

In [1]:

```
", "2014", "2015", "2016", "2017", "2018", "2019"]
2013":3, "2014":4, "2015":5, "2016":6, "2017":7, "2018":8, "2019":9}

Sami", "Pollard", "Morris", "Samson", "Dhoni", "Kohli", "Sky"]
:2, "Sami":3, "Pollard":4, "Morris":5, "Samson":6, "Dhoni":7, "Kohli":8, "Sky":9}

490625, 21262500, 23034375, 24806250, 25244493, 27849149, 30453805, 23500000]
88377, 14232567, 14976754, 16324500, 18038573, 19752645, 21466718, 23180790]
250, 14410581, 15779912, 14500000, 16022500, 17545000, 19067500, 20644400]
50, 14410581, 15779912, 17149243, 18518574, 19450000, 22407474, 22458000]
1274, 13758000, 15202590, 16647180, 18091770, 19536360, 20513178, 21436271]
5000, 14410581, 15779912, 14500000, 16022500, 17545000, 19067500, 20644400]
960, 4574189, 13520500, 14940153, 16359805, 17779458, 18668431, 20068563]
796880, 6053663, 15506632, 16669630, 17832627, 18995624]
, 5546160, 6993708, 16402500, 17632688, 18862875]
0, 14410581, 15779912, 14200000, 15691000, 17182000, 18673000, 15000000]

ul_Salary, Smith_Salary, Sami_Salary, Pollard_Salary, Morris_Salary, Samson_Salary, Dhoni_Salary, Kohli_Salary, Sky_Salary])

, 35]
, 80]
, 69]
40]
71, 41]
9, 44]
2, 82]
, 27]
51]
2]

Smith_G, Sami_G, Pollard_G, Morris_G, Samson_G, Dhoni_G, Kohli_G, Sky_G])

0, 2078, 1616, 2133, 83, 782]
, 1312, 1129, 1170, 1245, 1154]
, 2111, 1683, 2036, 2089, 1743]
1970, 1245, 1920, 2112, 966]
03, 1784, 1113, 1296, 1297, 646]
8, 1438, 1025, 1232, 1281, 928]
, 1268, 1189, 1186, 1185, 1564]
161, 1850, 2280, 2593, 686]
26, 852, 0, 159, 904]
941, 1082, 1463, 1028, 1331]

PTS, Smith_PTS, Sami_PTS, Pollard_PTS, Morris_PTS, Samson_PTS, Dhoni_PTS, Kohli_PTS, Sky_PTS])
```

In [2]: Salary

```
Out[2]: 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 [3]: Games

```
Out[3]: 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 [4]: Points
```

```
Out[4]: 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 [5]: Pdict
```

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

```
In [6]: Salary // Games
```

```
C:\Users\sidra\AppData\Local\Temp\ipykernel_27824\3536023082.py:1: RuntimeWarning: divide by zero encountered in floor_divide
Salary // Games
```

```
Out[6]: array([[ 199335,  230113,  237690,  259298,  315539,  302515,  435249,
  357040,  5075634,  671428],
 [ 146341,  223582,  164492,  180159,  197062,  226729,  300642,
  274342,  271730,  289759],
 [ 58503,  74719,  173883,  177908,  207630,  183544,  258427,
  230855,  247629,  299194],
 [ 46420,  72216,  169366,  218342,  228694,  222717,  336701,
  290298,  291006,  561450],
 [ 54794,  58618,  73917,  174151,  185397,  213425,  335032,
  257057,  288918,  522835],
 [ 47828,  61380,  185895,  187150,  225427,  188311,  281096,
  237094,  241360,  469190],
 [ 40310,  52815,  45199,  58643,  300455,  186751,  272663,
  253992,  301103,  244738],
 [ 0, 0, 52140, 60595, 58498, 77611, 234948,
  205797, 220155, 703541],
 [ 0, 0, 0, 59540, 66467, 68471, 179325,
  0, 1763268, 369860],
 [ 40425, 75322, 255710, 182412, 204933, 186842, 320224,
  249014, 345796, 241935]], dtype=int32)
```

```
In [7]: np.round(Salary // Games)
```

```
C:\Users\sidra\AppData\Local\Temp\ipykernel_27824\2034936389.py:1: RuntimeWarning: divide by zero encountered in floor_divide
np.round(Salary // Games)
```

```
Out[7]: array([[ 199335,  230113,  237690,  259298,  315539,  302515,  435249,
  357040,  5075634,  671428],
 [ 146341,  223582,  164492,  180159,  197062,  226729,  300642,
  274342,  271730,  289759],
 [ 58503,  74719,  173883,  177908,  207630,  183544,  258427,
  230855,  247629,  299194],
 [ 46420,  72216,  169366,  218342,  228694,  222717,  336701,
  290298,  291006,  561450],
 [ 54794,  58618,  73917,  174151,  185397,  213425,  335032,
  257057,  288918,  522835],
 [ 47828,  61380,  185895,  187150,  225427,  188311,  281096,
  237094,  241360,  469190],
 [ 40310,  52815,  45199,  58643,  300455,  186751,  272663,
  253992,  301103,  244738],
 [ 0, 0, 52140, 60595, 58498, 77611, 234948,
  205797, 220155, 703541],
 [ 0, 0, 0, 59540, 66467, 68471, 179325,
  0, 1763268, 369860],
 [ 40425, 75322, 255710, 182412, 204933, 186842, 320224,
  249014, 345796, 241935]], dtype=int32)
```

```
In [8]: import warnings
warnings.filterwarnings('ignore')
```

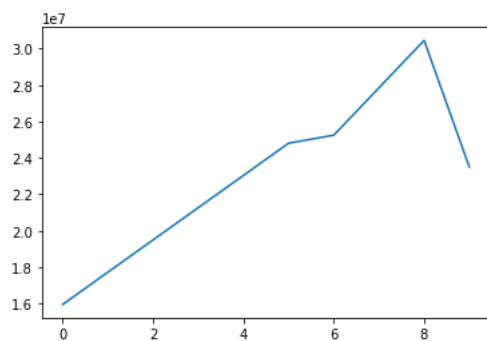
```
In [9]: import matplotlib.pyplot as plt # Library used for visualization
```

```
In [10]: Salary[0]
```

```
Out[10]: array([15946875, 17718750, 19490625, 21262500, 23034375, 24806250,  
                25244493, 27849149, 30453805, 23500000])
```

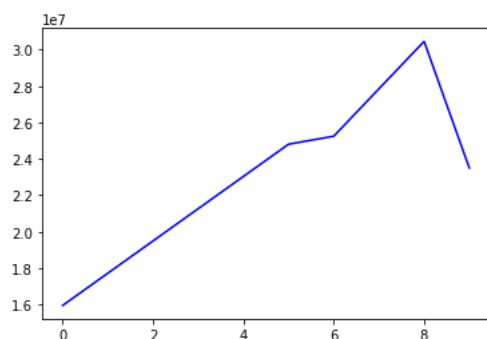
```
In [11]: plt.plot(Salary[0])
```

```
Out[11]: [<matplotlib.lines.Line2D at 0x19b7d00b070>]
```



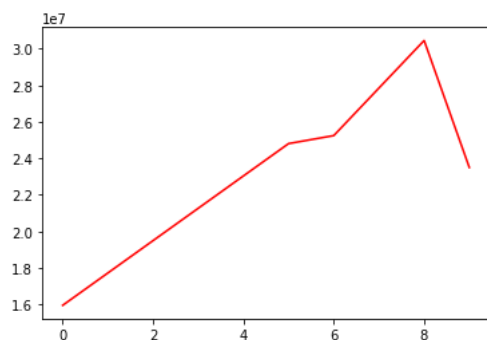
```
In [12]: plt.plot(Salary[0], c = 'b')
```

```
Out[12]: [<matplotlib.lines.Line2D at 0x19b7d798580>]
```



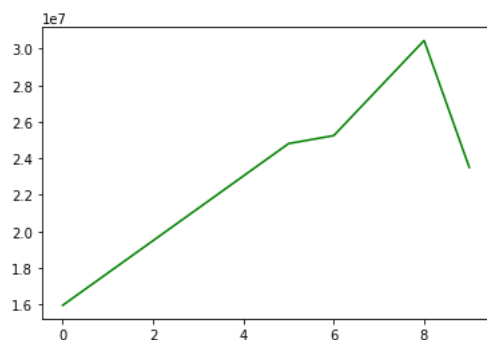
```
In [13]: plt.plot(Salary[0], c = 'r')
```

```
Out[13]: [<matplotlib.lines.Line2D at 0x19b7d80d0a0>]
```



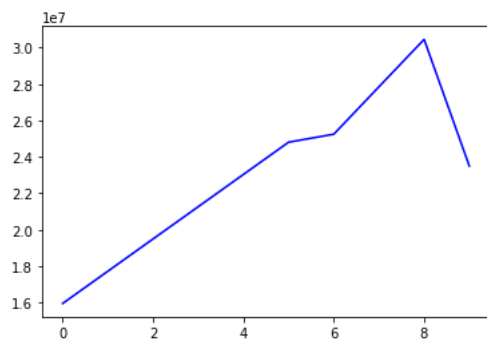
```
In [14]: plt.plot(Salary[0], c = 'g')
```

```
Out[14]: [<matplotlib.lines.Line2D at 0x19b7d869880>]
```



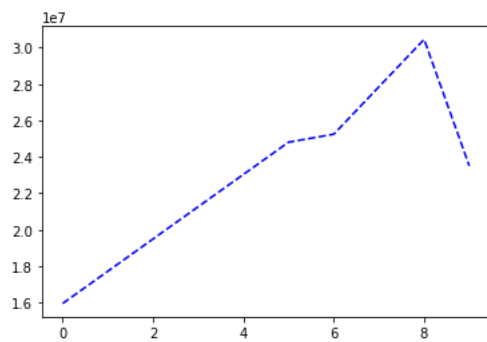
```
In [15]: plt.plot(Salary[0], c = 'b')
```

```
Out[15]: [<matplotlib.lines.Line2D at 0x19b7d8cad30>]
```



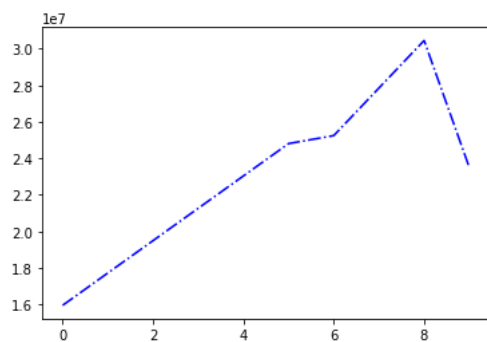
```
In [16]: plt.plot(Salary[0], c = 'b' , ls = '--')
```

```
Out[16]: [<matplotlib.lines.Line2D at 0x19b7d93e1c0>]
```



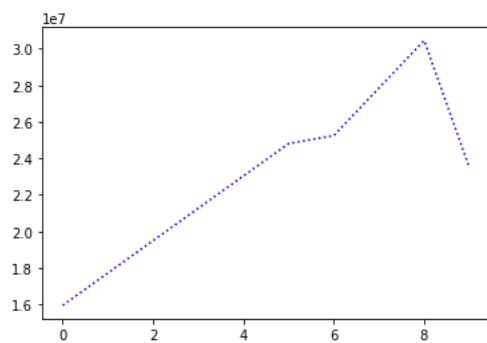
```
In [17]: plt.plot(Salary[0], c = 'b', ls = '-.-')
```

```
Out[17]: [<matplotlib.lines.Line2D at 0x19b7d99b880>]
```



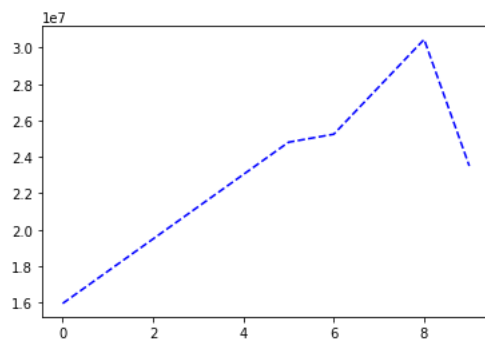
```
In [18]: plt.plot(Salary[0], c = 'b', ls = ':')
```

```
Out[18]: [<matplotlib.lines.Line2D at 0x19b7da0b0a0>]
```



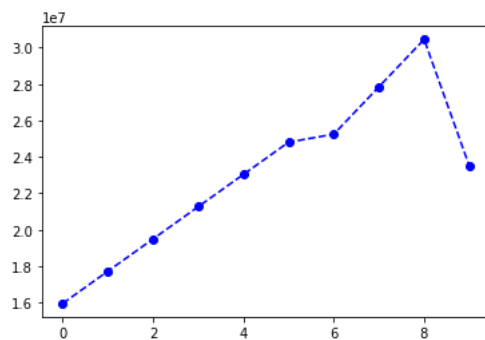
```
In [19]: plt.plot(Salary[0], c = 'b', ls = '--')
```

```
Out[19]: [<matplotlib.lines.Line2D at 0x19b7ea3d370>]
```



```
In [20]: plt.plot(Salary[0], c = 'b', ls = '--', marker = 'o')
```

```
Out[20]: [<matplotlib.lines.Line2D at 0x19b7ea9d910>]
```



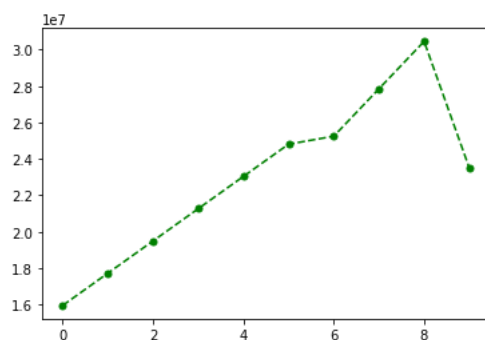
```
In [21]: Games[0]
```

```
Out[21]: array([80, 77, 82, 82, 73, 82, 58, 78, 6, 35])
```

```
In [22]: %matplotlib inline # to minimize the size of the graph  
plt.rcParams['figure.figsize'] = 7,3
```

UsageError: unrecognized arguments: # to minimize the size of the graph

```
In [23]: plt.plot(Salary[0], c = 'g', ls = '--', marker = 'o', ms = 5)  
plt.show()
```



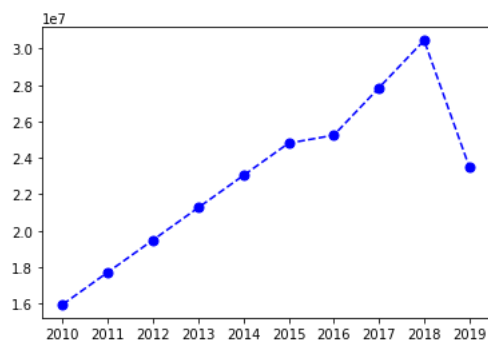
```
In [24]: Sdict
```

```
Out[24]: {'2010': 0,  
'2011': 1,  
'2012': 2,  
'2013': 3,  
'2014': 4,  
'2015': 5,  
'2016': 6,  
'2017': 7,  
'2018': 8,  
'2019': 9}
```

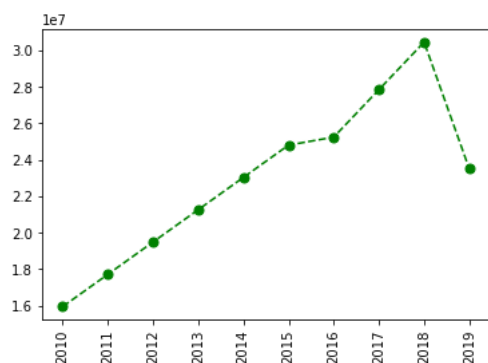
```
In [25]: Pdict
```

```
Out[25]: {'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 = 'b', ls = '--', marker = 'o', ms = 7)  
plt.xticks(list(range(0,10)), Seasons) # adding the year(Seasons)  
plt.show()
```



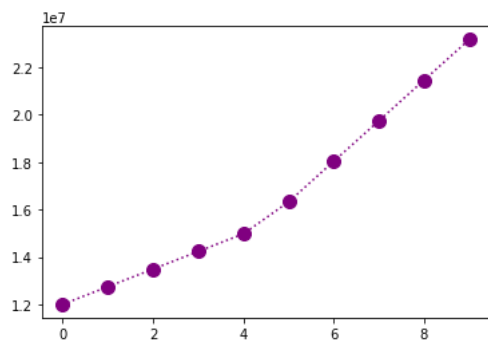
```
In [27]: plt.plot(Salary[0], c = 'g', ls = '--', marker = 'o', ms = 7)  
plt.xticks(list(range(0,10)), Seasons, rotation = 'vertical')  
plt.show()
```



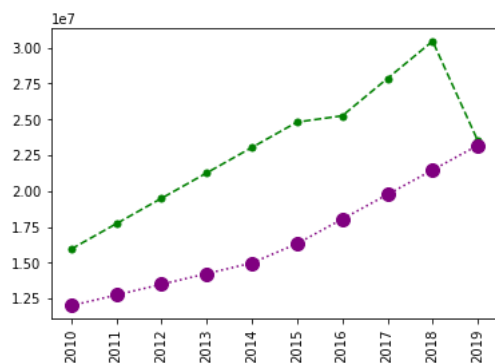
```
In [28]: Salary[1]
```

```
Out[28]: array([12000000, 12744189, 13488377, 14232567, 14976754, 16324500,  
               18038573, 19752645, 21466718, 23180790])
```

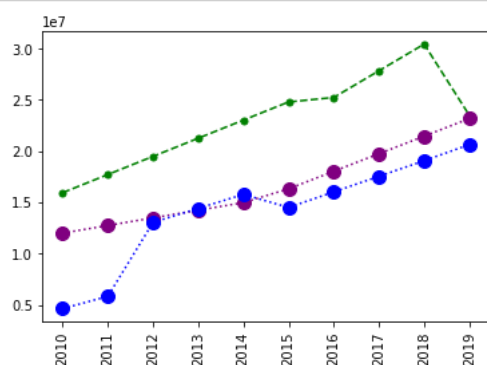
```
In [29]: plt.plot(Salary[1], c = 'purple', ls = ':', marker = 'o', ms = 10)  
plt.show()
```



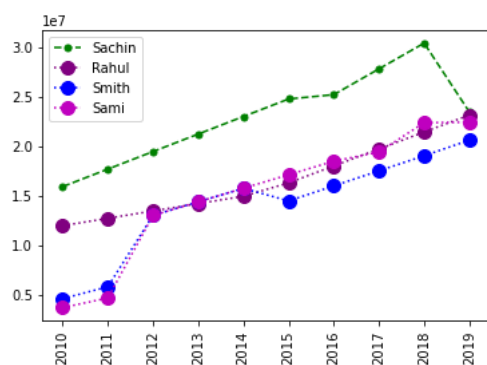
```
In [30]: plt.plot(Salary[0], c = 'g', ls = '--', marker = 'o', ms = 5)
plt.plot(Salary[1], c = 'purple', ls = ':', marker = 'o', ms = 10)
plt.xticks(list(range(0,10)), Seasons, rotation = 'vertical')
plt.show()
```



```
In [31]: plt.plot(Salary[0], c = 'g', ls = '--', marker = 'o', ms = 5)
plt.plot(Salary[1], c = 'purple', ls = ':', marker = 'o', ms = 10)
plt.plot(Salary[2], c = 'b', ls = ':', marker = 'o', ms = 10)
plt.xticks(list(range(0,10)), Seasons, rotation = 'vertical')
plt.show()
```

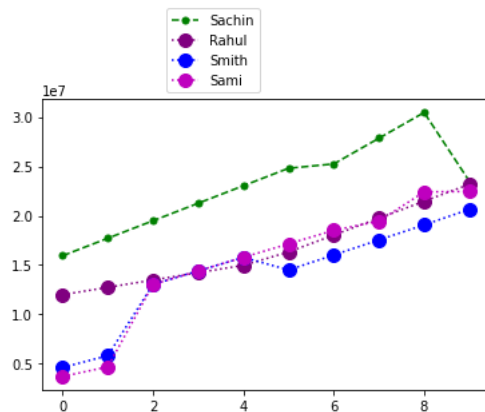


```
In [32]: plt.plot(Salary[0], c = 'g', ls = '--', marker = 'o', ms = 5, label= Players[0])
plt.plot(Salary[1], c = 'purple', ls = ':', marker = 'o', ms = 10, label= Players[1])
plt.plot(Salary[2], c = 'b', ls = ':', marker = 'o', ms = 10, label= Players[2])
plt.plot(Salary[3], c = 'm', ls = ':', marker = 'o', ms = 10, label= Players[3])
plt.legend() # Automatically it will create a color for the players
plt.xticks(list(range(0,10)), Seasons, rotation = 'vertical')
plt.show()
```



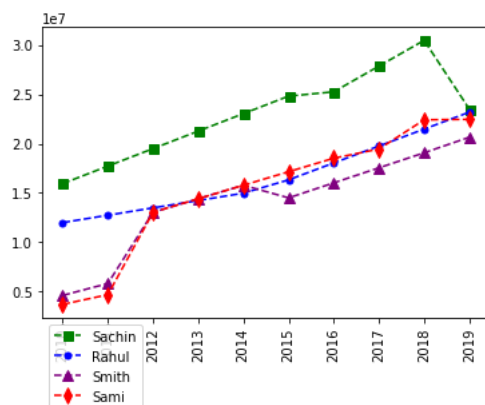
```
In [33]: plt.plot(Salary[0], c = 'g', ls = '--', marker = 'o', ms = 5, label= Players[0])
plt.plot(Salary[1], c = 'purple', ls = ':', marker = 'o', ms = 10, label= Players[1])
plt.plot(Salary[2], c = 'b', ls = ':', marker = 'o', ms = 10, label= Players[2])
plt.plot(Salary[3], c = 'm', ls = ':', marker = 'o', ms = 10, label= Players[3])
plt.legend(loc = 'lower right', bbox_to_anchor=(0.5, 1))
```

Out[33]: <matplotlib.legend.Legend at 0x19b7eaac970>



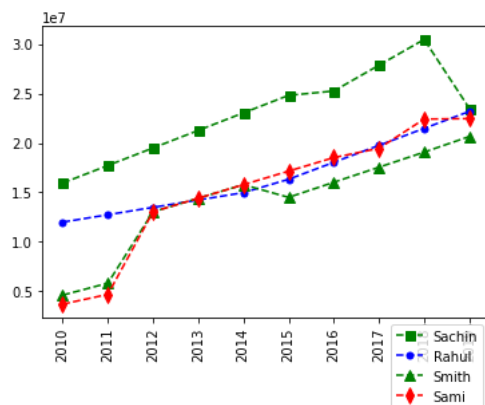
```
In [34]: plt.plot(Salary[0], c='Green', ls = '--', marker = 's', ms = 7, label = Players[0])
plt.plot(Salary[1], c='Blue', ls = '--', marker = 'o', ms = 5, label = Players[1])
plt.plot(Salary[2], c='purple', ls = '--', marker = '^', ms = 8, label = Players[2])
plt.plot(Salary[3], c='Red', ls = '--', 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()
```



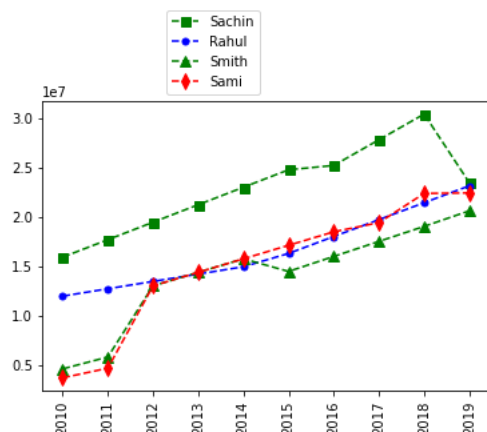
```
In [35]: plt.plot(Salary[0], c='Green', ls = '--', marker = 's', ms = 7, label = Players[0])
plt.plot(Salary[1], c='Blue', ls = '--', marker = 'o', ms = 5, label = Players[1])
plt.plot(Salary[2], c='Green', ls = '--', marker = '^', ms = 8, label = Players[2])
plt.plot(Salary[3], c='Red', ls = '--', 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()
```




```
In [36]: plt.plot(Salary[0], c='Green', ls = '--', marker = 's', ms = 7, label = Players[0])
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[2])
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')

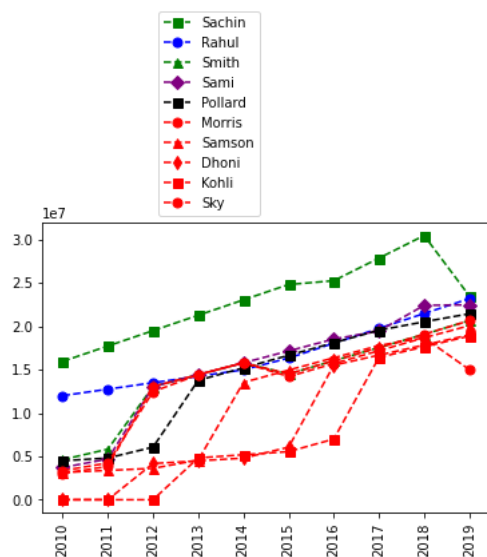
plt.show()
```



```
In [38]: plt.plot(Salary[0], c='Green', ls = '--', marker = 's', ms = 7, label = Players[0])
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[2])
plt.plot(Salary[3], c='Purple', ls = '--', marker = 'D', ms = 7, label = Players[3])
plt.plot(Salary[4], c='Black', ls = '--', marker = 's', ms = 7, label = Players[4])
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 = 'lower right',bbox_to_anchor=(0.5,1) )
plt.xticks(list(range(0,10)), Seasons,rotation='vertical')

plt.show()
```

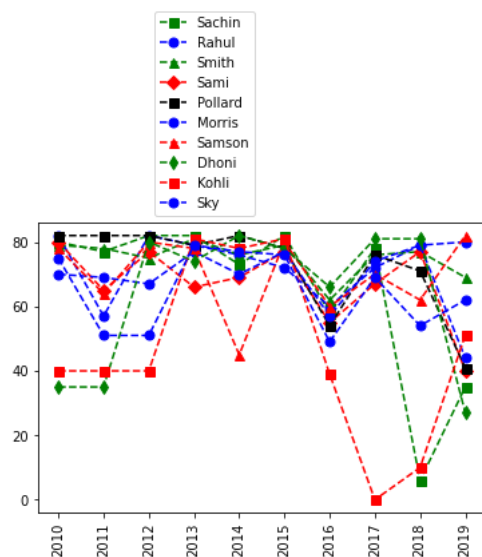


```
In [39]: # we can visualize the how many games played by a player
```

```
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()
```



```
In [ ]: # when the graph is limited then we use BI
```