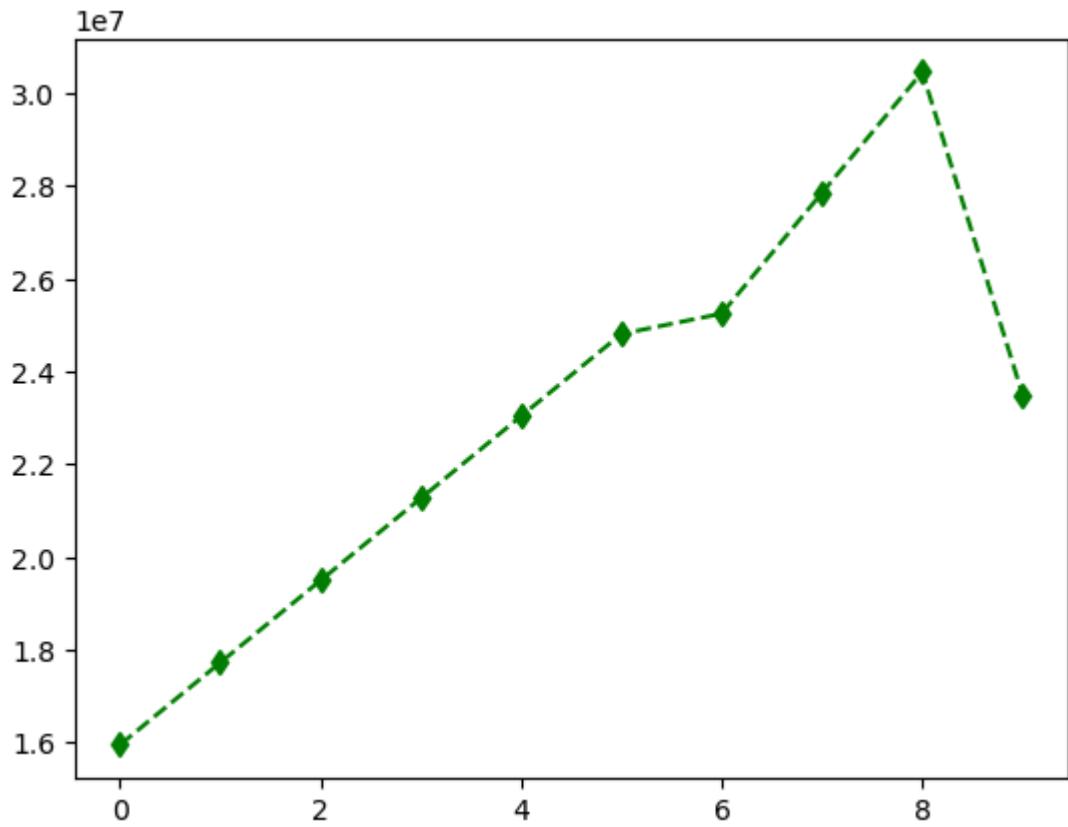
```
In [24]: plt.plot(Salary[0], c="g", ls='--')
plt.show()
```



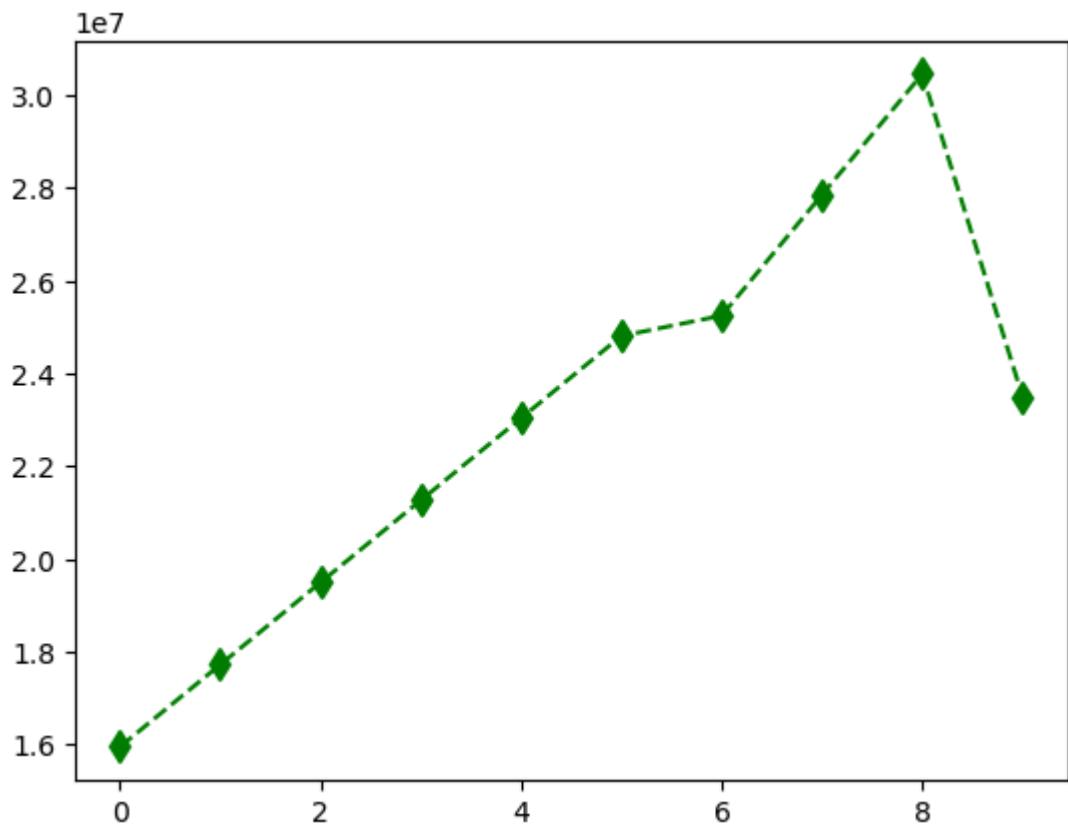
```
In [25]: plt.plot(Salary[0], c="g", ls='--', marker='o')
plt.show()
```



```
In [26]: plt.plot(Salary[0], c="g", ls='--', marker='d')
plt.show()
```



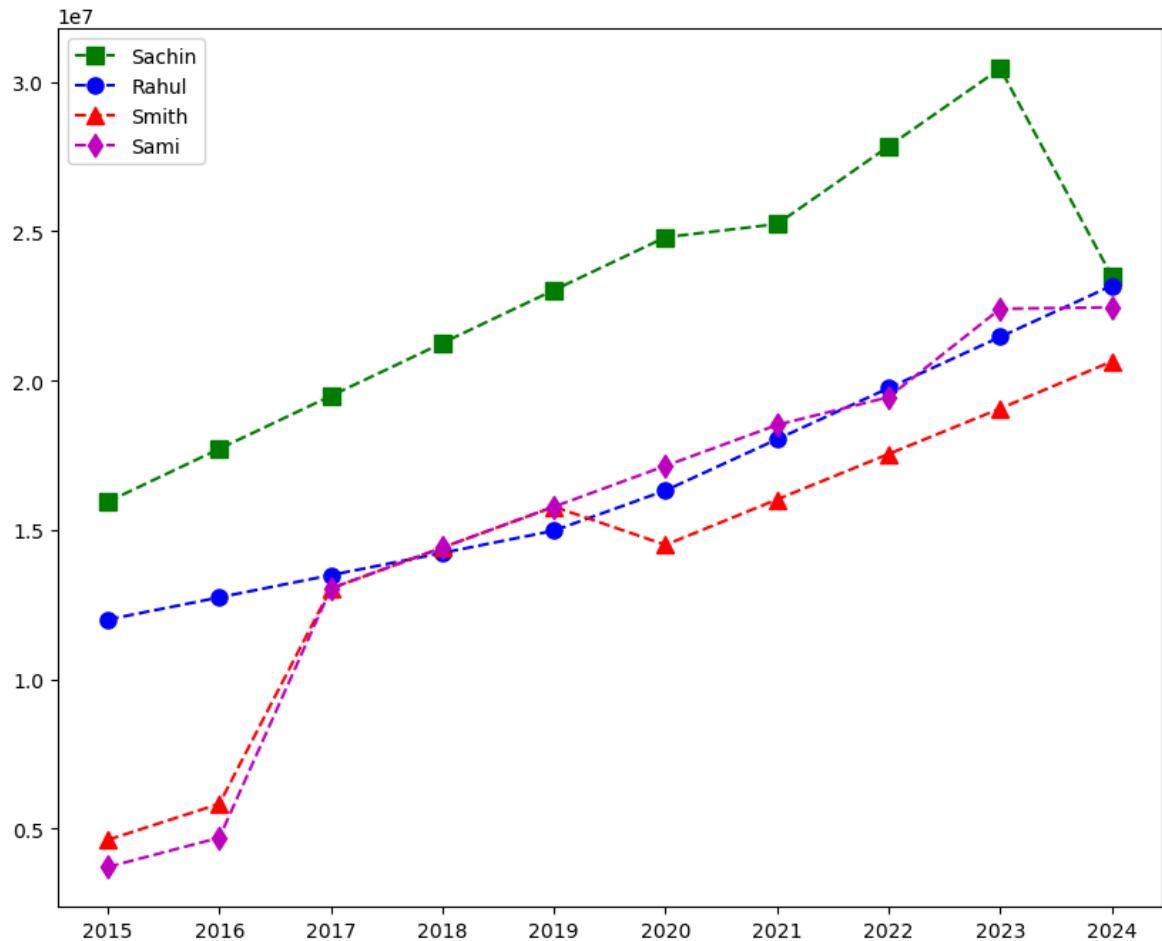
```
In [27]: plt.plot(Salary[0], c="g", ls='--', marker='d', ms=8)  
plt.show()
```



```
In [99]: %matplotlib inline  
plt.rcParams['figure.figsize']=10,8
```

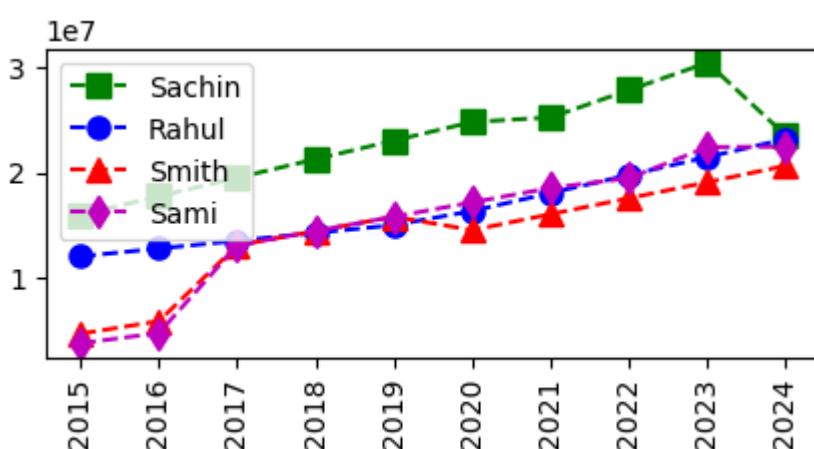
```
In [100]:
```

```
plt.plot(Salary[0], c="g",ls='--',marker='s',ms=8,label=Players[0])
plt.plot(Salary[1], c="b",ls='--',marker='o',ms=8,label=Players[1])
plt.plot(Salary[2], c="r",ls='--',marker='^',ms=8,label=Players[2])
plt.plot(Salary[3], c="m",ls='--',marker='d',ms=8,label=Players[3])
plt.xticks(list(range(0,10)),Seasons)
plt.legend()
plt.show()
```



```
In [30]:
```

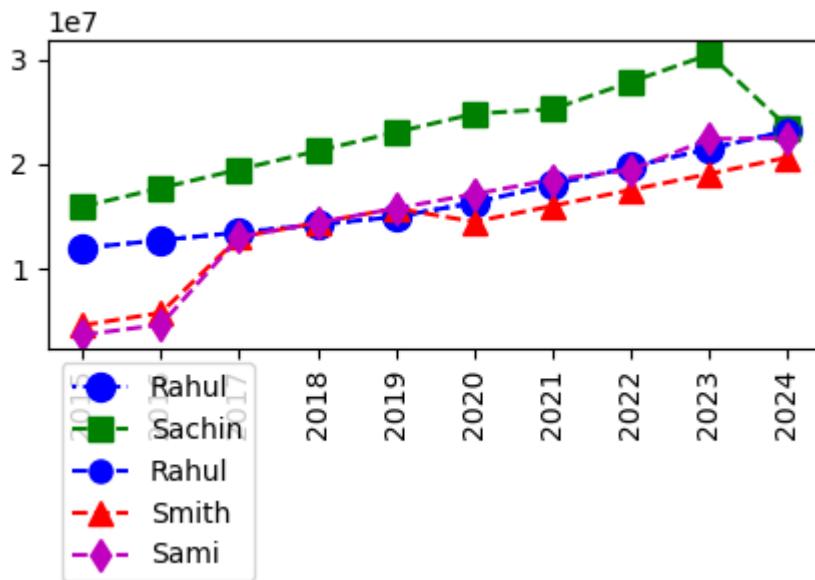
```
plt.plot(Salary[0], c="g",ls='--',marker='s',ms=8,label=Players[0])
plt.plot(Salary[1], c="b",ls='--',marker='o',ms=8,label=Players[1])
plt.plot(Salary[2], c="r",ls='--',marker='^',ms=8,label=Players[2])
plt.plot(Salary[3], c="m",ls='--',marker='d',ms=8,label=Players[3])
plt.xticks(list(range(0,10)),Seasons,rotation='vertical')
plt.legend()
plt.show()
```



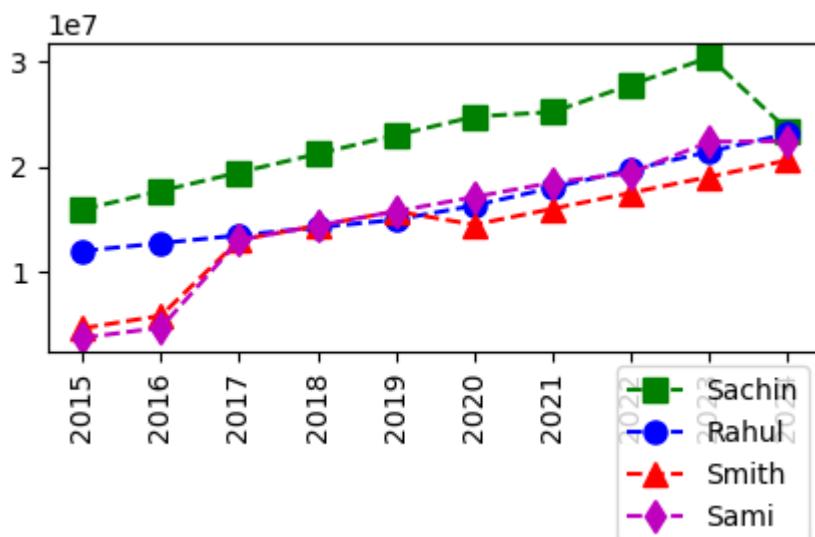
```
In [31]: plt.plot(Salary[1], c="b",ls=':',marker='o',ms=10,label=Players[1])
```

```
Out[31]: [
```

```
In [32]: plt.plot(Salary[0], c="g",ls='--',marker='s',ms=8,label=Players[0])
plt.plot(Salary[1], c="b",ls='--',marker='o',ms=8,label=Players[1])
plt.plot(Salary[2], c="r",ls='--',marker='^',ms=8,label=Players[2])
plt.plot(Salary[3], c="m",ls='--',marker='d',ms=8,label=Players[3])
plt.xticks(list(range(0,10)),Seasons,rotation='vertical')
plt.legend(loc='upper left',bbox_to_anchor=(0,0))
plt.show()
```

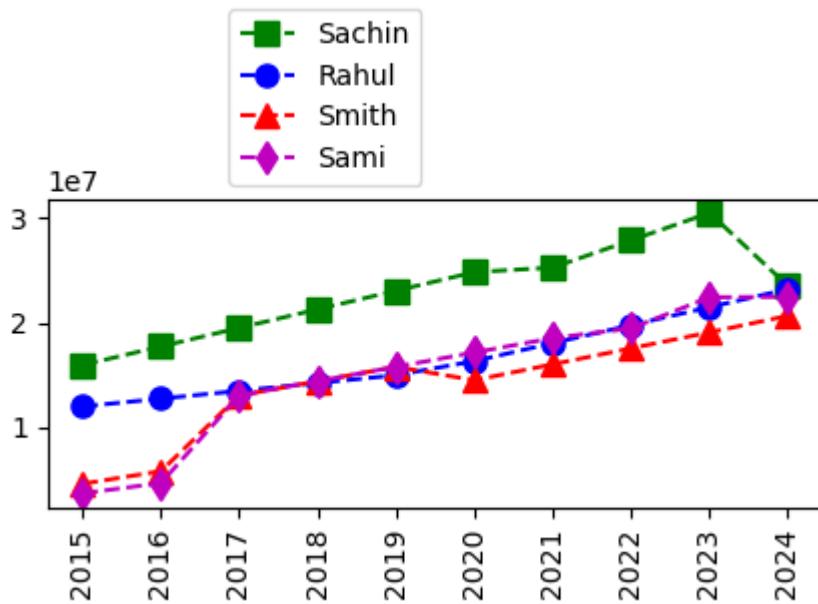


```
In [33]: plt.plot(Salary[0], c="g",ls='--',marker='s',ms=8,label=Players[0])
plt.plot(Salary[1], c="b",ls='--',marker='o',ms=8,label=Players[1])
plt.plot(Salary[2], c="r",ls='--',marker='^',ms=8,label=Players[2])
plt.plot(Salary[3], c="m",ls='--',marker='d',ms=8,label=Players[3])
plt.xticks(list(range(0,10)),Seasons,rotation='vertical')
plt.legend(loc='upper right',bbox_to_anchor=(1,0))
plt.show()
```

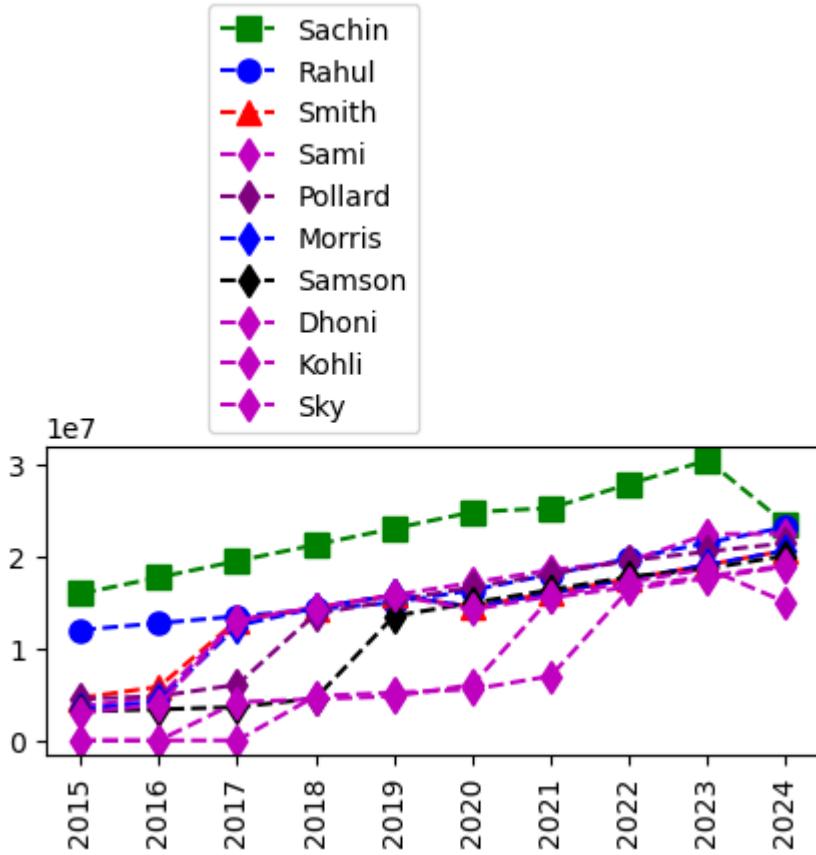


```
In [34]: plt.plot(Salary[0], c="g",ls='--',marker='s',ms=8,label=Players[0])
plt.plot(Salary[1], c="b",ls='--',marker='o',ms=8,label=Players[1])
plt.plot(Salary[2], c="r",ls='--',marker='^',ms=8,label=Players[2])
```

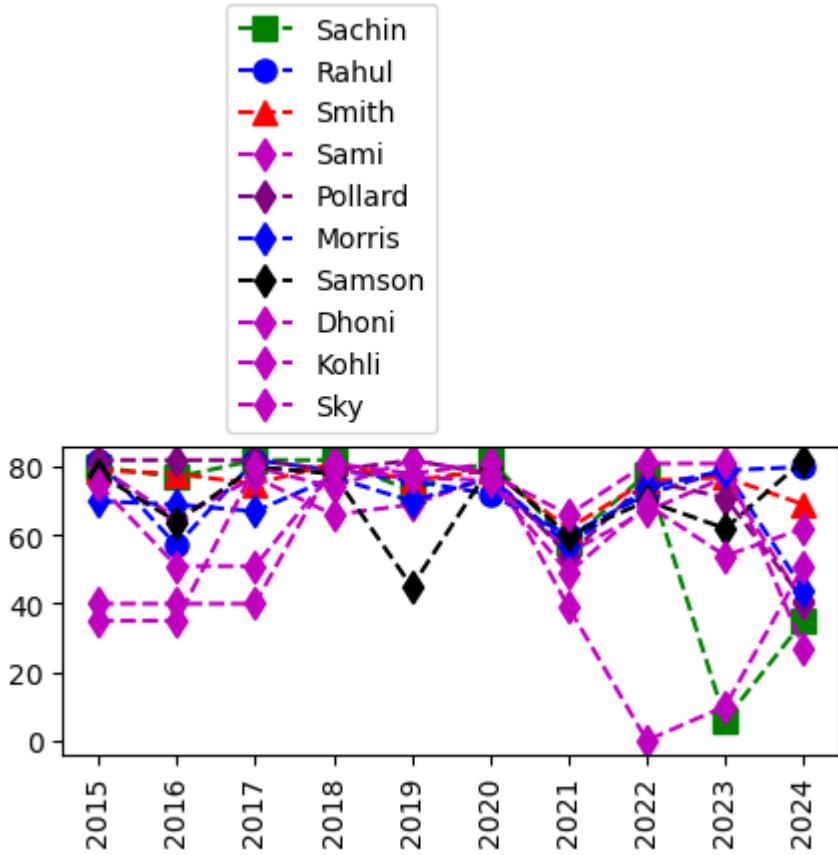
```
plt.plot(Salary[3], c="m",ls='--',marker='d',ms=8,label=Players[3])
plt.xticks(list(range(0,10)),Seasons,rotation='vertical')
plt.legend(loc='lower right',bbox_to_anchor=(0.5,1))
plt.show()
```



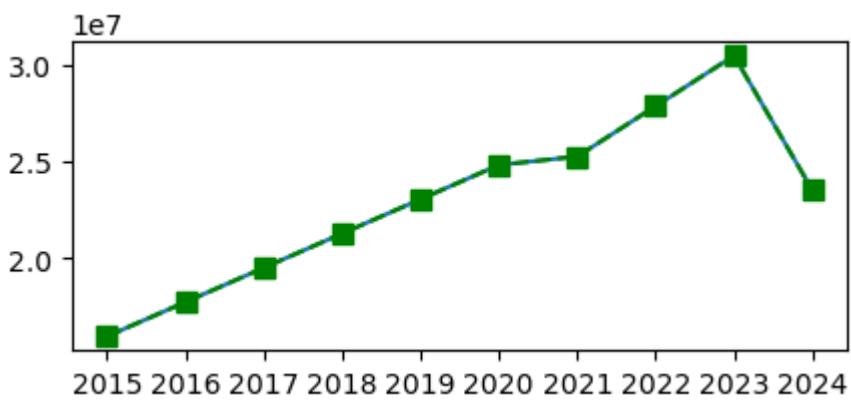
```
In [35]: plt.plot(Salary[0], c="g",ls='--',marker='s',ms=8,label=Players[0])
plt.plot(Salary[1], c="b",ls='--',marker='o',ms=8,label=Players[1])
plt.plot(Salary[2], c="r",ls='--',marker='^',ms=8,label=Players[2])
plt.plot(Salary[3], c="m",ls='--',marker='d',ms=8,label=Players[3])
plt.plot(Salary[4], c="Purple",ls='--',marker='d',ms=8,label=Players[4])
plt.plot(Salary[5], c="b",ls='--',marker='d',ms=8,label=Players[5])
plt.plot(Salary[6], c="Black",ls='--',marker='d',ms=8,label=Players[6])
plt.plot(Salary[7], c="m",ls='--',marker='d',ms=8,label=Players[7])
plt.plot(Salary[8], c="m",ls='--',marker='d',ms=8,label=Players[8])
plt.plot(Salary[9], c="m",ls='--',marker='d',ms=8,label=Players[9])
plt.xticks(list(range(0,10)),Seasons,rotation='vertical')
plt.legend(loc='lower right',bbox_to_anchor=(0.5,1))
plt.show()
```



```
In [36]: plt.plot(Games[0], c="g",ls='--',marker='s',ms=8,label=Players[0])
plt.plot(Games[1], c="b",ls='--',marker='o',ms=8,label=Players[1])
plt.plot(Games[2], c="r",ls='--',marker='^',ms=8,label=Players[2])
plt.plot(Games[3], c="m",ls='--',marker='d',ms=8,label=Players[3])
plt.plot(Games[4], c="Purple",ls='--',marker='d',ms=8,label=Players[4])
plt.plot(Games[5], c="b",ls='--',marker='d',ms=8,label=Players[5])
plt.plot(Games[6], c="Black",ls='--',marker='d',ms=8,label=Players[6])
plt.plot(Games[7], c="m",ls='--',marker='d',ms=8,label=Players[7])
plt.plot(Games[8], c="m",ls='--',marker='d',ms=8,label=Players[8])
plt.plot(Games[9], c="m",ls='--',marker='d',ms=8,label=Players[9])
plt.xticks(list(range(0,10)),Seasons,rotation='vertical')
plt.legend(loc='lower right',bbox_to_anchor=(0.5,1))
plt.show()
```



```
In [98]: plt.plot(Salary[0], c="g",ls='--',marker='s',ms=7)
plt.xticks(list(range(0,10)),Seasons,rotation='horizontal')
plt.show()
```



```
In [ ]:
```