

```

In [6]: #Import numpy
import numpy as np

#Seasons
Seasons = ["2010", "2011", "2012", "2013", "2014", "2015", "2016", "2017", "2018", "2019"] # data in list
Sdict = {"2010":0, "2011":1, "2012":2, "2013":3, "2014":4, "2015":5, "2016":6, "2017":7, "2018":8, "2019":9} # dictionary

#Players
Players = ["Sachin", "Rahul", "Smith", "Sami", "Pollard", "Morris", "Samson", "Dhoni", "Kohli", "Sky"]
Pdict = {"Sachin":0, "Rahul":1, "Smith":2, "Sami":3, "Pollard":4, "Morris":5, "Samson":6, "Dhoni":7, "Kohli":8, "Sky":9}

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

#Matrix
Salary = np.array([Sachin_Salary, Rahul_Salary, Smith_Salary, Sami_Salary, Pollard_Salary, Morris_Salary, Samson_Salary, Dhoni_Salary, Kohli_Salary, Sky_Salary])

#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, Samson_G, Dhoni_G, Kohli_G, Sky_G])

#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, Morris_PTS, Samson_PTS, Dhoni_PTS, Kohli_

```

```

In [19]: # print(Salary)
print("Players Salarys",Salary)

```

```

Players Salarys [[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 [17]: print("Games = \n",Games)

```

```
Games =
[[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 [9]: print(Points)
```

```
[[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 [25]: my_data = np.arange(0,20)
         my_data
```

```
Out[25]: array([ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9, 10, 11, 12, 13, 14, 15, 16,
                17, 18, 19])
```

```
In [29]: # Reshape and store in the same variable
         my_data = np.reshape(my_data,(2,10))
         print(my_data)
```

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

```
In [37]: my_data = np.reshape(my_data,(4,5))
         my_data
         # bydefault matrix has 'C' order structure
```

```
Out[37]: array([[ 0,  1,  2,  3,  4],
                [ 5,  6,  7,  8,  9],
                [10, 11, 12, 13, 14],
                [15, 16, 17, 18, 19]])
```

```
In [27]: mat_1 = np.reshape(my_data,(5,4))
mat_1
```

```
Out[27]: array([[ 0,  1,  2,  3],
                [ 4,  5,  6,  7],
                [ 8,  9, 10, 11],
                [12, 13, 14, 15],
                [16, 17, 18, 19]])
```

```
In [39]: MAT_1 = np.reshape(my_data,(5,4),order = 'C')
```

```
In [41]: MAT_1
```

```
Out[41]: array([[ 0,  1,  2,  3],
                [ 4,  5,  6,  7],
                [ 8,  9, 10, 11],
                [12, 13, 14, 15],
                [16, 17, 18, 19]])
```

```
In [59]: MAT_2 = np.reshape(MAT_1,(5,4),order = 'A')
print(MAT_1,'\n\n\n',MAT_2)
MAT_3 = np.reshape(MAT_2,(5,4),order = 'F')
```

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

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

```
In [61]: MAT_1
```

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

```
In [65]: MAT_1[-4:-2]
```

```
Out[65]: array([[ 4,  5,  6,  7],
                [ 8,  9, 10, 11]])
```

```
In [71]: MAT_1[-3,-3]
```

```
Out[71]: 9
```

```
In [77]: MAT_1 = np.reshape(MAT_1,(5,4), order = 'A')
MAT_1
```

```
Out[77]: array([[ 0,  1,  2,  3],
                [ 4,  5,  6,  7],
                [ 8,  9, 10, 11],
                [12, 13, 14, 15],
                [16, 17, 18, 19]])
```

```
In [79]: MAT_1
```

```
Out[79]: array([[ 0,  1,  2,  3],
                [ 4,  5,  6,  7],
                [ 8,  9, 10, 11],
                [12, 13, 14, 15],
                [16, 17, 18, 19]])
```

```
In [49]: Games
```

```
Out[49]: 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 [51]: Games[0] # 0th row
```

```
Out[51]: array([80, 77, 82, 82, 73, 82, 58, 78, 6, 35])
```

```
In [53]: Games[5] # 5th row
```

```
Out[53]: array([70, 69, 67, 77, 70, 77, 57, 74, 79, 44])
```

```
In [81]: Games[0:5] # slice from 0th row to 5-1 rows
```

```
Out[81]: 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 [83]: Games[0,5]
```

```
Out[83]: 82
```

```
In [85]: Games[2,6]
```

```
Out[85]: 62
```

```
In [61]: Points
```

```
Out[61]: 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 [63]: Games

```
Out[63]: 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 [65]: Pdict

```
Out[65]: {'Sachin': 0,
          'Rahul': 1,
          'Smith': 2,
          'Sami': 3,
          'Pollard': 4,
          'Morris': 5,
          'Samson': 6,
          'Dhoni': 7,
          'Kohli': 8,
          'Sky': 9}
```

```
In [105... # played ga,es by players
print(Games[0])
print("Games played by sachin from 2010 to 2019 :\n",Games[Pdict['Sachin']])
print()
# Smith -2
```

```
print(Games[2])
print("Games played by Smith from 2010 to 2019 :\n",Games[Pdict['Smith']])
print()
# Dhoni
print("Games played by Dhoni from 2010 to 2019 :\n",Games[Pdict['Dhoni']])
```

```
[80 77 82 82 73 82 58 78  6 35]
Games played by sachin from 2010 to 2019 :
[80 77 82 82 73 82 58 78  6 35]
```

```
[79 78 75 81 76 79 62 76 77 69]
Games played by Smith from 2010 to 2019 :
[79 78 75 81 76 79 62 76 77 69]
```

```
Games played by Dhoni from 2010 to 2019 :
[35 35 80 74 82 78 66 81 81 27]
```

In []:

In [73]: Points

```
Out[73]: 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 [116...

```
Points[0]
print("Sachins point from 2010 to 2019 =",Points[Pdict['Sachin']])
print("Smiths points frim 2010 to 2019 =",Points[Pdict['Smith']])
print("Dhonis points from 2010 to 2019 =",Points[Pdict['Dhoni']])
# Points[Pdict[
```

```
Sachins point from 2010 to 2019 = [2832 2430 2323 2201 1970 2078 1616 2133  83  782]
Smiths points frim 2010 to 2019 = [2478 2132 2250 2304 2258 2111 1683 2036 2089 1743]
Dhonis points from 2010 to 2019 = [ 903  903 1624 1871 2472 2161 1850 2280 2593  686]
```


In [75]: Salary

```
Out[75]: 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 [122... # printing salary from salary matrix
print(Salary[0]) # sachin salary
print(Salary[2]) # Smith Salary

print("Sachin Salary from 2010 to 2019 ",Salary[Pdict['Sachin']])
print()
print("Smith Salary from 2010 to 2019 ",Salary[Pdict['Smith']])
print()
print("Dhoni Salary from 2010 to 2019 ",Salary[Pdict['Dhoni']])
```

```
[15946875 17718750 19490625 21262500 23034375 24806250 25244493 27849149
 30453805 23500000]
[ 4621800  5828090 13041250 14410581 15779912 14500000 16022500 17545000
 19067500 20644400]
Sachin Salary from 2010 to 2019 [15946875 17718750 19490625 21262500 23034375 24806250 25244493 27849149
 30453805 23500000]

Smith Salary from 2010 to 2019 [ 4621800  5828090 13041250 14410581 15779912 14500000 16022500 17545000
 19067500 20644400]

Dhoni Salary from 2010 to 2019 [          0          0 4171200 4484040 4796880 6053663 15506632 16669630
 17832627 18995624]
```

In [95]: Sdict

Out[95]: {'2010': 0,
 '2011': 1,
 '2012': 2,
 '2013': 3,
 '2014': 4,
 '2015': 5,
 '2016': 6,
 '2017': 7,
 '2018': 8,
 '2019': 9}

In [130...

```
# printing salary in matrix (only one)

Salary[Pdict['Sachin']][Sdict['2010']]

Salary[Pdict['Dhoni']][Sdict['2015']]
Salary[0][0]

# dhoni salary in 2016
print("Dhoni Salary in 2016 = ",Salary[Pdict['Dhoni']][Sdict['2016']])
# for i in range(0,10):
#     print(Players[i],"Salary in ",Seasons[i],Salary[i][i])

print("Smith salary in 2019 is : ",Salary[Pdict['Smith']][Sdict['2019']])
```

Dhoni Salary in 2016 = 15506632
 Smith salary in 2019 is : 20644400

In [132...

Games

Out[132...

```
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 [134...

Points

Out[134...

```
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 [138...

Games/Points

```
C:\Users\Abhishek\AppData\Local\Temp\ipykernel_7632\1183005256.py:1: RuntimeWarning: invalid value encountered in divide
  Games/Points
```

```
Out[138... array([[0.02824859, 0.03168724, 0.03529918, 0.03725579, 0.03705584,
        0.03946102, 0.03589109, 0.03656821, 0.07228916, 0.04475703],
       [0.04960678, 0.03997195, 0.04609331, 0.04680095, 0.04694256,
        0.05487805, 0.05314438, 0.06153846, 0.06345382, 0.06932409],
       [0.03188055, 0.03658537, 0.03333333, 0.03515625, 0.0336581 ,
        0.03742302, 0.03683898, 0.03732809, 0.03685974, 0.03958692],
       [0.03770028, 0.03455609, 0.03892821, 0.04388298, 0.03551209,
        0.03908629, 0.04417671, 0.03489583, 0.03645833, 0.04140787],
       [0.06346749, 0.05682606, 0.04837758, 0.04864532, 0.05455755,
        0.04372197, 0.04851752, 0.05864198, 0.05474171, 0.06346749],
       [0.04452926, 0.04420243, 0.0447861 , 0.0441008 , 0.04171633,
        0.05354659, 0.05560976, 0.06006494, 0.06167057, 0.04741379],
       [0.06200318, 0.05797101, 0.04750594, 0.04379562, 0.05350773,
        0.06309148, 0.05046257, 0.05902192, 0.05232068, 0.05242967],
       [0.03875969, 0.03875969, 0.04926108, 0.03955104, 0.03317152,
        0.0360944 , 0.03567568, 0.03552632, 0.03123795, 0.0393586 ],
       [0.06700168, 0.06700168, 0.06700168, 0.05951506, 0.04817789,
        0.03998026, 0.04577465,          nan, 0.06289308, 0.05641593],
       [0.03676471, 0.0365068 , 0.04066986, 0.03310981, 0.03765281,
        0.03915507, 0.04528651, 0.04716336, 0.05252918, 0.04658152]])
```

```
In [142... np.round(Salary/Games)
```

C:\Users\Abhishek\AppData\Local\Temp\ipykernel_7632\3232172828.py:1: RuntimeWarning: divide by zero encountered in divide

```
np.round(Salary/Games)
```

```
Out[142...] array([[ 199336.,  230114.,  237691.,  259299.,  315539.,  302515.,
         435250.,  357040.,  5075634.,  671429.],
       [ 146341.,  223582.,  164492.,  180159.,  197063.,  226729.,
        300643.,  274342.,  271731.,  289760.],
       [  58504.,   74719.,  173883.,  177908.,  207630.,  183544.,
        258427.,  230855.,  247630.,  299194.],
       [  46420.,   72216.,  169367.,  218342.,  228694.,  222717.,
        336701.,  290299.,  291006.,  561450.],
       [  54795.,   58619.,   73918.,  174152.,  185397.,  213425.,
        335033.,  257057.,  288918.,  522836.],
       [  47829.,   61380.,  185896.,  187150.,  225427.,  188312.,
        281096.,  237095.,  241361.,  469191.],
       [  40311.,   52815.,   45200.,   58643.,  300456.,  186752.,
        272663.,  253992.,  301104.,  244739.],
       [     0.,     0.,   52140.,   60595.,   58499.,   77611.,
        234949.,  205798.,  220156.,  703542.],
       [     0.,     0.,     0.,   59541.,   66468.,   68471.,
        179326.,    inf,  1763269.,  369860.],
       [  40426.,   75322.,  255711.,  182412.,  204934.,  186842.,
        320224.,  249014.,  345796.,  241935.]])
```

ignore warninngs

```
In [144...] # we can avoid warning by warnings
import warnings
warnings.filterwarnings('ignore')
```

```
In [150...] # np.round(Salary/Points) # no warnings
```

import numpy as np

import matplotlib.pyplot as plt

%matplotlib inline

```
In [3]: import numpy as np
import matplotlib.pyplot as plt

%matplotlib inline # this code help to keep graph in same page
```

UsageError: unrecognized arguments: # this code help to keep graph in same page

```
In [ ]: # import numpy as np
# import matplotlib.pyplot as plt
# %matplotlib inline
```

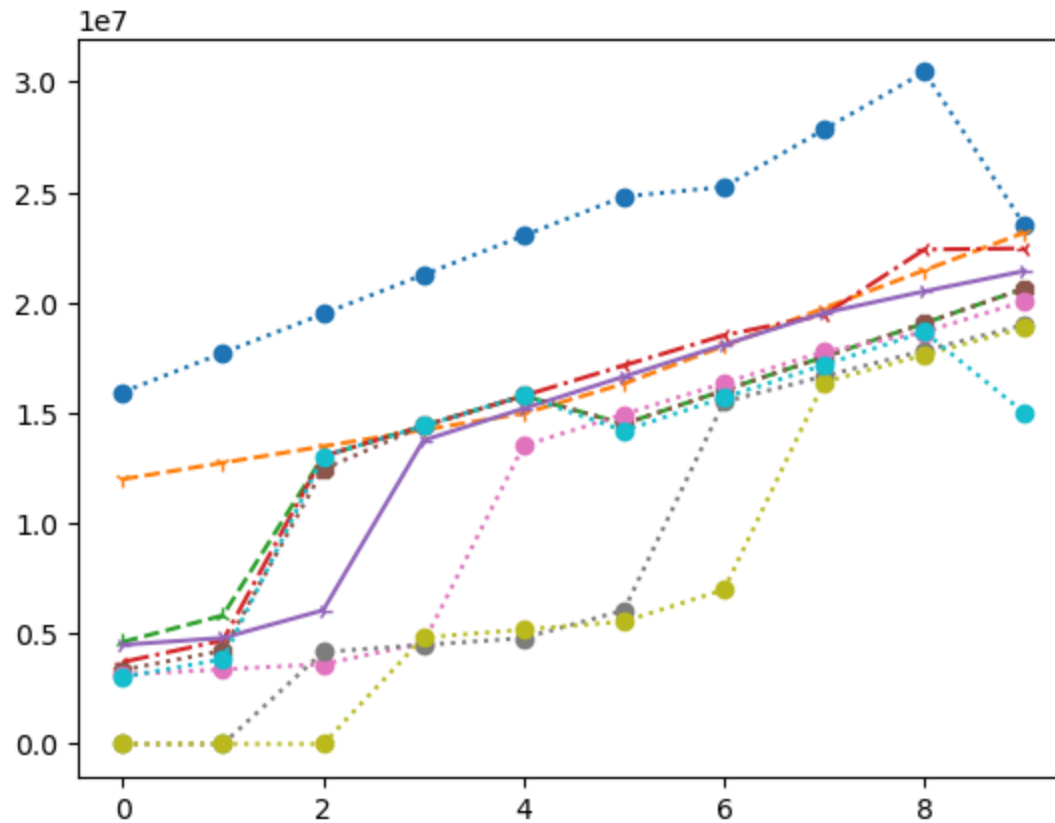
```
In [160... Salary[0]
```

```
Out[160... array([15946875, 17718750, 19490625, 21262500, 23034375, 24806250,
        25244493, 27849149, 30453805, 23500000])
```

```
In [210... # Salary graph
plt.plot(Salary[0],ls = ':',marker = 'o')
plt.plot(Salary[1],ls = '--',marker = '1')
plt.plot(Salary[2],ls = '--',marker = '2')
plt.plot(Salary[3],ls = '-.',marker = '3')
plt.plot(Salary[4],ls = '-',marker = '4')
plt.plot(Salary[5],ls = ':',marker = '8')

plt.plot(Salary[6],ls = ':', marker = 'o')
plt.plot(Salary[7],ls = ':', marker = 'o')
plt.plot(Salary[8],ls = ':', marker = 'o')
plt.plot(Salary[9],ls = ':', marker = 'o')
# plt.plot(Salary[],ls = ':', marker = 'o')
# plt.plot(Salary[],ls = ':', marker = 'o')
```

```
Out[210... [<matplotlib.lines.Line2D at 0x1a197429520>]
```



In [118...

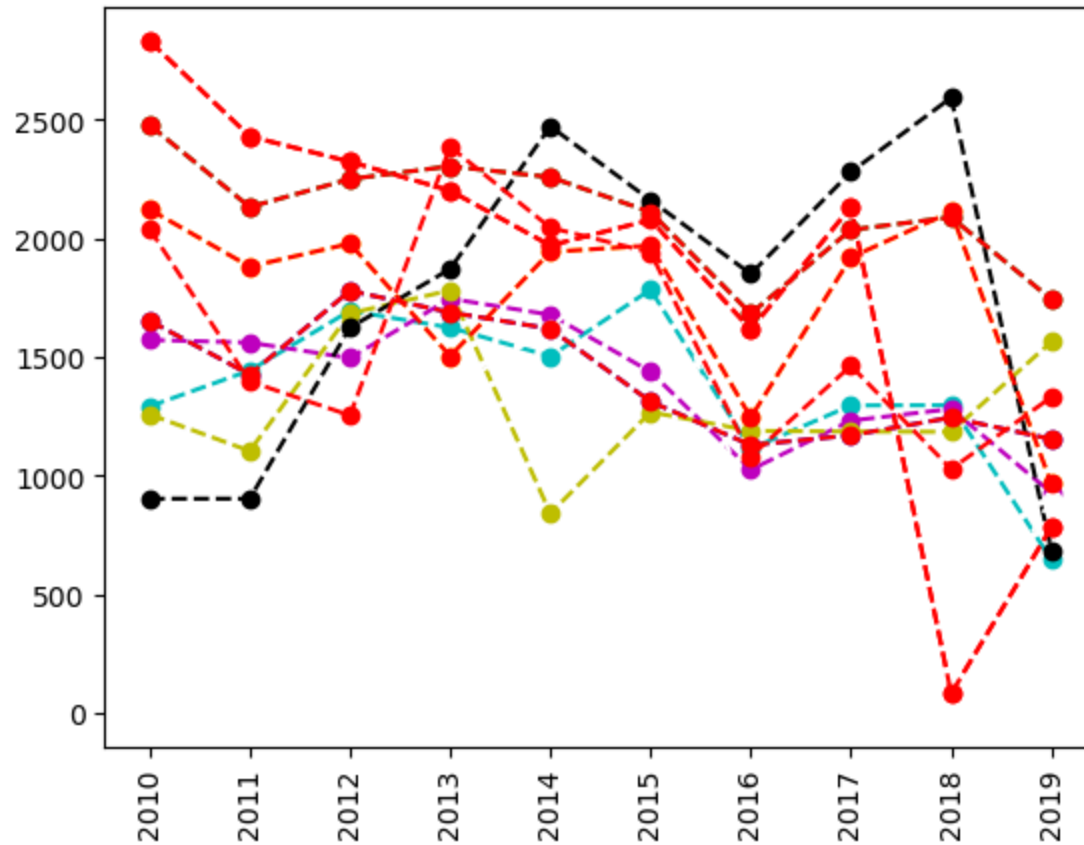
Points graph

```

plt.plot(Points[0],ls = '--',marker = 'o', c = 'red' ,label = Players[0])
plt.plot(Points[1],ls = '--',marker = 'o', c = 'blue', label = Players[1])
plt.plot(Points[2],ls = '--',marker = 'o', c = 'green', label = Players[2])
plt.plot(Points[3],ls = '--',marker = 'o', c = 'yellow', label = Players[3])
plt.plot(Points[4],ls = '--',marker = 'o', c = 'c' ,label = Players[4])
plt.plot(Points[5],ls = '--',marker = 'o', c = 'm', label = Players[5])
plt.plot(Points[6],ls = '--',marker = 'o', c = 'y', label = Players[6])
plt.plot(Points[7],ls = '--',marker = 'o', c = 'k', label = Players[7])
plt.plot(Points[8],ls = '--',marker = 'o', c = 'w', label = Players[8])
plt.plot(Points[9],ls = '--',marker = 'o', c = 'red', label = Players[9])
plt.plot(Points[0],ls = '--',marker = 'o', c = 'red', label = Players[0])
plt.plot(Points[1],ls = '--',marker = 'o', c = 'red', label = Players[1])
plt.plot(Points[2],ls = '--',marker = 'o', c = 'red', label = Players[2])
plt.plot(Points[3],ls = '--',marker = 'o', c = 'red', label = Players[3])

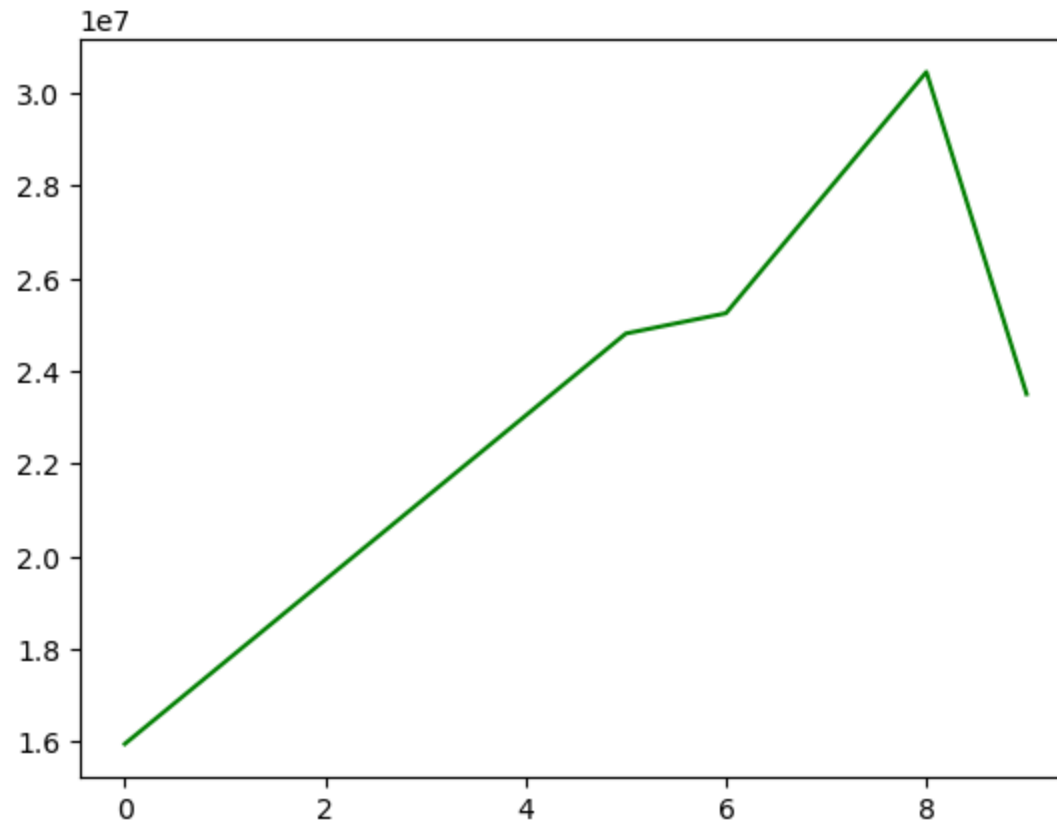
```

```
plt.xticks(list(range(0,10)),Seasons, rotation = 'vertical')
plt.show()
```



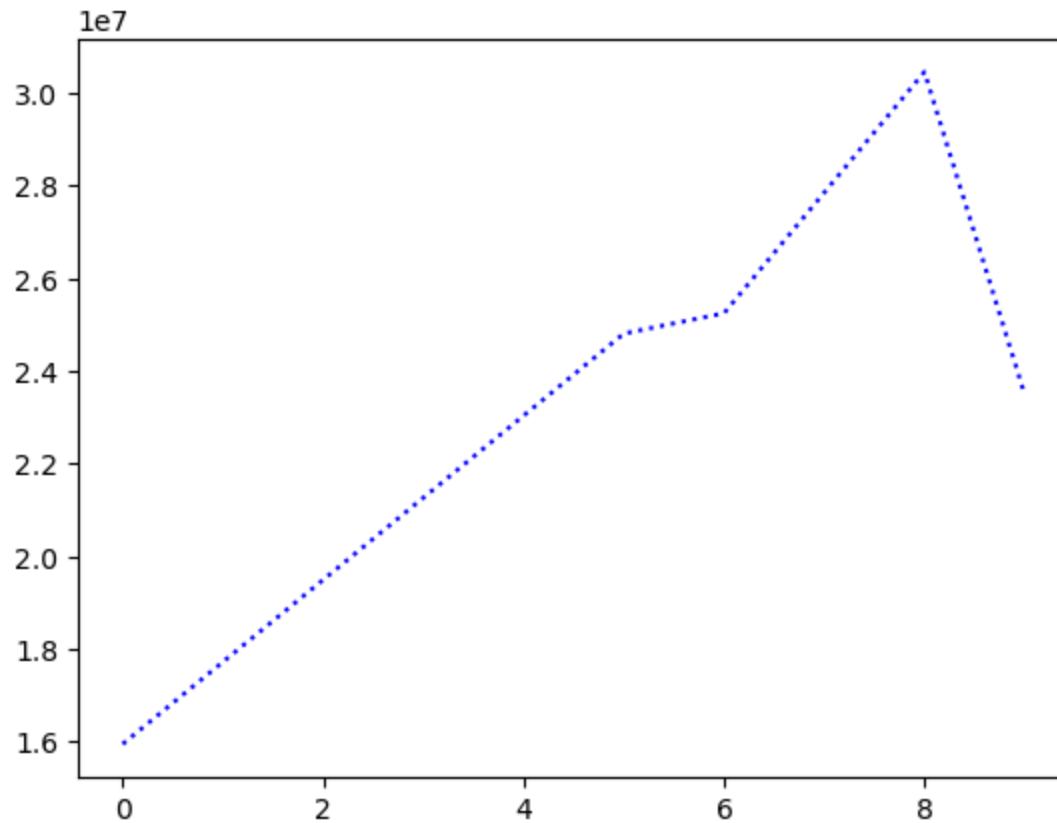
```
In [224...] plt.plot(Salary[0] , c = 'green')
```

```
Out[224...] [<matplotlib.lines.Line2D at 0x1a199147b60>]
```

```
In [238... plt.plot(Salary[0],ls = ':',c = 'blue')
```

```
Out[238... [<matplotlib.lines.Line2D at 0x1a199c21c10>]
```



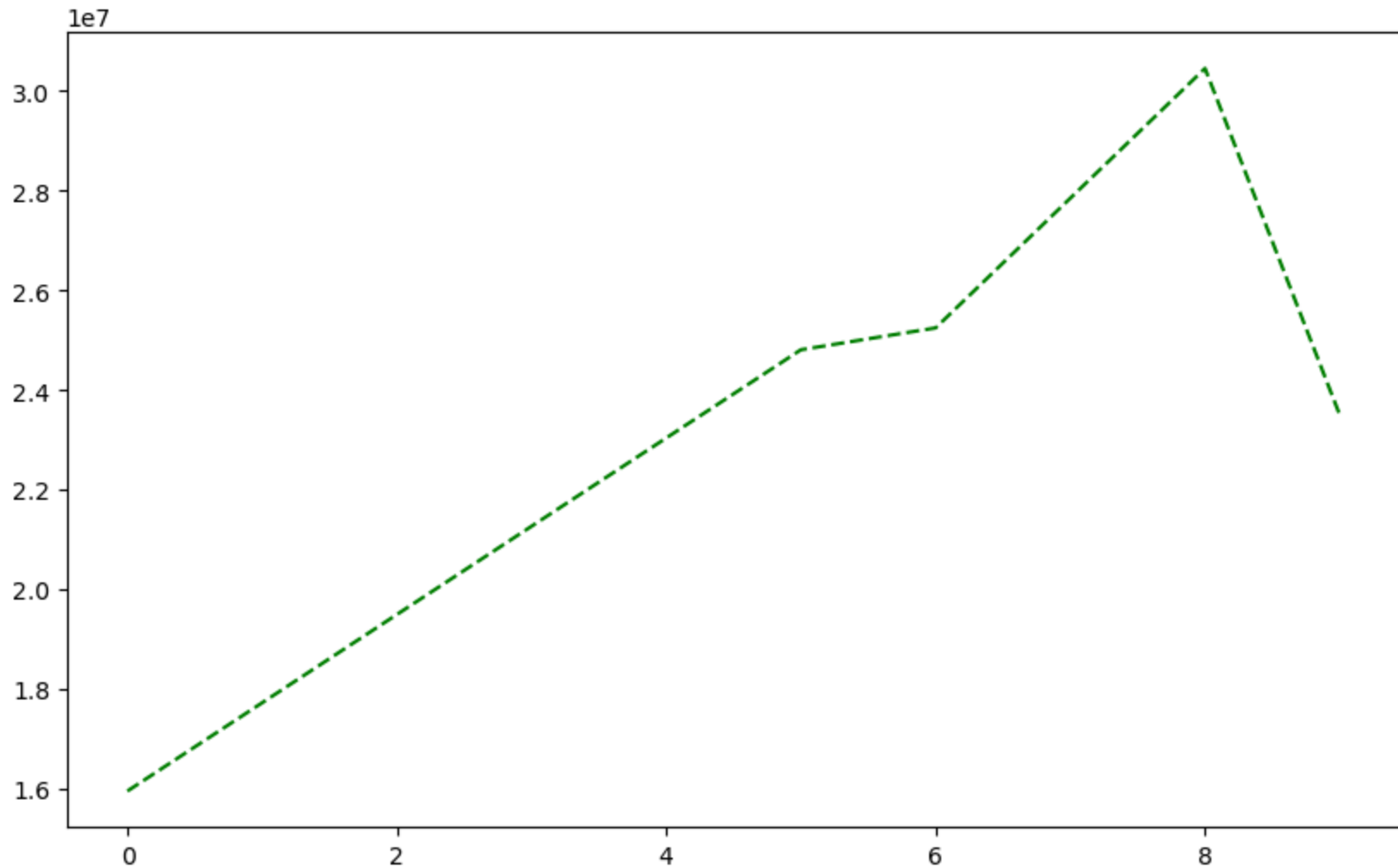
frame size

%matplotlib inline

`plt.rcParams['figure.figsize'] = 10,6` # used to change size of graph

```
In [262... plt.plot(Salary[0],ls = '--', c ='green')
```

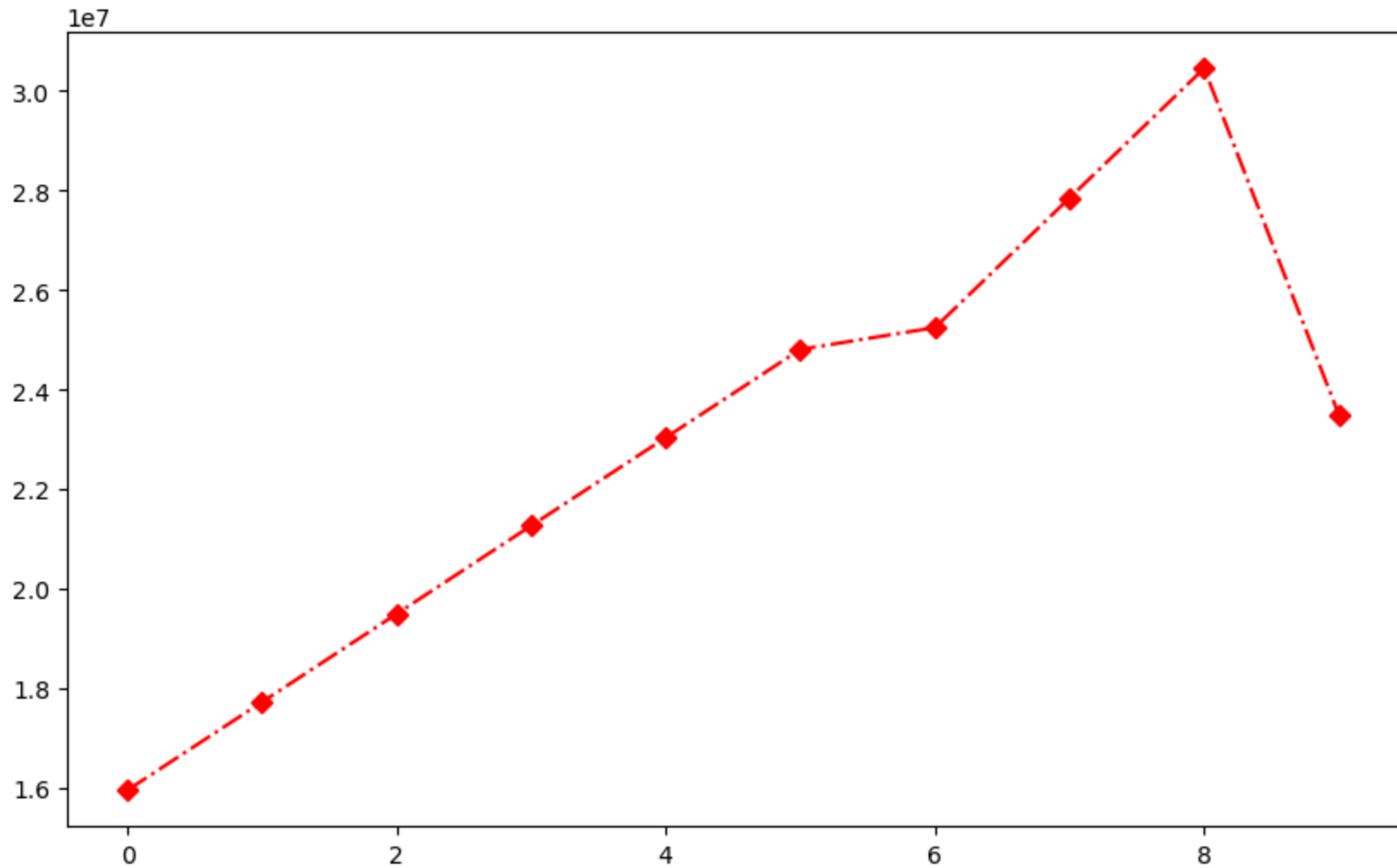
```
Out[262... [<matplotlib.lines.Line2D at 0x1a19adfc320>]
```



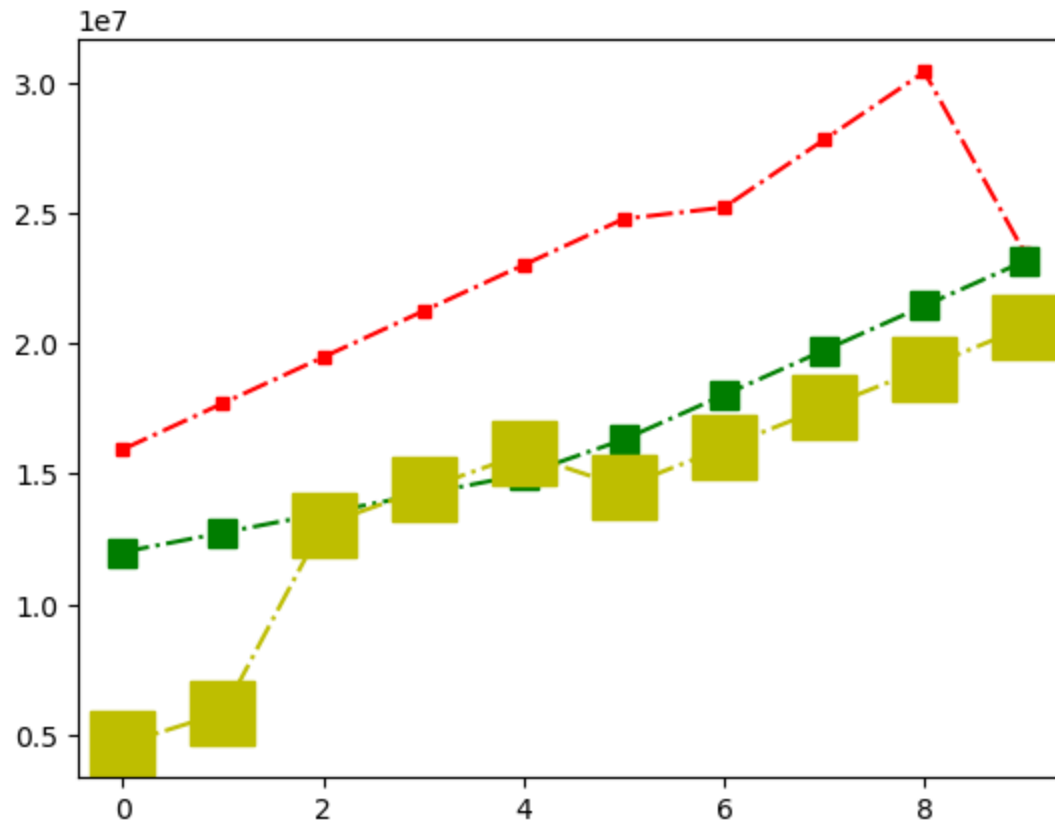
In [280...

```
# marker
# plt.plot(Salary[0],ls = '-.', c = 'red', marker = 's')
# plt.plot(Salary[0],ls = '-.', c = 'red', marker = 'p')
# plt.plot(Salary[0],ls = '-.', c = 'red', marker = 'P')
# plt.plot(Salary[0],ls = '-.', c = 'red', marker = 'H')
# plt.plot(Salary[0],ls = '-.', c = 'red', marker = 'h')
# plt.plot(Salary[0],ls = '-.', c = 'red', marker = 'x')
# plt.plot(Salary[0],ls = '-.', c = 'red', marker = 'd')
plt.plot(Salary[0],ls = '-.', c = 'red', marker = 'D')
```

Out[280... [`<matplotlib.lines.Line2D at 0x1a199144320>`]



```
In [8]: # ms == marker size
plt.plot(Salary[0],ls = '-.', c = 'red', marker = 's',ms = 5)
plt.plot(Salary[1],ls = '-.', c = 'g', marker = 's',ms = 10)
plt.plot(Salary[2],ls = '-.', c = 'y', marker = 's',ms = 23)
plt.show()
```



```
In [10]: l1 = list(range(0,10))  
l1
```

```
Out[10]: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

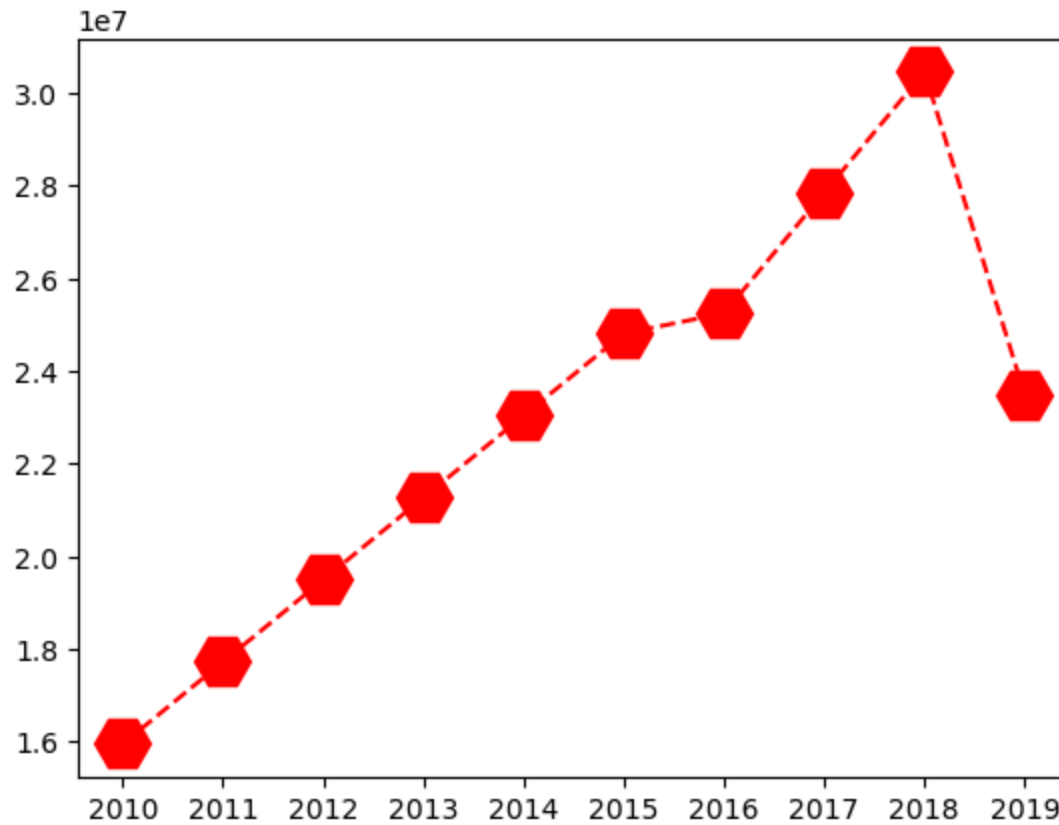
```
In [12]: Sdict
```

```
Out[12]: {'2010': 0,  
          '2011': 1,  
          '2012': 2,  
          '2013': 3,  
          '2014': 4,  
          '2015': 5,  
          '2016': 6,  
          '2017': 7,  
          '2018': 8,  
          '2019': 9}
```

```
In [14]: Pdict
```

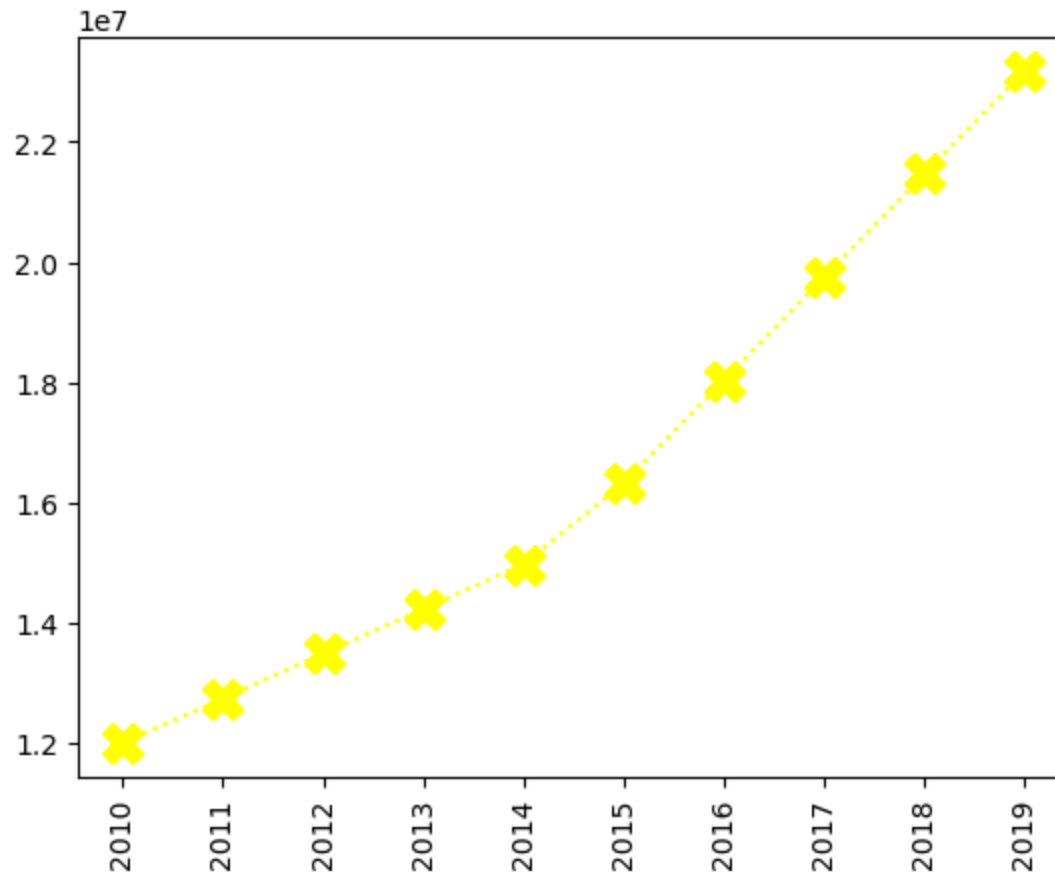
```
Out[14]: {'Sachin': 0,  
          'Rahul': 1,  
          'Smith': 2,  
          'Sami': 3,  
          'Pollard': 4,  
          'Morris': 5,  
          'Samson': 6,  
          'Dhoni': 7,  
          'Kohli': 8,  
          'Sky': 9}
```

```
In [26]: plt.plot(Salary[0],c = 'red', ls = '--', marker = 'H', ms = 20)  
         # plt.xticks(list(range(0,10)),Seasons) # code for x-axis and it's values / default horizontal  
         plt.xticks(11,Seasons)  
         plt.show()
```

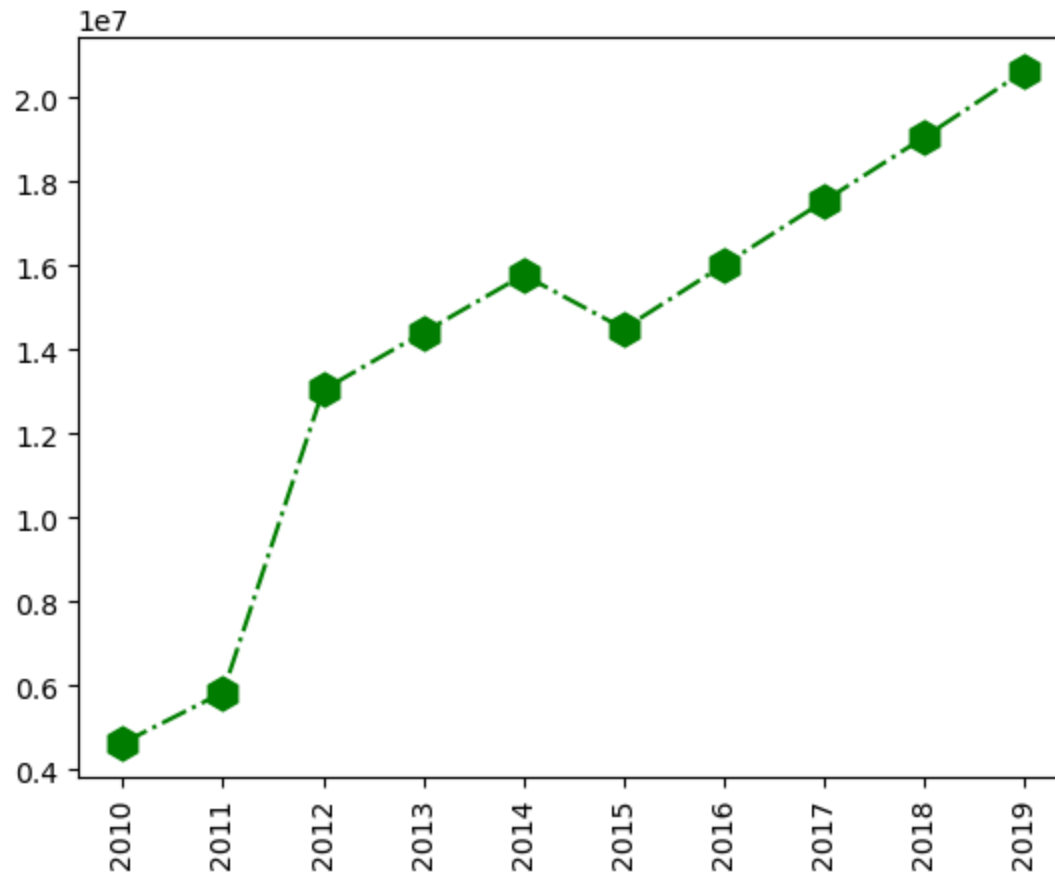


```
In [34]: # xticks horizontal to vertical
# plt.plot(Salary[2], c = 'Green', marker = 'h', ls = '-.', ms = 12)
# plt.xticks(list(range(0,10)),Seasons, rotation = 'vertical')
# plt.show()

plt.plot(Salary[1], c = 'yellow', ls = ':', marker = 'X', ms = 14)
plt.xticks(11,Seasons,rotation = 'vertical')
plt.show()
```

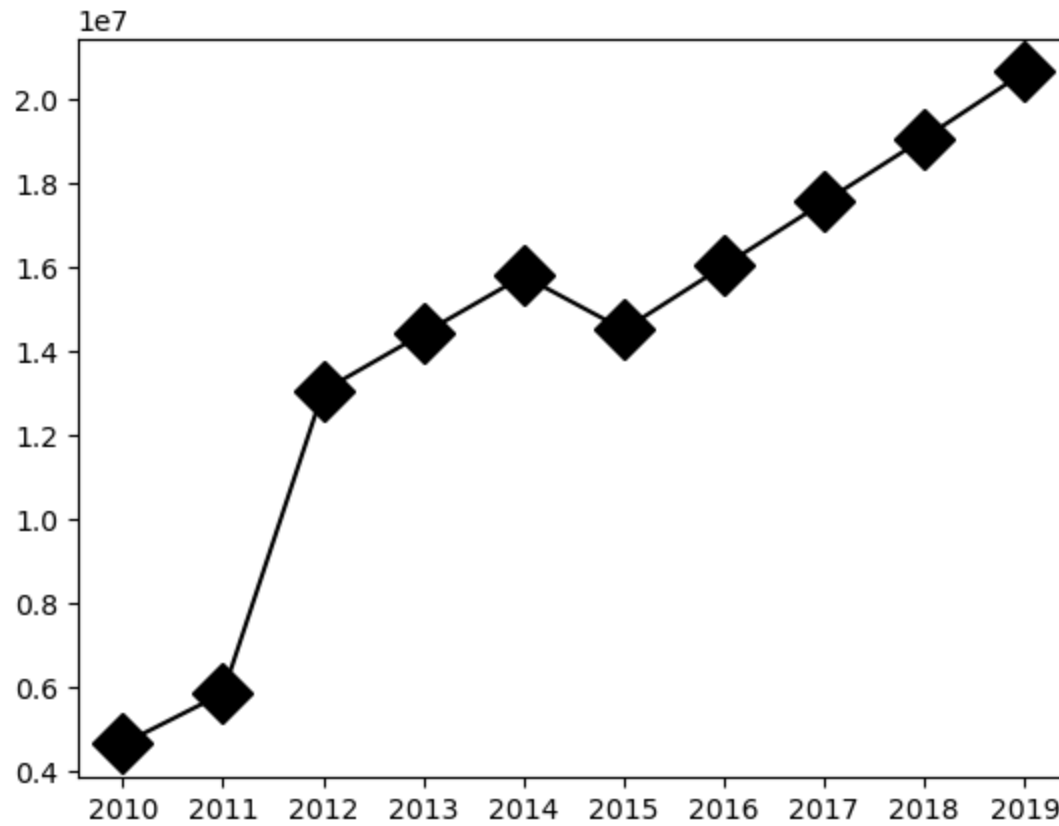


```
In [38]: # xticks horizontal to vertical
plt.plot(Salary[2], c = 'Green', marker = 'h', ls = '-.', ms = 12)
plt.xticks(list(range(0,10)),Seasons, rotation = 'vertical')
plt.show()
```

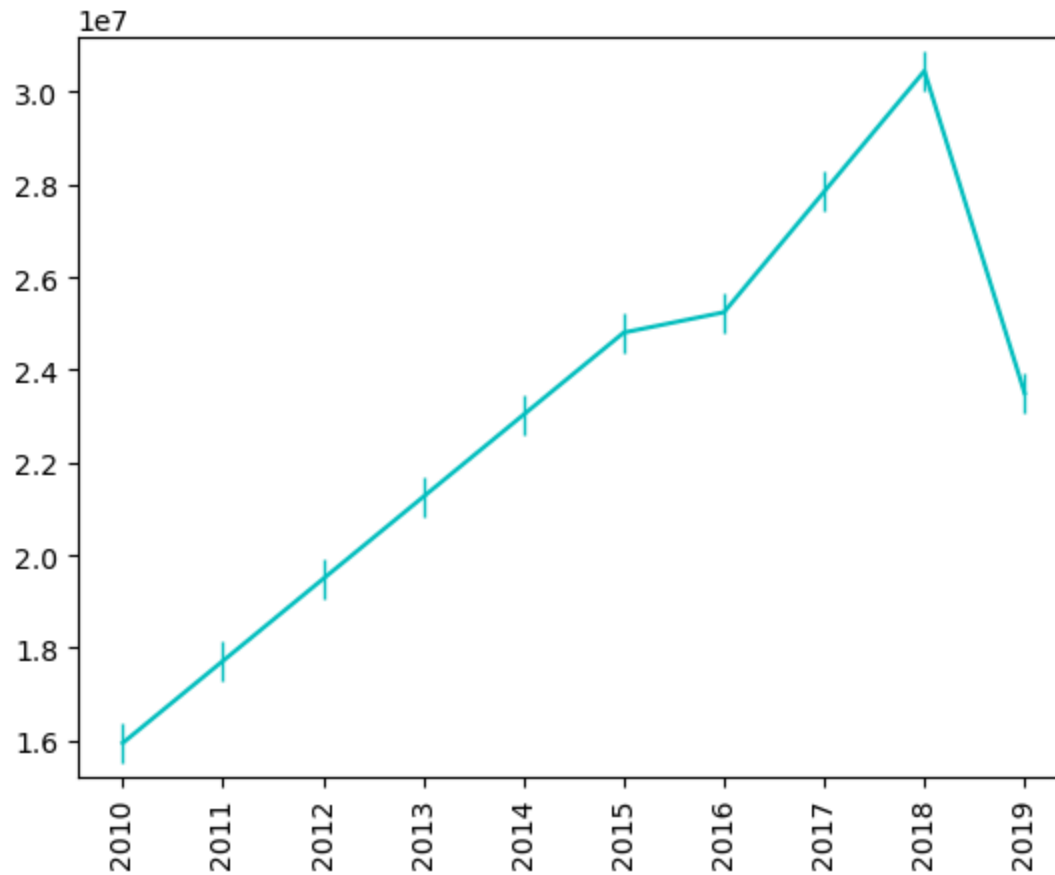



label - argument

```
In [90]: # Label argument
li = Salary[0]
plt.plot(Salary[2],c = 'black', marker = 'D',ms = 15, label = Players[2])
plt.xticks(list(range(0,10)), Seasons,rotation = 'horizontal')
# plt.yticks(li,[100,200,300,400,500,600,700,800,900,1000])
plt.show()
```



```
In [94]: plt.plot(Salary[0],c = 'c', marker = '|',ls = '-', ms = 15, label = Players[0])  
plt.xticks(list(range(0,10,)),Seasons,rotation = 'vertical')  
plt.show()
```



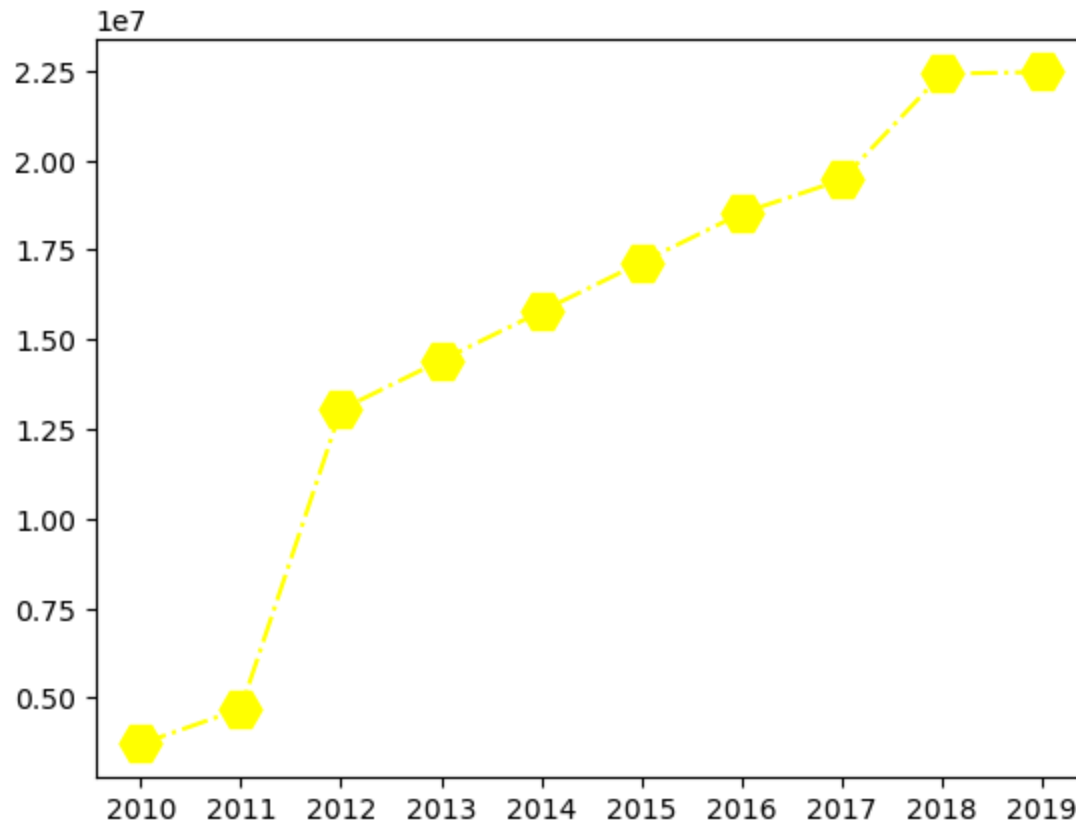
In [96]: Salary[0]

Out[96]: array([15946875, 17718750, 19490625, 21262500, 23034375, 24806250,
25244493, 27849149, 30453805, 23500000])

In [98]: Salary[1]

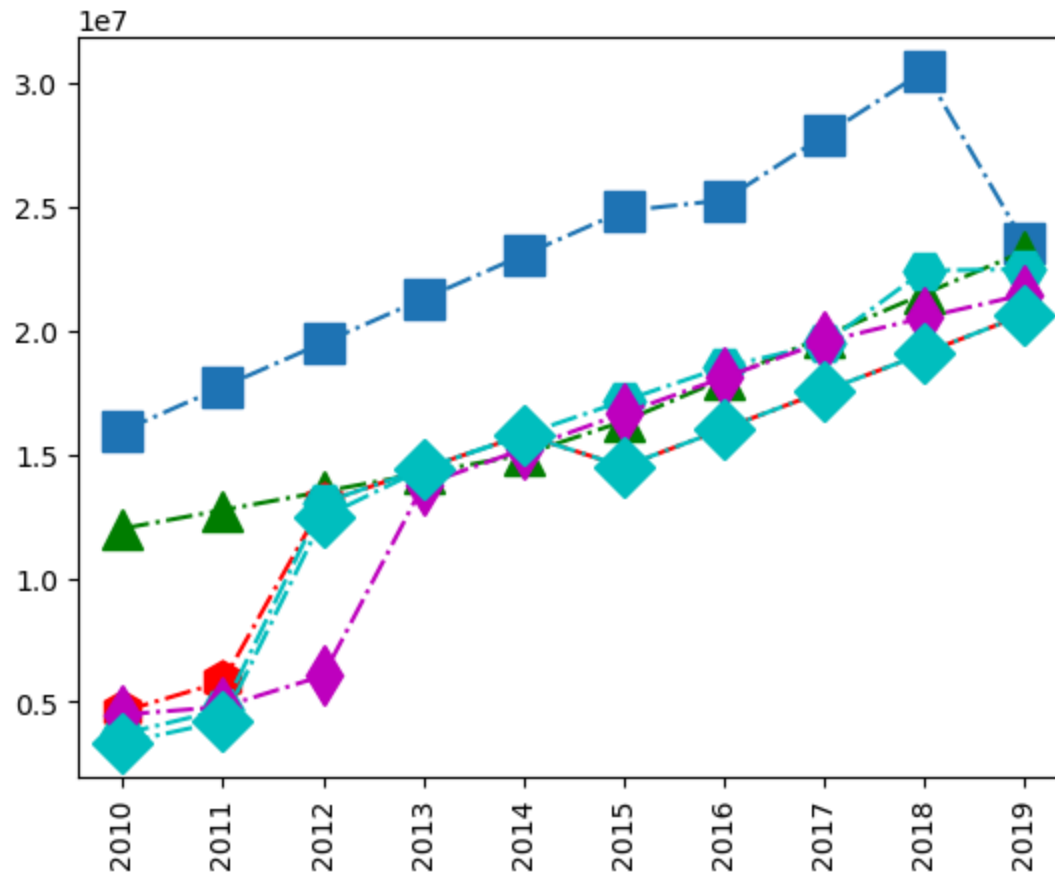
Out[98]: array([12000000, 12744189, 13488377, 14232567, 14976754, 16324500,
18038573, 19752645, 21466718, 23180790])

```
In [110]: plt.plot(Salary[3],marker = 'H',c = 'yellow', ls = '-.', ms = 15, label = Players[1])
plt.xticks(list(range(0,10)),Seasons, rotation = 'horizontal')
plt.show()
```



```
In [136... plt.plot(Salary[0], marker = 's', ms = '15', label = Players[0], ls = '-.')
plt.plot(Salary[1], marker = '^', ms = '15', label = Players[1], ls = '-.', color = 'g')
plt.plot(Salary[2], marker = 'h', ms = '15', label = Players[2], ls = '-.', color = 'r')
plt.plot(Salary[3], marker = 'H', ms = '15', label = Players[3], ls = '-.', color = 'c')
plt.plot(Salary[4], marker = 'd', ms = '15', label = Players[4], ls = '-.', color = 'm')
plt.plot(Salary[5], marker = 'D', ms = '15', label = Players[5], ls = '-.', color = 'c')

# plt.legend()
plt.xticks(list(range(0,10)),Seasons, rotation = 'vertical')
plt.show()
```

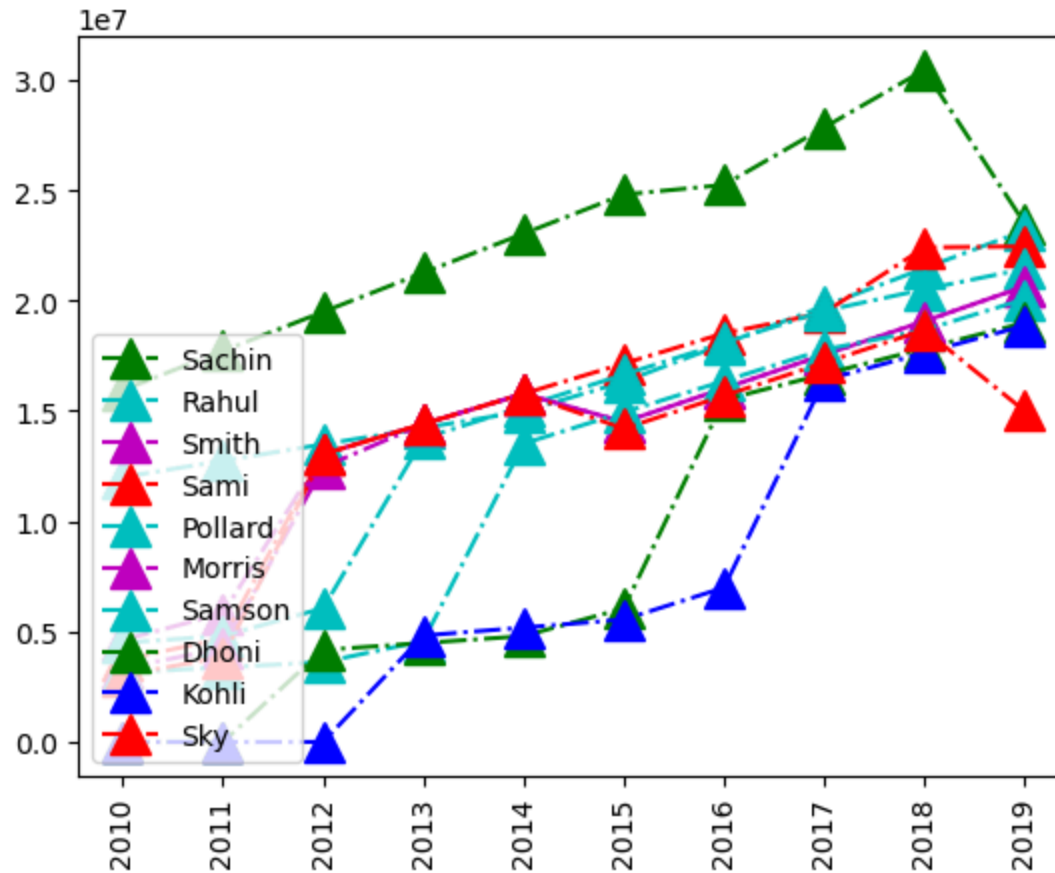


In [152... `# How to add a plt.plot(Salary[0], marker = '^', color = 'g', ls = '-.', ms = 15, label = Players[0])`
`# plt.legend()`

```
plt.plot(Salary[0], marker = '^', color = 'g', ls = '-.', ms = 15, label = Players[0])
plt.plot(Salary[1], marker = '^', color = 'c', ls = '-.', ms = 15, label = Players[1])
plt.plot(Salary[2], marker = '^', color = 'm', ls = '-.', ms = 15, label = Players[2])
plt.plot(Salary[3], marker = '^', color = 'r', ls = '-.', ms = 15, label = Players[3])
plt.plot(Salary[4], marker = '^', color = 'c', ls = '-.', ms = 15, label = Players[4])
plt.plot(Salary[5], marker = '^', color = 'm', ls = '-.', ms = 15, label = Players[5])
plt.plot(Salary[6], marker = '^', color = 'c', ls = '-.', ms = 15, label = Players[6])
plt.plot(Salary[7], marker = '^', color = 'g', ls = '-.', ms = 15, label = Players[7])
plt.plot(Salary[8], marker = '^', color = 'b', ls = '-.', ms = 15, label = Players[8])
plt.plot(Salary[9], marker = '^', color = 'r', ls = '-.', ms = 15, label = Players[9])
# chnage the legend location loc
```

```
plt.legend(loc = 'lower left')
# plt.legend(loc = 'upper left')
# plt.legend(loc = 'lower right')
# plt.legend(loc = 'upper right')

plt.xticks(list(range(0,10)),Seasons, rotation = 'vertical')
plt.show()
```

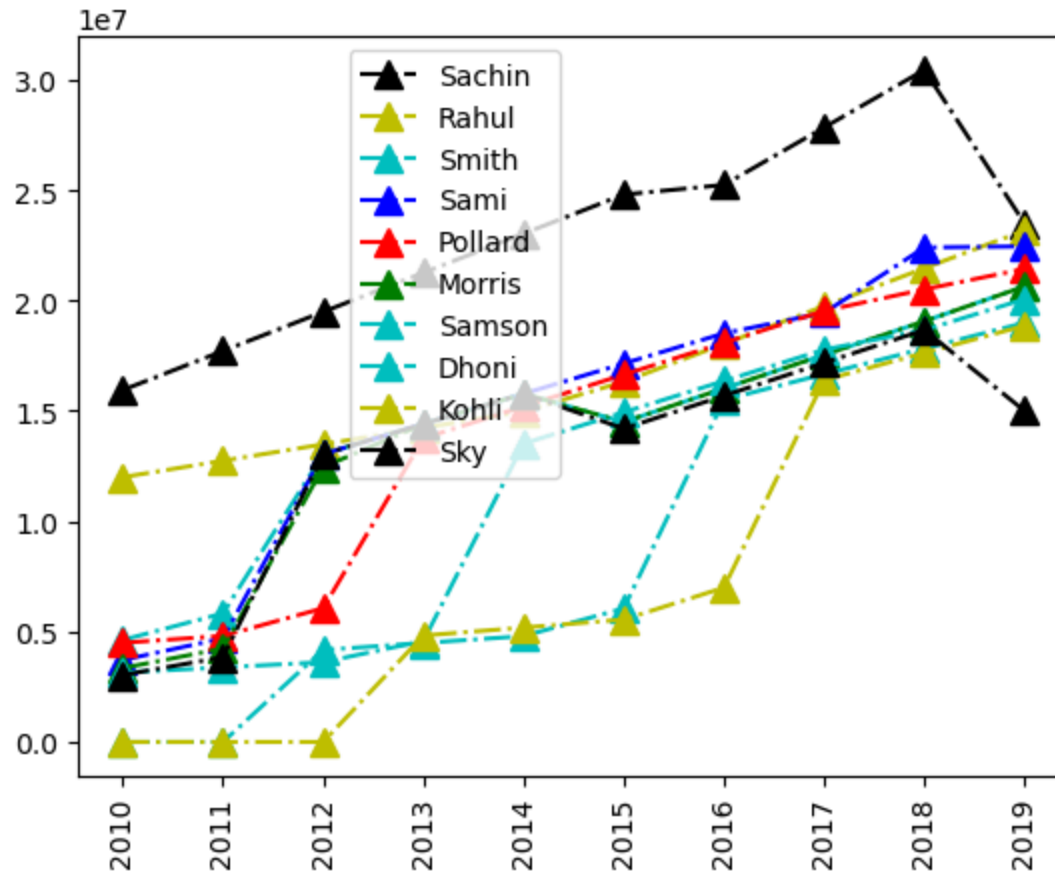


In [160...

```
plt.plot(Salary[0],marker = '^', ls = '-.-', ms = 10, label = Players[0], color = 'k')
plt.plot(Salary[1],marker = '^', ls = '-.-', ms = 10, label = Players[1], color = 'y')
plt.plot(Salary[2],marker = '^', ls = '-.-', ms = 10, label = Players[2], color = 'c')
plt.plot(Salary[3],marker = '^', ls = '-.-', ms = 10, label = Players[3], color = 'b')
plt.plot(Salary[4],marker = '^', ls = '-.-', ms = 10, label = Players[4], color = 'r')
plt.plot(Salary[5],marker = '^', ls = '-.-', ms = 10, label = Players[5], color = 'g')
plt.plot(Salary[6],marker = '^', ls = '-.-', ms = 10, label = Players[6], color = 'c')
```

```
plt.plot(Salary[7],marker = '^', ls = '-.-', ms = 10, label = Players[7], color = 'c')
plt.plot(Salary[8],marker = '^', ls = '-.-', ms = 10, label = Players[8], color = 'y')
plt.plot(Salary[9],marker = '^', ls = '-.-', ms = 10, label = Players[9], color = 'k')

plt.legend(loc = 'upper right', bbox_to_anchor = (0.5,1))
plt.xticks(list(range(0,10)),Seasons, rotation = 'vertical')
plt.show()
```



In []: