IPL DATA ANALYSIS

In [2]:
```python
import numpy as np
```

In [3]:
```python
#Seasons
Seasons = ["2010","2011","2012","2013","2014","2015","2016","2017","2018","2019"

Sdict = {"2010":0,"2011":1,"2012":2,"2013":3,"2014":4,"2015":5,"2016":6,"2017":7

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

#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]:  Games
```

```
Out[4]:  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 [5]:  points
```

```
Out[5]:  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 [6]:  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 [7]:  np.reshape(mydata,(4,5)) # 5rows and 4columns
```

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

mydata

```
In [8]:  #np.reshape(mydata,(5,4)  , order = "c" ) # 'C' means to read / write the elemen

         matr1 = np.reshape(mydata,(5,4), order = 'c' ) # elements are horizontally
         matr1
```

```
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]:  matr1
```

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

```
In [10]:  matr1[4,3]
```

Out[10]: 19

In [11]: `matr1[3,3]`

Out[11]: 15

In [12]: `matr1`

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

In [13]: `matr1[-3,-1]`

Out[13]: 11

In [14]: `matr1`

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

In [15]: `mydata`

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

In [16]:
```python
matr2 = np.reshape(mydata,(5,4) , order = "F") #reshape behavior are -'C'  , 'F'

matr2
```

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

In [17]: `matr2[4,3]  # 4th row and 3rd column  , numpy indexing and slicing start from 0`

Out[17]: 19

In [18]: `matr2[0,2]`

Out[18]: 10

In [19]: `matr2[0:2]`

Out[19]:
```
array([[ 0,  5, 10, 15],
       [ 1,  6, 11, 16]])
```

In [20]: `matr2`

```
Out[20]:   array([[ 0,  5, 10, 15],
                  [ 1,  6, 11, 16],
                  [ 2,  7, 12, 17],
                  [ 3,  8, 13, 18],
                  [ 4,  9, 14, 19]])
```

In [21]:   `matr2[1,2]`

Out[21]:   11

In [22]:   `matr2`

```
Out[22]:   array([[ 0,  5, 10, 15],
                  [ 1,  6, 11, 16],
                  [ 2,  7, 12, 17],
                  [ 3,  8, 13, 18],
                  [ 4,  9, 14, 19]])
```

In [23]:   `matr2[-2,-1]`

Out[23]:   18

In [24]:   `matr2[-3,-3]`

Out[24]:   7

In [25]:   `matr2`

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

In [26]:   `matr2[0:2]`

```
Out[26]:   array([[ 0,  5, 10, 15],
                  [ 1,  6, 11, 16]])
```

In [27]:   `mydata`

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

In [28]:   ```
           matr3 = np.reshape(mydata,(5,4),order = 'A')
           matr3
           ```

```
Out[28]:   array([[ 0,  1,  2,  3],
                  [ 4,  5,  6,  7],
                  [ 8,  9, 10, 11],
                  [12, 13, 14, 15],
                  [16, 17, 18, 19]])
```

In [29]:   `matr2`

```
Out[29]:  array([[ 0,  5, 10, 15],
                 [ 1,  6, 11, 16],
                 [ 2,  7, 12, 17],
                 [ 3,  8, 13, 18],
                 [ 4,  9, 14, 19]])
```

```
In [30]:  matr1
```

```
Out[30]:  array([[ 0,  1,  2,  3],
                 [ 4,  5,  6,  7],
                 [ 8,  9, 10, 11],
                 [12, 13, 14, 15],
                 [16, 17, 18, 19]])
```

```
In [31]:  a1 = ["welcome" , "to" , "datascience"]

          a2 = ['required' , 'hard' , 'work']

          a3 =[1,2,3]
```

```
In [32]:  [a1,a2,a3]
```

```
Out[32]:  [['welcome', 'to', 'datascience'], ['required', 'hard', 'work'], [1, 2, 3]]
```

```
In [33]:  np.array([a1,a2,a3]) # u11 - unicode 11 character: 3*3 matrix
```

```
Out[33]:  array([['welcome', 'to', 'datascience'],
                 ['required', 'hard', 'work'],
                 ['1', '2', '3']], dtype='<U11')
```

```
In [34]:  Games
```

```
Out[34]:  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 [35]:  Games[0]
```

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

```
In [36]:  Games[0:5]
```

```
Out[36]:  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 [37]:  Games[0,5]
```

```
Out[37]:  82
```

```
In [38]:   Games[0,2]
```

```
Out[38]:   82
```

```
In [39]:   Games
```

```
Out[39]:   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 [40]:   Games[0:2]
```

```
Out[40]:   array([[80, 77, 82, 82, 73, 82, 58, 78,  6, 35],
                  [82, 57, 82, 79, 76, 72, 60, 72, 79, 80]])
```

```
In [41]:   games =  Games
```

```
In [42]:   games[1:2]
```

```
Out[42]:   array([[82, 57, 82, 79, 76, 72, 60, 72, 79, 80]])
```

```
In [43]:   games[2]
```

```
Out[43]:   array([79, 78, 75, 81, 76, 79, 62, 76, 77, 69])
```

```
In [44]:   games
```

```
Out[44]:   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 [45]:   games[2,8]
```

```
Out[45]:   77
```

```
In [46]:   games
```

```
Out[46]: 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 [47]: `games[-3:-1]`

```
Out[47]: array([[35, 35, 80, 74, 82, 78, 66, 81, 81, 27],
                [40, 40, 40, 81, 78, 81, 39,  0, 10, 51]])
```

In [48]: `id(games)`

Out[48]: 1876622585232

In [49]: `id(Games)`

Out[49]: 1876622585232

In [50]: `points`

```
Out[50]: 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 [51]:
```python
Points = points.copy()

print(id(Points))
```
1876623179184

In [52]: `print(id(points))`
1876622584848

In [53]: `Points = points`

In [54]: `Points`

```
Out[54]:  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 [55]:  Points[0]
```

```
Out[55]:  array([2832, 2430, 2323, 2201, 1970, 2078, 1616, 2133,   83,  782])
```

```
In [56]:  Points
```

```
Out[56]:  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 [57]:  Points[6,1]
```

```
Out[57]:  1104
```

```
In [58]:  Points[3:6]
```

```
Out[58]:  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 [59]:  Points
```

```
Out[59]:  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 [60]:  Points[-6,-1]
```

```
Out[60]:  646
```

```
In [61]:  # dictionary

          # dict does not maintain the order
```

```python
dict1 = {'key1': 'val1' , 'key2':'val2','key3' : 'val3'}

type(dict1)
```

Out[61]: dict

In [62]:
```python
dict1['key2']
```

Out[62]: 'val2'

In [63]:
```python
dict2 = {'bang':2 , 'hyd' : ' we are hear' , 'pune' :True}
dict2
```

Out[63]: {'bang': 2, 'hyd': ' we are hear', 'pune': True}

In [64]:
```python
dict3 = {'germany': 'I have been here' , 'France':2, 'spain': True}
dict3
```

Out[64]: {'germany': 'I have been here', 'France': 2, 'spain': True}

In [65]:
```python
dict3['germany']
```

Out[65]: 'I have been here'

In [66]:
```python
games
```

Out[66]: 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 [67]:
```python
Pdict
```

Out[67]: {'Sachin': 0,
          'Rahul': 1,
          'Smith': 2,
          'Sami': 3,
          'Pollard': 4,
          'Morris': 5,
          'Samson': 6,
          'Dhoni': 7,
          'Kohli': 8,
          'Sky': 9}

In [68]:
```python
Pdict['Sachin']
```

Out[68]: 0

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

Out[69]: array([80, 77, 82, 82, 73, 82, 58, 78, 6, 35])

```
In [70]:  Games
```

```
Out[70]:  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 [71]:  Pdict['Rahul']
```

```
Out[71]:  1
```

```
In [72]:  Games[1]
```

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

Games

```
In [74]:  Games[Pdict['Rahul']]
```

```
Out[74]:  array([82, 57, 82, 79, 76, 72, 60, 72, 79, 80])
```

```
In [75]:  Points
```

```
Out[75]:  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 [76]:  Salary
```

Out[76]: 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 [77]: Salary[2,4]

Out[77]: 15779912

In [78]: Salary

Out[78]: 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 [79]: Pdict['Sky']

Out[79]: 9

In [80]: Sdict['2019']

Out[80]: 9

In [81]: Salary[Pdict['Sky']][Sdict['2019']]

Out[81]:   15000000

In [82]:
```python
#Seasons
Seasons = ["2010","2011","2012","2013","2014","2015","2016","2017","2018","2019"

Sdict = {"2010":0,"2011":1,"2012":2,"2013":3,"2014":4,"2015":5,"2016":6,"2017":7

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

#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 [83]:
```python
Salary
```

```
Out[83]:  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 [84]:  Games
```

```
Out[84]:  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 [85]:  import warnings
          warnings.filterwarnings("ignore")
          Salary/Games
```

```
Out[85]: 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]])
```

Visualization

```
In [87]: import numpy as np , matplotlib.pyplot as plt

         %matplotlib.inline # keep the plot inside jupyter nots insted of getting in othe
```

```
UsageError: Line magic function `%matplotlib.inline` not found.
```

```
In [120…  Salary
```

```
Out[120…    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…
```python
Salary[0]
```

Out[122…
```
array([15946875, 17718750, 19490625, 21262500, 23034375, 24806250,
       25244493, 27849149, 30453805, 23500000])
```

In [123…
```python
plt.plot(Salary[0])
plt.title('Salary')
```

Out[123…   Text(0.5, 1.0, 'Salary')

In [124…
```python
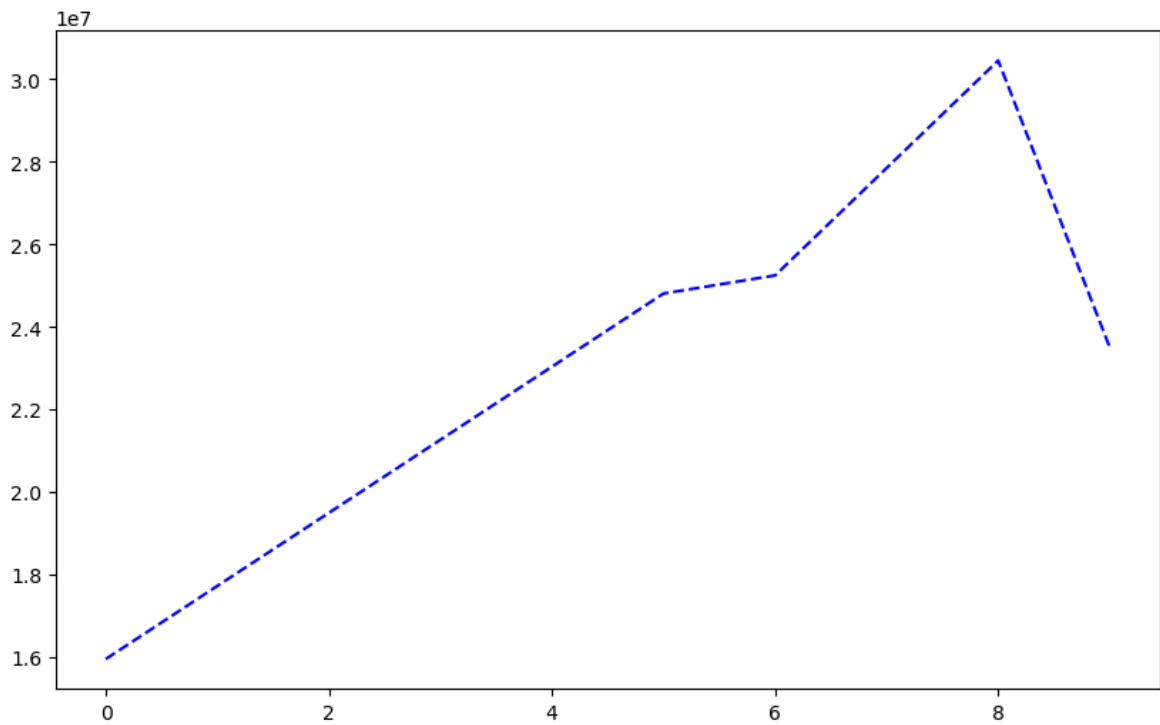plt.plot(Salary[0],c = 'red') # color = red
plt.show()
```
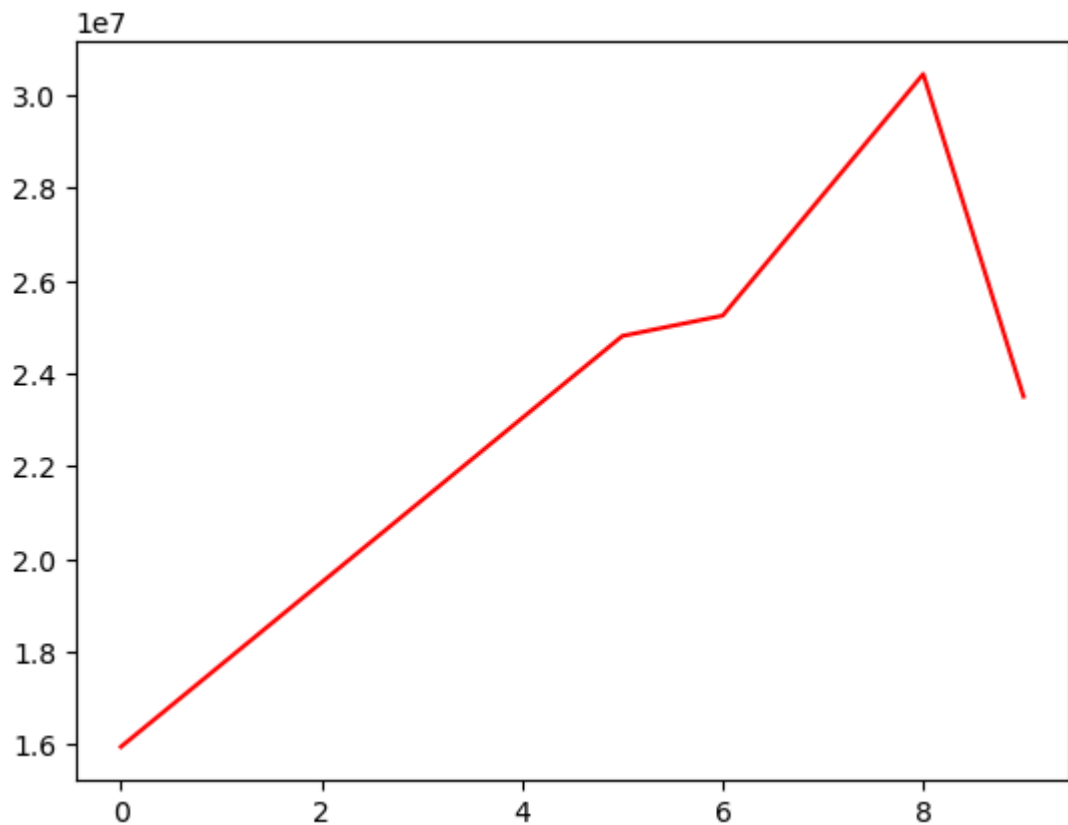
```
In [ ]:
```

Colors

The supported color abbreviations are the single letter codes

```
=============    ===============================
character        color
=============    ===============================
``'b'``          blue
``'g'``          green
``'r'``          red
``'c'``          cyan
``'m'``          magenta
``'y'``          yellow
``'k'``          black
``'w'``          white
=============    ===============================
```

```
In [127…    %matplotlib inline
            plt.rcParams['figure.figsize'] = 10,6
            plt.show()
            plt.plot(Salary[0] , c = 'Blue' , ls ='dashed')
```

```
Out[127…    [<matplotlib.lines.Line2D at 0x1b4f05ced10>]
```

Line Styles

```
=============      ==============================
character          description
=============      ==============================
``'-'``            solid line style
``'--'``           dashed line style
``'-.'``           dash-dot line style
``':'``            dotted line style
=============      ==============================
```

In [129…   ```python
            plt.plot(Salary[0] , c = "Green" , ls = '--' ,marker ='o') #
            ```

Out[129…    [<matplotlib.lines.Line2D at 0x1b4f0611cd0>]

```
In [130... plt.rcParams['figure.figsize'] = 10,10 # runtime configuration parameter
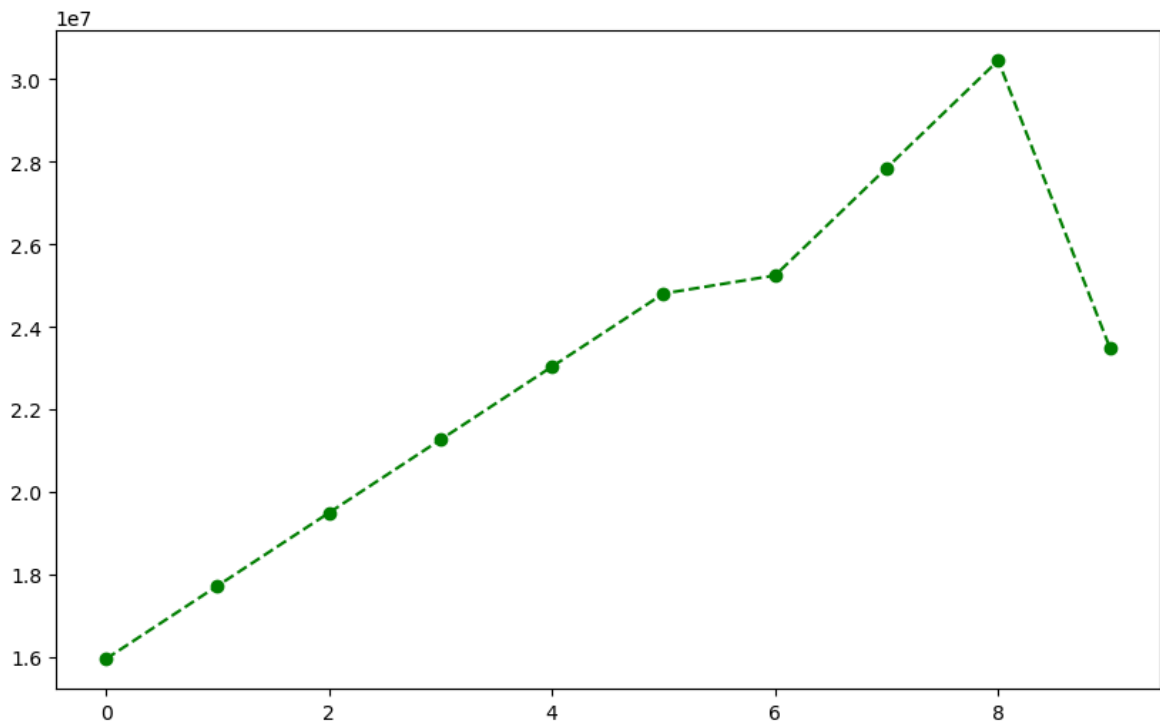
          # this is actually work on graph size
```

```
In [131... plt.plot(Salary[0] ,c = 'Green' , ls = '--' , marker = 's' ,ms =10)
          plt.show()
```

```
list(range(0,10))
```

[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

```
Sdict
```

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

```
Pdict
```

Out[134…   {'Sachin': 0,
   'Rahul': 1,
            'Smith': 2,
            'Sami': 3,
            'Pollard': 4,
            'Morris': 5,
            'Samson': 6,
            'Dhoni': 7,
            'Kohli': 8,
            'Sky': 9}

In [135…
```python
plt.plot(Salary[0],c = 'Green', ls = '--' ,marker = 's' ,ms = 7 )# c = color , #
```

Out[135…   [<matplotlib.lines.Line2D at 0x1b4f0687b90>]
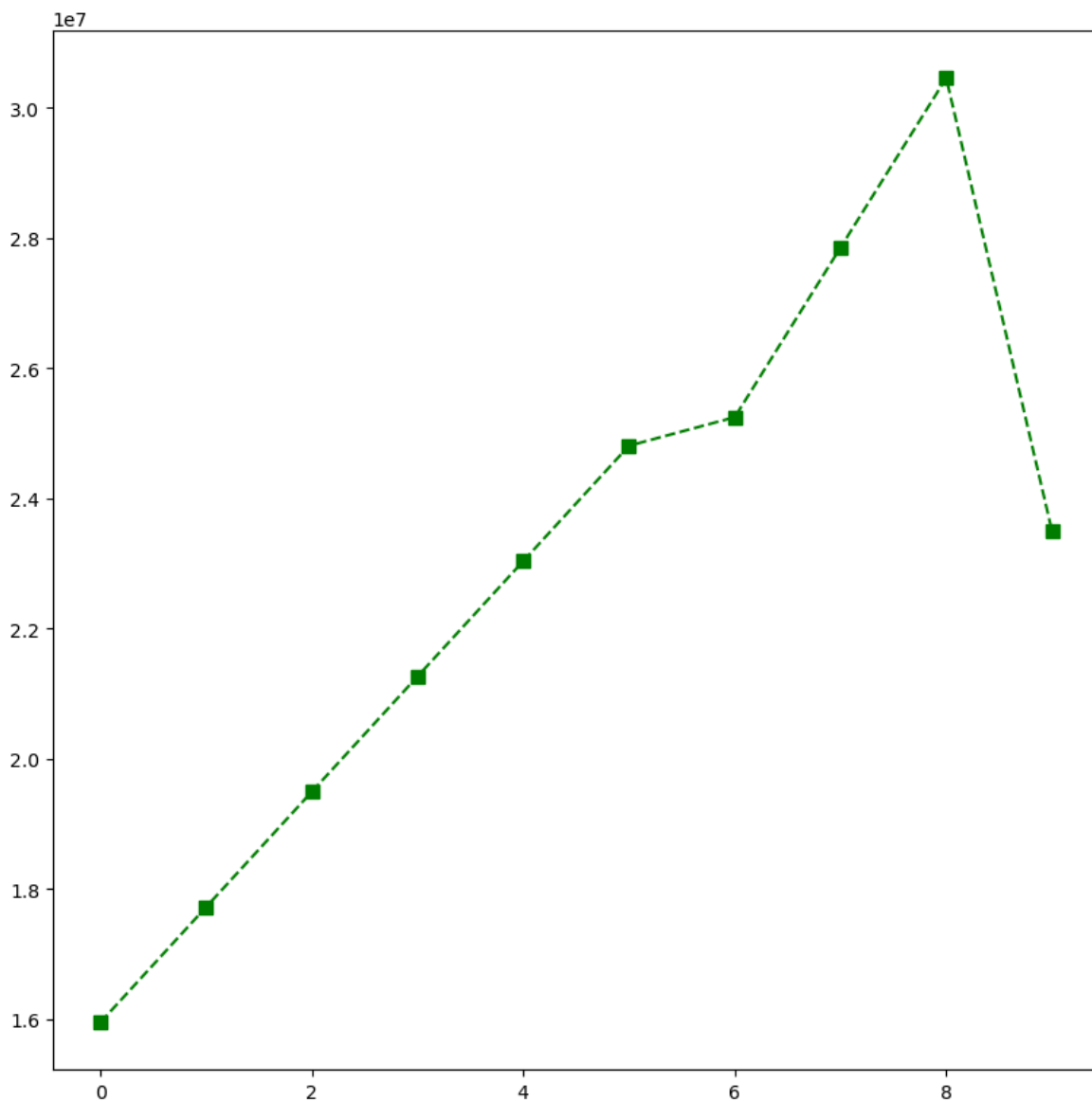
Markers

```
=============    ===============================
character        description
=============    ===============================
``'.'``          point marker
``','``          pixel marker
``'o'``          circle marker
``'v'``          triangle_down marker
```

```
        ``'^'``            triangle_up marker
        ``'<'``            triangle_left marker
        ``'>'``            triangle_right marker
        ``'1'``            tri_down marker
        ``'2'``            tri_up marker
        ``'3'``            tri_left marker
        ``'4'``            tri_right marker
        ``'8'``            octagon marker
        ``'s'``            square marker
        ``'p'``            pentagon marker
        ``'P'``            plus (filled) marker
        ``'*'``            star marker
        ``'h'``            hexagon1 marker
        ``'H'``            hexagon2 marker
        ``'+'``            plus marker
        ``'x'``            x marker
        ``'X'``            x (filled) marker
        ``'D'``            diamond marker
        ``'d'``            thin_diamond marker
        ``'|'``            vline marker
        ``'_'``            hline marker
        =============      ==============================
```

In [137…   `list(range(0,10))`

Out[137…   `[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]`

In [138…   `Sdict`

Out[138…
```
{'2010': 0,
 '2011': 1,
 '2012': 2,
 '2013': 3,
 '2014': 4,
 '2015': 5,
 '2016': 6,
 '2017': 7,
 '2018': 8,
 '2019': 9}
```

In [139…   `Pdict`

Out[139…
```
{'Sachin': 0,
 'Rahul': 1,
 'Smith': 2,
 'Sami': 3,
 'Pollard': 4,
 'Morris': 5,
 'Samson': 6,
 'Dhoni': 7,
 'Kohli': 8,
 'Sky': 9}
```

In [140…
```python
plt.plot(Salary[0], c = "Green" , ls ='--' , marker = 's' ,ms = 7, markeredgecol
plt.xticks(list(range(0,10)) ,Seasons) # markerfacecolor = mfc
plt.show()
```
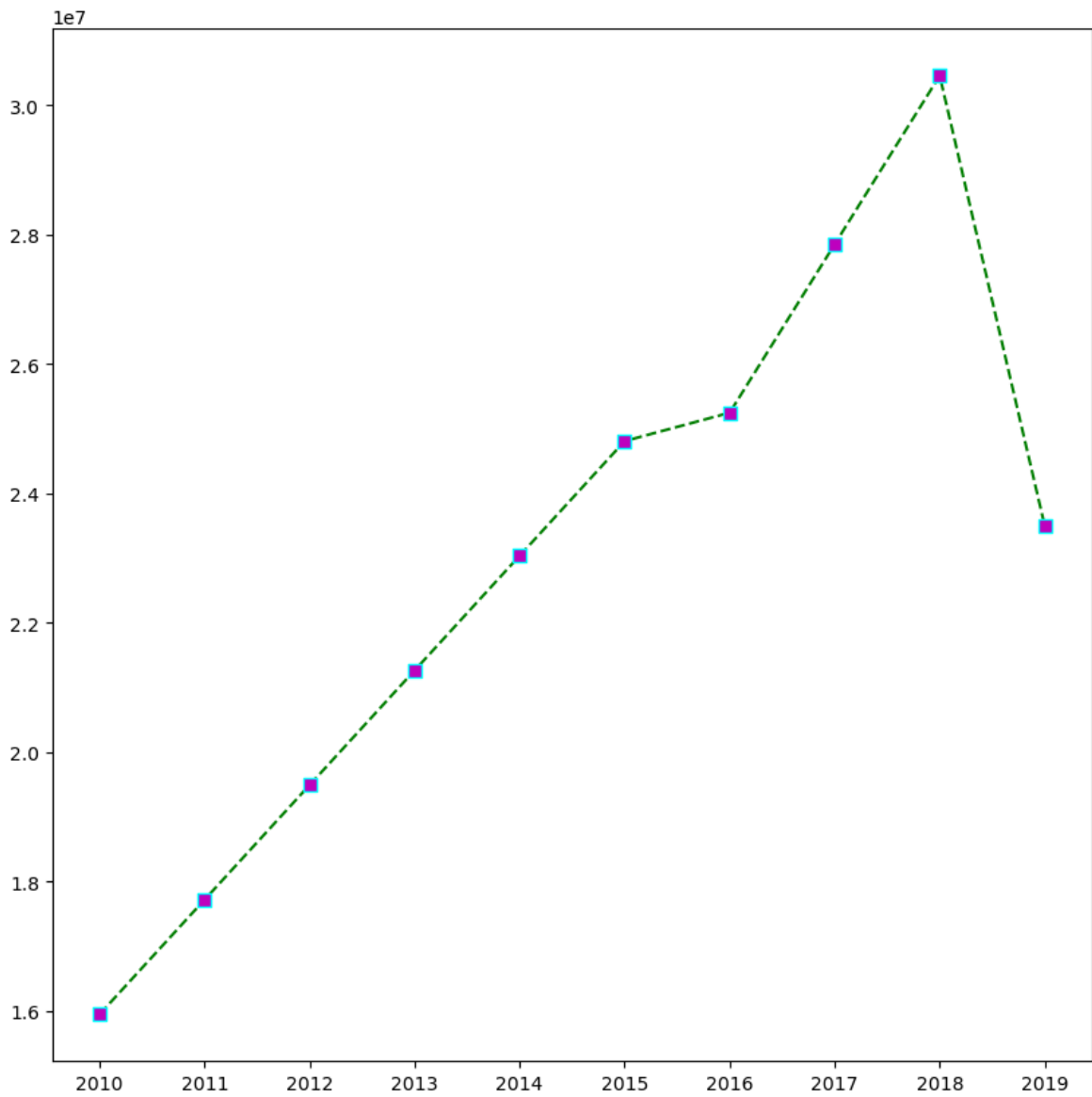
```
In [141... plt.plot(Salary[0] , c = "Green" , ls = ':' , marker = 's' , ms = 7 , label = Pl
          plt.xticks(list(range(0,10)),Seasons , rotation = 'vertical')
          plt.show()
```
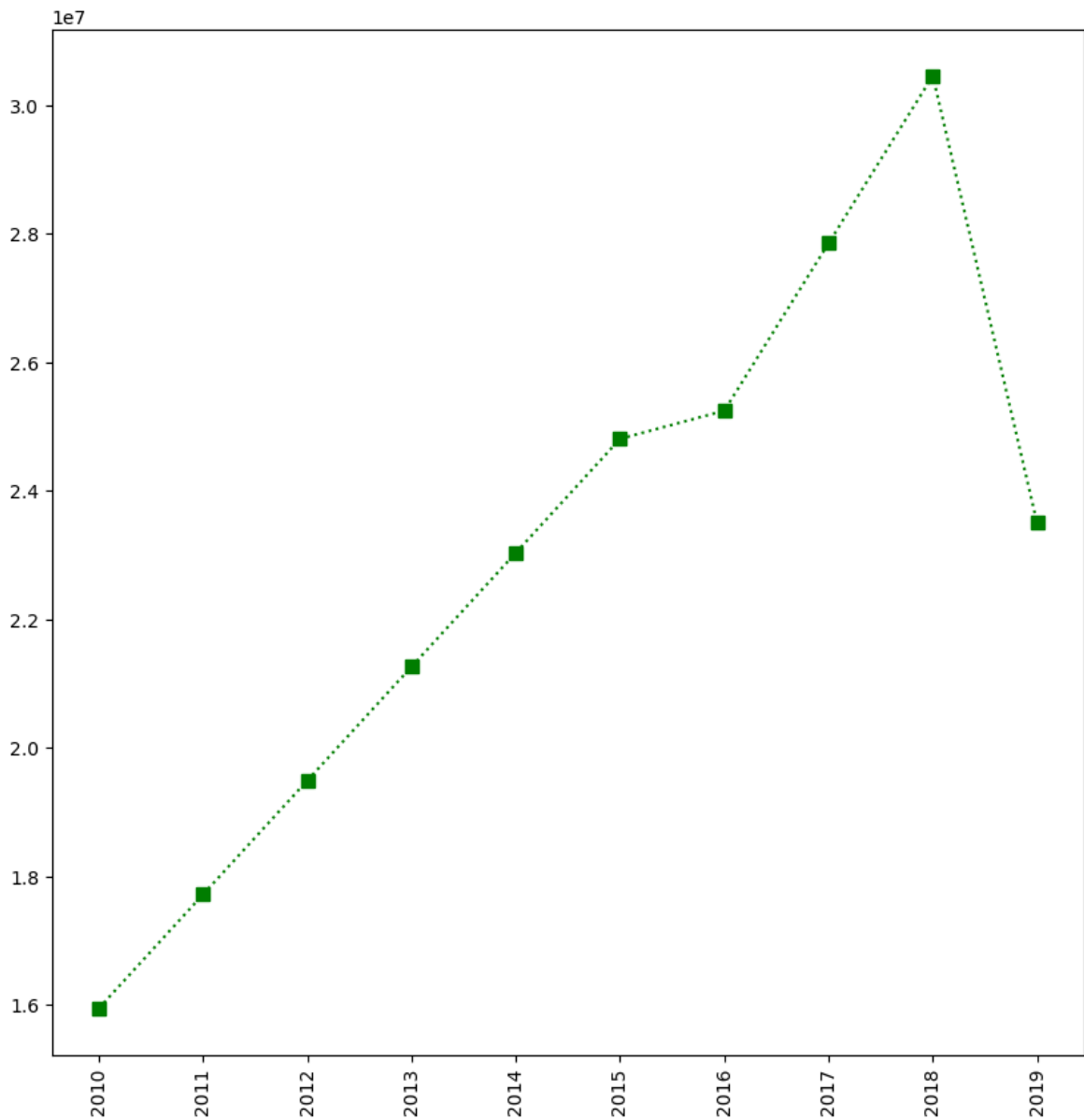
In [142...   Games

Out[142...   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 [143...
```python
plt.plot(Salary[0] , c = 'Green' ,ls = '--' , marker = 's' ,ms = 7 ,label = Play
plt.xticks(list(range(0,10)) , Seasons, rotation  = 'horizontal')
plt.show()
```
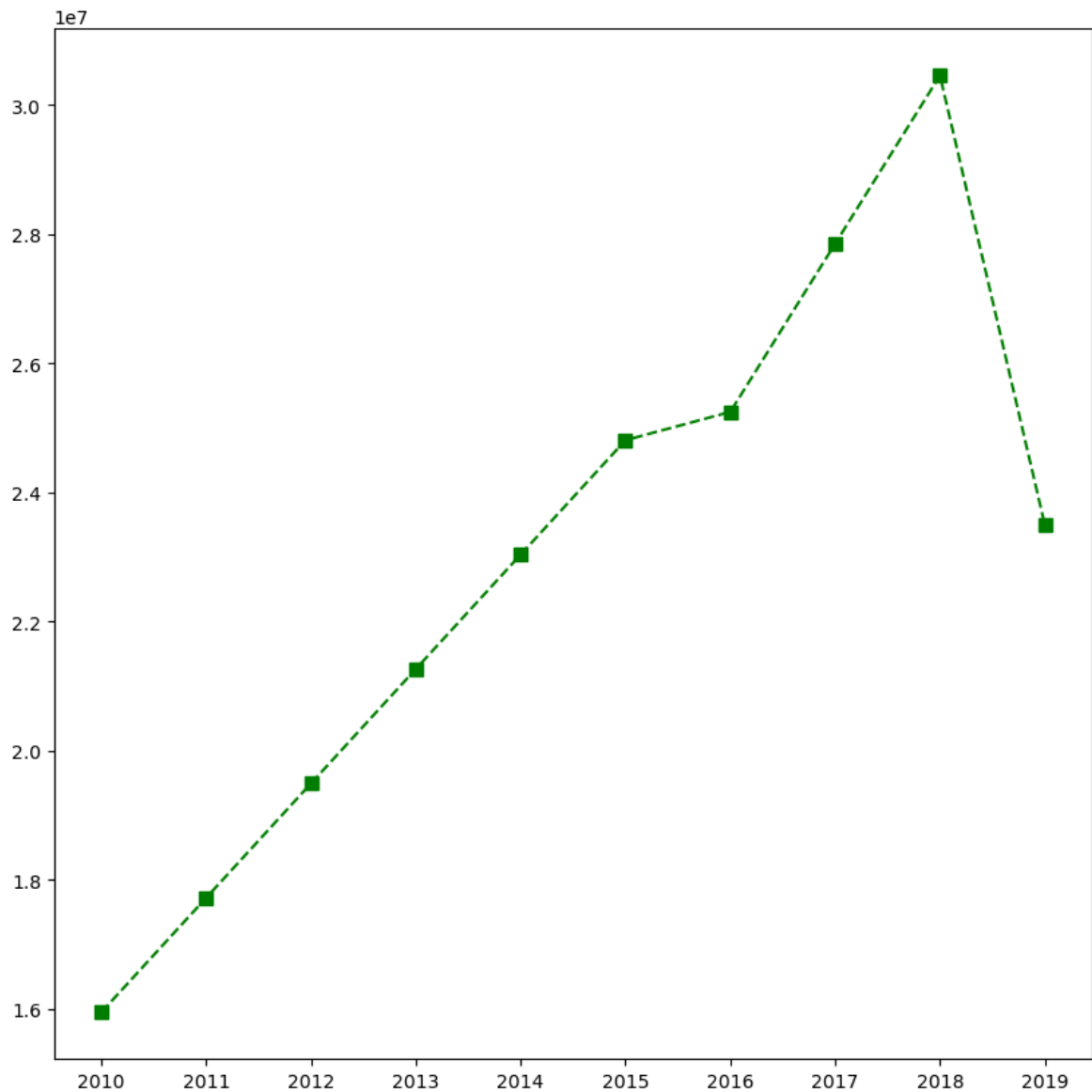
```
In [144…    Salary[0]
```

```
Out[144…    array([15946875, 17718750, 19490625, 21262500, 23034375, 24806250,
                   25244493, 27849149, 30453805, 23500000])
```

```
In [145…    Salary[0]
```

```
Out[145…    array([15946875, 17718750, 19490625, 21262500, 23034375, 24806250,
                   25244493, 27849149, 30453805, 23500000])
```

```
In [146…    plt.plot(Salary[1] , c = "Blue" , ls = ":" , marker = "o" , ms = 10 ,label =Play
            plt.show()
```
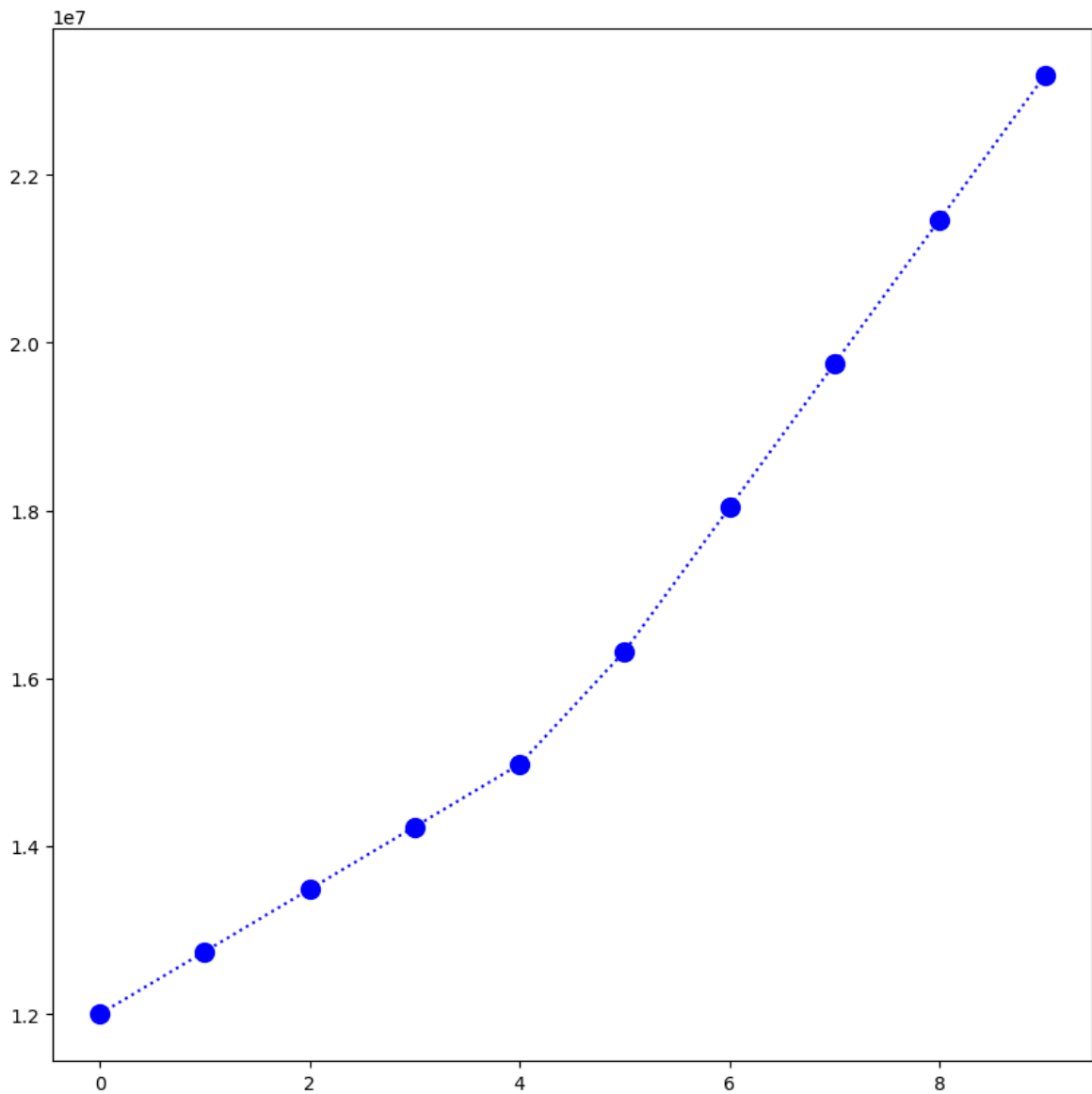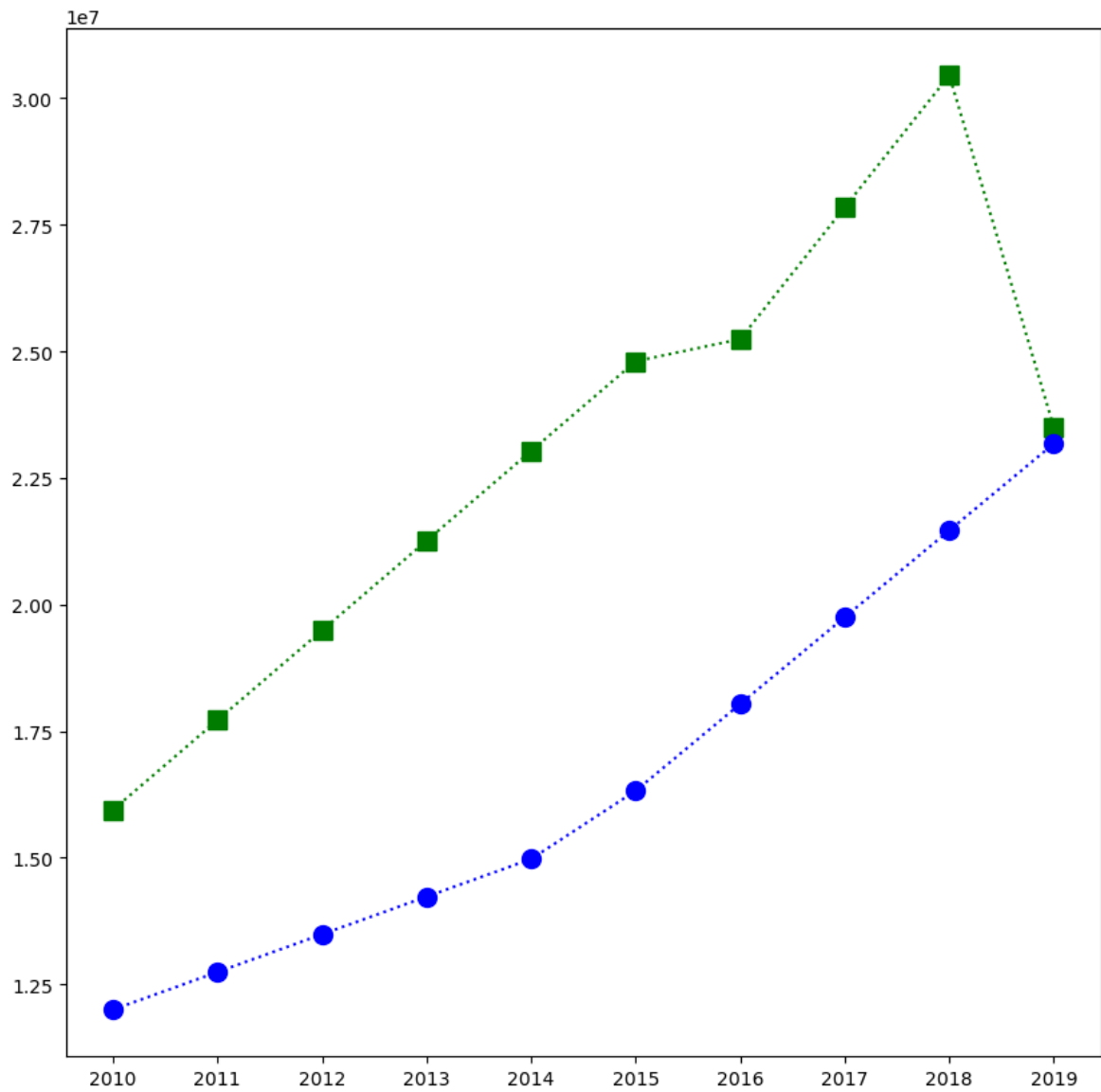
```
In [147…   # More visualization
```

```
In [148…   plt.plot(Salary[0] , c = 'Green' , ls = "dotted" , marker = "s" , ms = 10 , labe
           plt.plot(Salary[1] , c = "Blue" , ls = ":" ,marker = "o" , ms = 10 , label = Pla

           plt.xticks(list(range(0,10)) , Seasons ,rotation = 'horizontal')

           plt.show()
```

```
plt.plot(Salary[0] , c = 'Green' , ls = '--' ,marker = 's' ,ms = 7,label = Playe
plt.plot(Salary[1] , c = 'Blue' , ls = "--" , marker = "o" , ms = 5,label = Play

plt.plot(Salary[2] , c = 'purple' , ls = '--' , marker = '^' ,ms = 8 , label = P
plt.xticks(list(range(0,10)), Seasons , rotation = 'vertical')

plt.show()   # this .show we use for show the graph
```
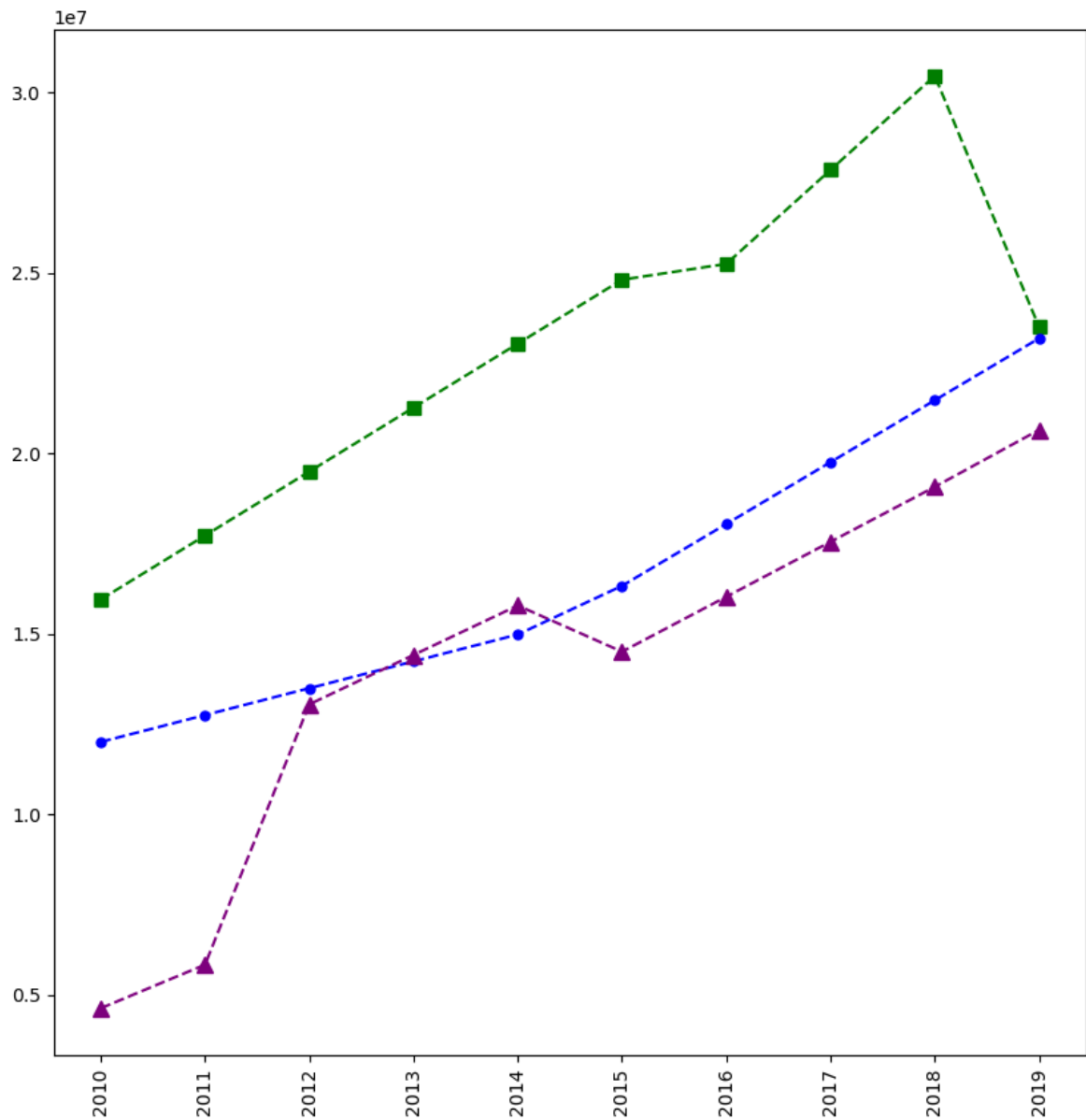
```
plt.plot(Salary[0], c = "Green" ,ls = "--" , marker = 's' , ms = 7 , label = Pla
plt.plot(Salary[1] , c = "Blue" , ls = '-' ,marker = 'o' ,ms = 8,label = Players
plt.plot(Salary[2] , c = "purple" , ls = "dotted" , marker = "^" ,ms =9 , label
plt.plot(Salary[3],c = "red" ,ls = ":" , marker = "d" ,ms = 8 , label = Players[
plt.legend()

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

plt.show()
```
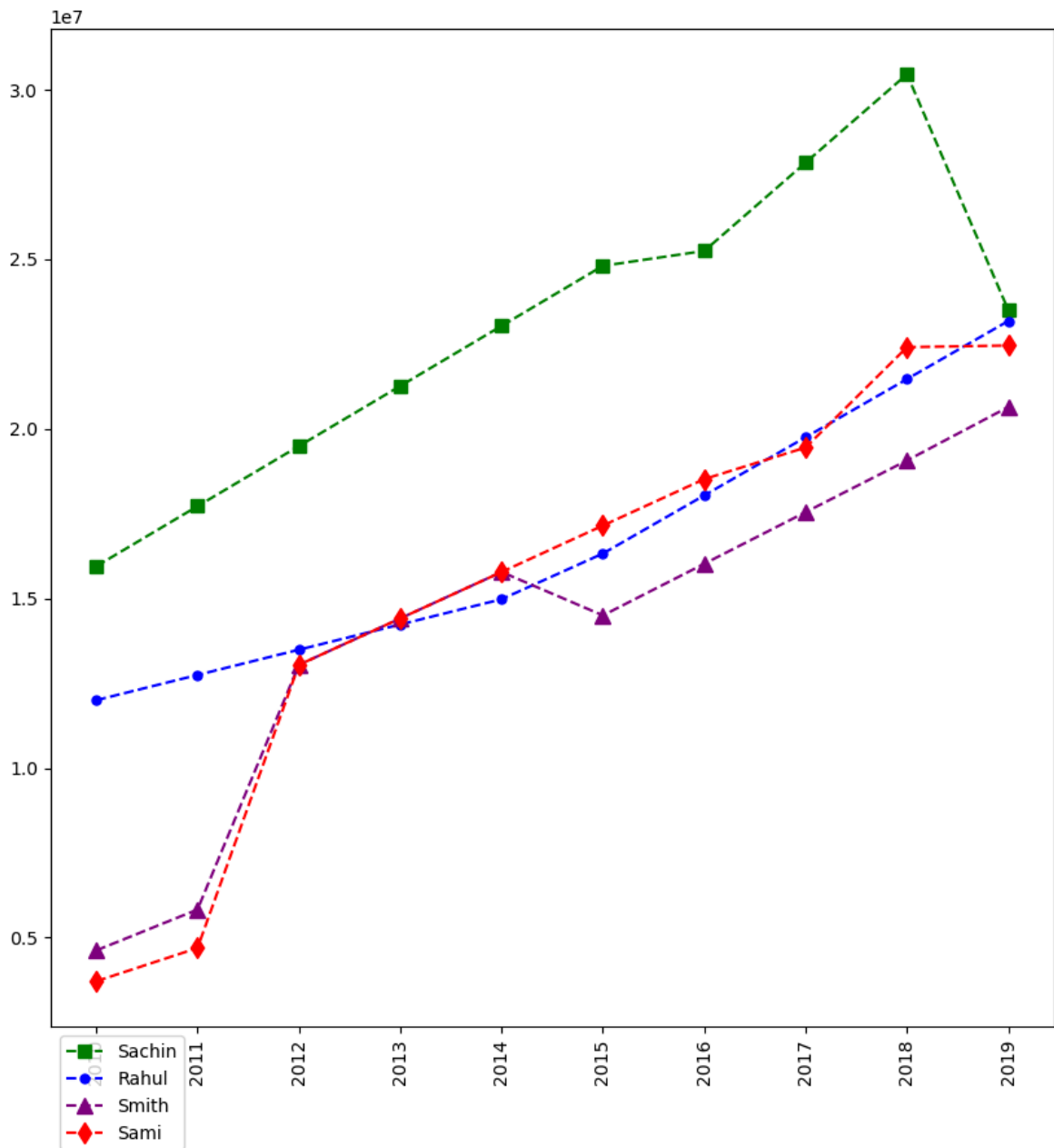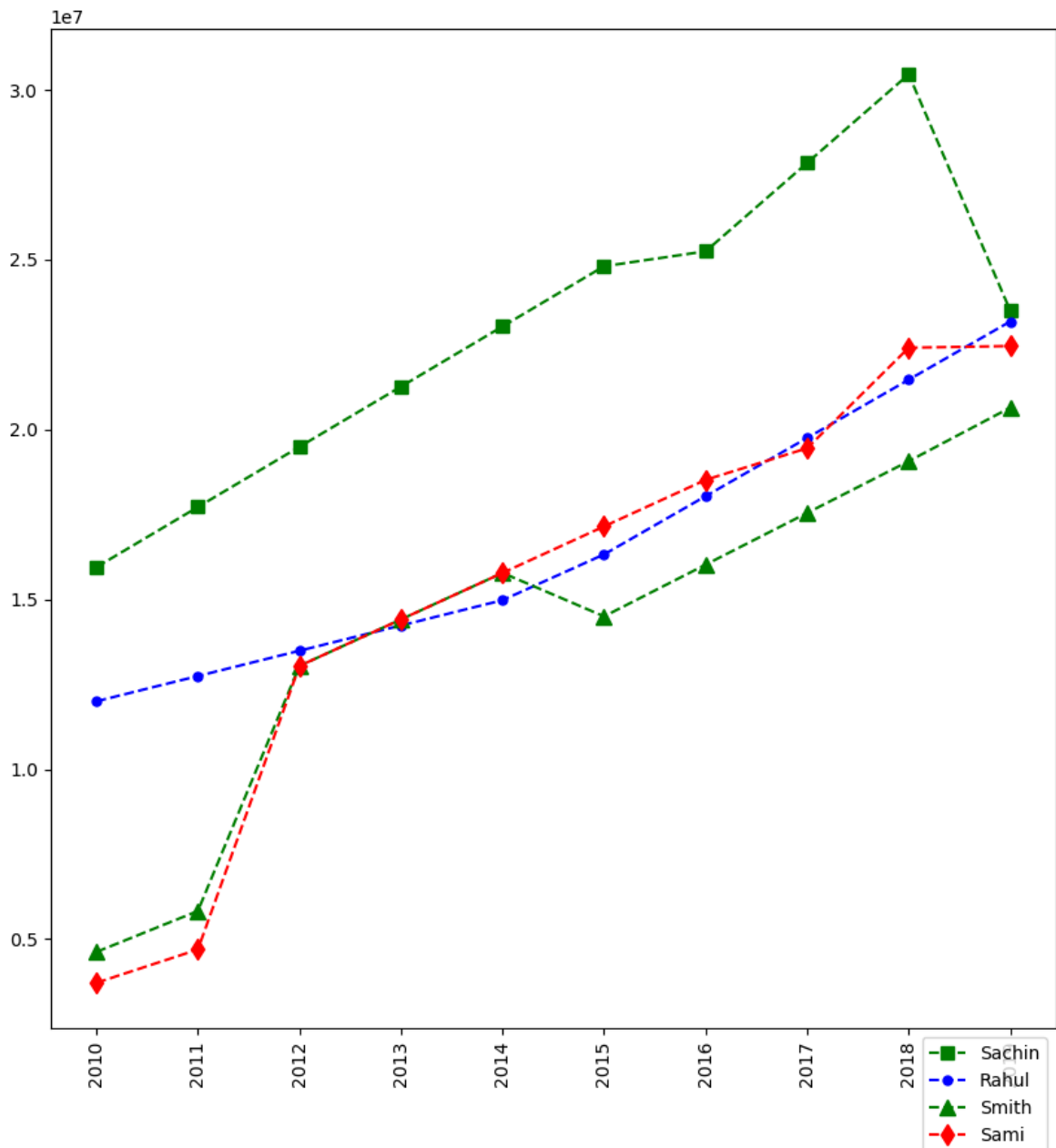
```
plt.plot(Salary[0], c='Green', ls = '--', marker = 's', ms = 7, label = Players[
plt.plot(Salary[1], c='Blue', ls = '--', marker = 'o', ms = 5, label = Players[1
plt.plot(Salary[2], c='purple', ls = '--', marker = '^', ms = 8, label = Players
plt.plot(Salary[3], c='Red', ls = '--', marker = 'd', ms = 8, label = Players[3]

plt.legend(loc = "upper left" , bbox_to_anchor = (0,0))
plt.xticks(list(range(0,10)),Seasons , rotation = "vertical")
plt.show()
```
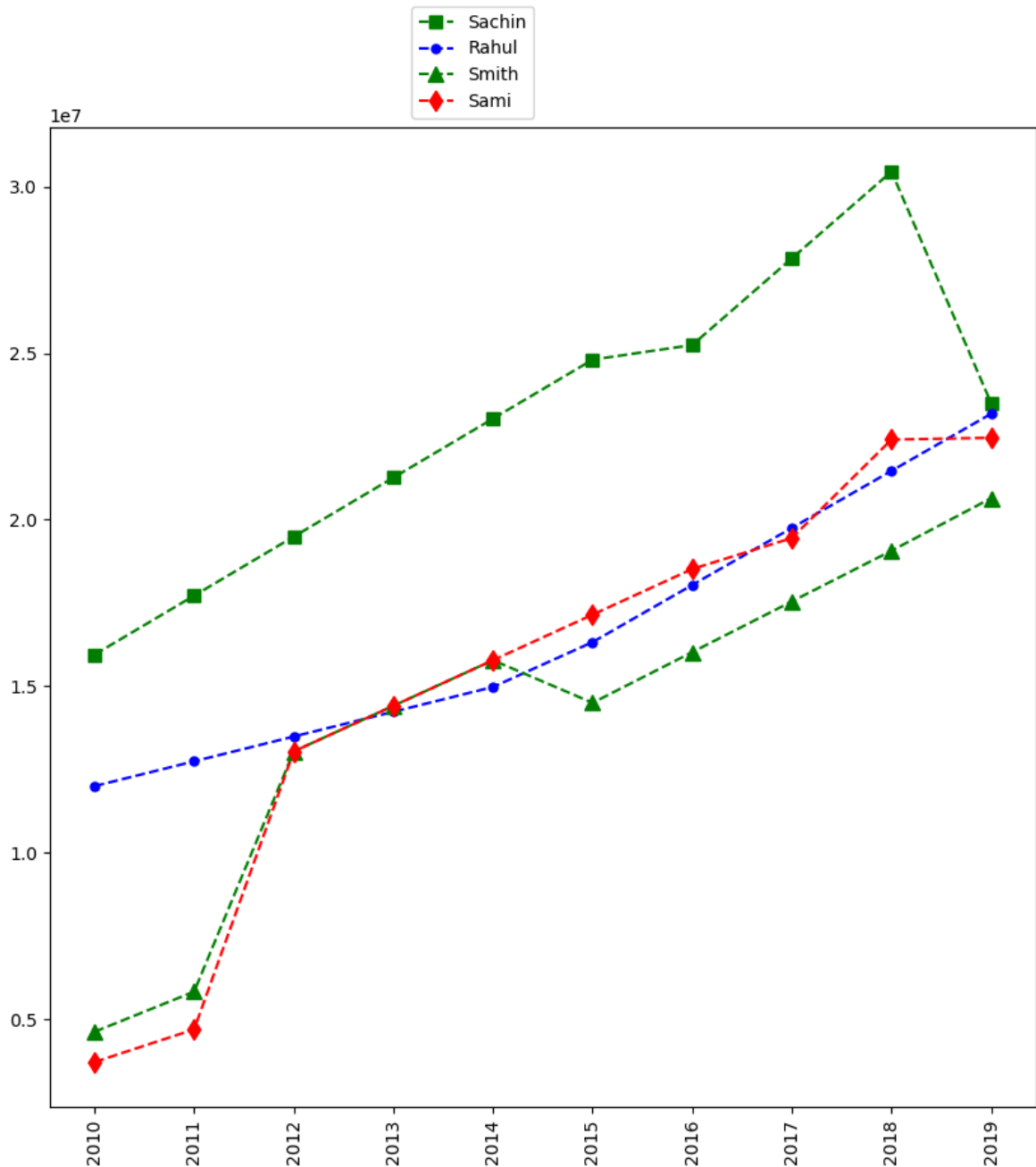
```
plt.plot(Salary[0], c='Green', ls = '--', marker = 's', ms = 7, label = Players[
plt.plot(Salary[1], c='Blue', ls = '--', marker = 'o', ms = 5, label = Players[1
plt.plot(Salary[2], c='Green', ls = '--', marker = '^', ms = 8, label = Players[
plt.plot(Salary[3], c='Red', ls = '--', marker = 'd', ms = 8, label = Players[3]
plt.legend(loc = 'upper right',bbox_to_anchor=(1,0) )
plt.xticks(list(range(0,10)), Seasons,rotation='vertical')

plt.show()
```

```
plt.plot(Salary[0], c='Green', ls = '--', marker = 's', ms = 7, label = Players[
plt.plot(Salary[1], c='Blue', ls = '--', marker = 'o', ms = 5, label = Players[1
plt.plot(Salary[2], c='Green', ls = '--', marker = '^', ms = 8, label = Players[
plt.plot(Salary[3], c='Red', ls = '--', marker = 'd', ms = 8, label = Players[3]
plt.legend(loc = 'lower right',bbox_to_anchor=(0.5,1) )
plt.xticks(list(range(0,10)), Seasons,rotation='vertical')

plt.show()
```
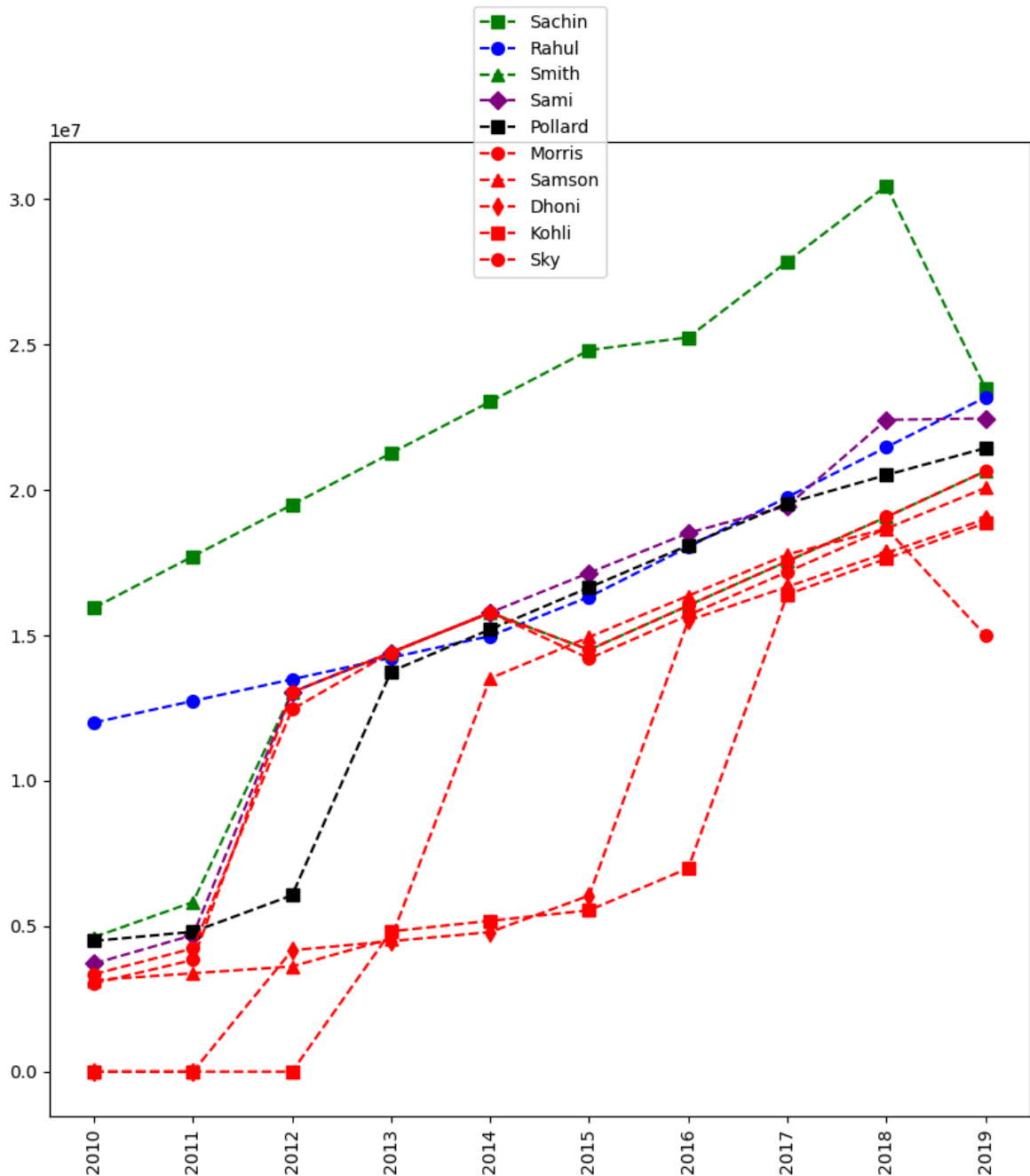
```
plt.plot(Salary[0], c='Green', ls = '--', marker = 's', ms = 7, label = Players[
plt.plot(Salary[1], c='Blue', ls = '--', marker = 'o', ms = 7, label = Players[1
plt.plot(Salary[2], c='Green', ls = '--', marker = '^', ms = 7, label = Players[
plt.plot(Salary[3], c='Purple', ls = '--', marker = 'D', ms = 7, label = Players
plt.plot(Salary[4], c='Black', ls = '--', marker = 's', ms = 7, label = Players[
plt.plot(Salary[5], c='Red', ls = '--', marker = 'o', ms = 7, label = Players[5]
plt.plot(Salary[6], c='Red', ls = '--', marker = '^', ms = 7, label = Players[6]
plt.plot(Salary[7], c='Red', ls = '--', marker = 'd', ms = 7, label = Players[7]
plt.plot(Salary[8], c='Red', ls = '--', marker = 's', ms = 7, label = Players[8]
plt.plot(Salary[9], c='Red', ls = '--', marker = 'o', ms = 7, label = Players[9]

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

plt.show()
```

```
In [180…    plt.plot(Games[0], c='Green', ls = '--', marker = 's', ms = 7, label = Players[0
            plt.plot(Games[1], c='Blue', ls = '--', marker = 'o', ms = 7, label = Players[1]
            plt.plot(Games[2], c='Green', ls = '--', marker = '^', ms = 7, label = Players[2
            plt.plot(Games[3], c='Red', ls = '--', marker = 'D', ms = 7, label = Players[3])
            plt.plot(Games[4], c='Black', ls = '--', marker = 's', ms = 7, label = Players[4
            plt.plot(Games[5], c='Blue', ls = '--', marker = 'o', ms = 7, label = Players[5]
            plt.plot(Games[6], c='red', ls = '--', marker = '^', ms = 7, label = Players[6])
            plt.plot(Games[7], c='Green', ls = '--', marker = 'd', ms = 7, label = Players[7
            plt.plot(Games[8], c='Red', ls = '--', marker = 's', ms = 7, label = Players[8])
            plt.plot(Games[9], c='Blue', ls = '--', marker = 'o', ms = 7, label = Players[9]


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

            plt.show()
```