

IPL Data Analysis Project

We will analyze the players

- Here, we have taken top 10 highest paid player in 2015-2024 season.
- We will analyze how 10 players have been playing over the past 10 years & we had the data for past 10yrs.
- Our main goal is to find trends, patterns & their performance for the past 10 yrs.

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

#Seasons
Seasons = ["2015", "2016", "2017", "2018", "2019", "2020", "2021", "2022", "2023", "2024"]
Sdict = {"2015":0, "2016":1, "2017":2, "2018":3, "2019":4, "2020":5, "2021":6, "2022":7, "2023":8, "2024":9}

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

#Salaries
Sachin_Salary = [15946875, 17718750, 19490625, 21262500, 23034375, 24806250, 25244493, 27850000, 30500000, 33500000, 36500000, 39500000, 42500000, 45500000, 48500000, 51500000, 54500000]
Rahul_Salary = [12000000, 12744189, 13488377, 14232567, 14976754, 16324500, 18038573, 19750000, 21500000, 23200000, 25000000, 26700000, 28400000, 30100000, 31800000, 33500000, 35200000]
Smith_Salary = [4621800, 5828090, 13041250, 14410581, 15779912, 14500000, 16022500, 17545000, 19060000, 20580000, 22100000, 23620000, 25140000, 26660000, 28180000, 29700000, 31220000, 32740000]
Sami_Salary = [3713640, 4694041, 13041250, 14410581, 15779912, 17149243, 18518574, 19450000, 20930000, 22410000, 23890000, 25370000, 26850000, 28330000, 29810000, 31290000, 32770000, 34250000]
Pollard_Salary = [4493160, 4806720, 6061274, 13758000, 15202590, 16647180, 18091770, 19536000, 21080000, 22620000, 24160000, 25700000, 27240000, 28780000, 30320000, 31860000, 33400000, 34940000]
Morris_Salary = [3348000, 4235220, 12455000, 14410581, 15779912, 14500000, 16022500, 17545000, 19060000, 20580000, 22100000, 23620000, 25140000, 26660000, 28180000, 29700000, 31220000, 32740000, 34260000]
Samson_Salary = [3144240, 3380160, 3615960, 4574189, 13520500, 14940153, 16359805, 17779450, 19200000, 20720000, 22240000, 23760000, 25280000, 26800000, 28320000, 29840000, 31360000, 32880000, 34400000]
Dhoni_Salary = [0, 0, 4171200, 4484040, 4796880, 6053663, 15506632, 16669630, 17832627, 18990000, 20510000, 22030000, 23550000, 25070000, 26590000, 28110000, 29630000, 31150000, 32670000, 34190000]
Kohli_Salary = [0, 0, 0, 4822800, 5184480, 5546160, 6993708, 16402500, 17632688, 18862875]
Sky_Salary = [3031920, 3841443, 13041250, 14410581, 15779912, 14200000, 15691000, 17182000]

#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_
```

In [2]: *Salary #matrix format*

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, 4822800, 5184480, 5546160,
 6993708, 16402500, 17632688, 18862875],
 [3031920, 3841443, 13041250, 14410581, 15779912, 14200000,
 15691000, 17182000, 18673000, 15000000]])

In [3]: *#Building first matrix*
 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]: Games[5] #5th row
```

```
Out[5]: array([70, 69, 67, 77, 70, 77, 57, 74, 79, 44])
```

```
In [6]: Games[0:5]
```

```
Out[6]: 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 [7]: Games
```

```
Out[7]: 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 [8]: Games[0,5]
```

```
Out[8]: 82
```

```
In [9]: Games[-3:-1]
```

```
Out[9]: array([[35, 35, 80, 74, 82, 78, 66, 81, 81, 27],  
               [40, 40, 40, 81, 78, 81, 39, 0, 10, 51]])
```

```
In [10]: Points
```

```
Out[10]: 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 [11]: Points[0]

```
Out[11]: array([2832, 2430, 2323, 2201, 1970, 2078, 1616, 2133, 83, 782])
```

In [12]: Points[:]

```
Out[12]: 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 [13]: Games

```
Out[13]: 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 [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 [15]: Pdict['Rahul']

```
Out[15]: 1
```

```
In [16]: Games[1]
```

```
Out[16]: array([82, 57, 82, 79, 76, 72, 60, 72, 79, 80])
```

```
In [17]: Games[Pdict['Rahul']]
```

```
Out[17]: array([82, 57, 82, 79, 76, 72, 60, 72, 79, 80])
```

Games

```
In [19]: Points
```

```
Out[19]: 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 [20]: Salary
```

```
Out[20]: 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 [21]: Games
```

```
Out[21]: 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 [22]: np.round(Salary/Games)
```

C:\Users\SUNITHA\AppData\Local\Temp\ipykernel_23288\3232172828.py:1: RuntimeWarning:
divide by zero encountered in divide
np.round(Salary/Games)

```
Out[22]: 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.]])
```

```
In [28]: import warnings
warnings.filterwarnings('ignore')
```

```
In [30]: import matplotlib.pyplot as plt #for visualisation
```

```
In [32]: %matplotlib inline #to keep the plot inside the jupyter notes instead of getting in
```

UsageError: unrecognized arguments: #to keep the plot inside the jupyter notes instead of getting in other screen

```
In [34]: Salary
```

```
Out[34]: 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]])
```

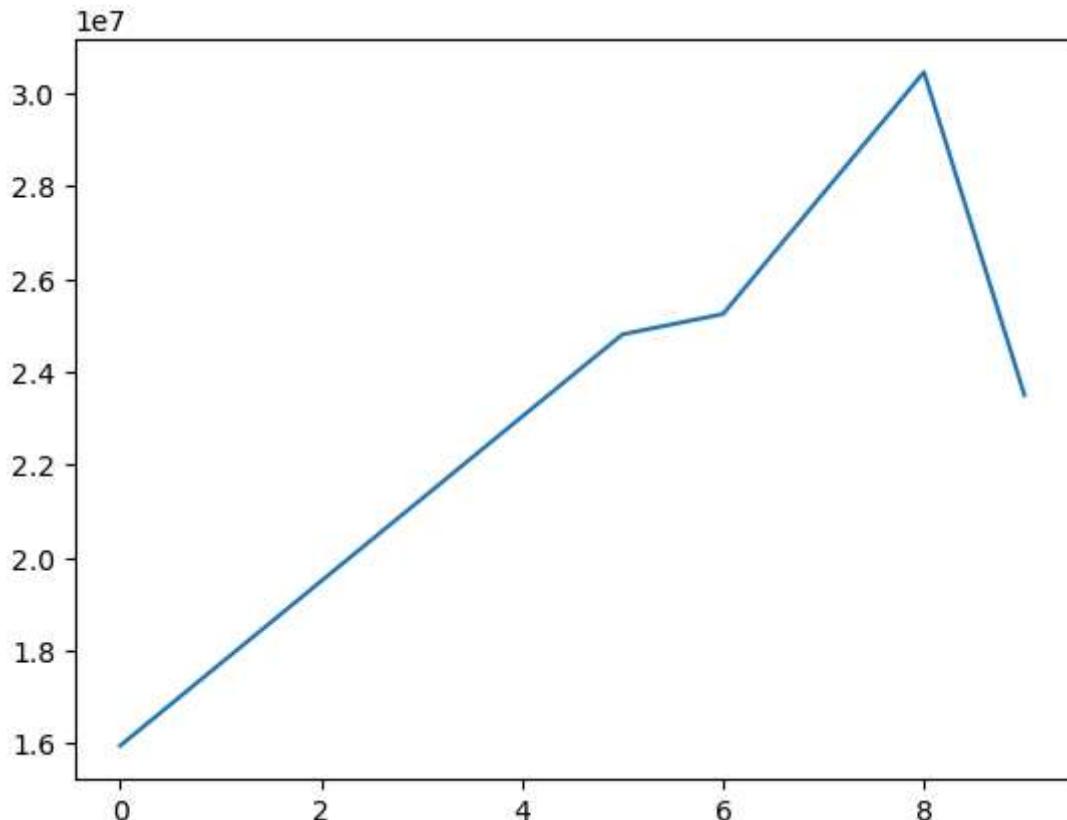
In [36]: `Salary[0]`

```
Out[36]: array([15946875, 17718750, 19490625, 21262500, 23034375, 24806250,
   25244493, 27849149, 30453805, 23500000])
```

In [38]: `plt.plot(Salary[0])`

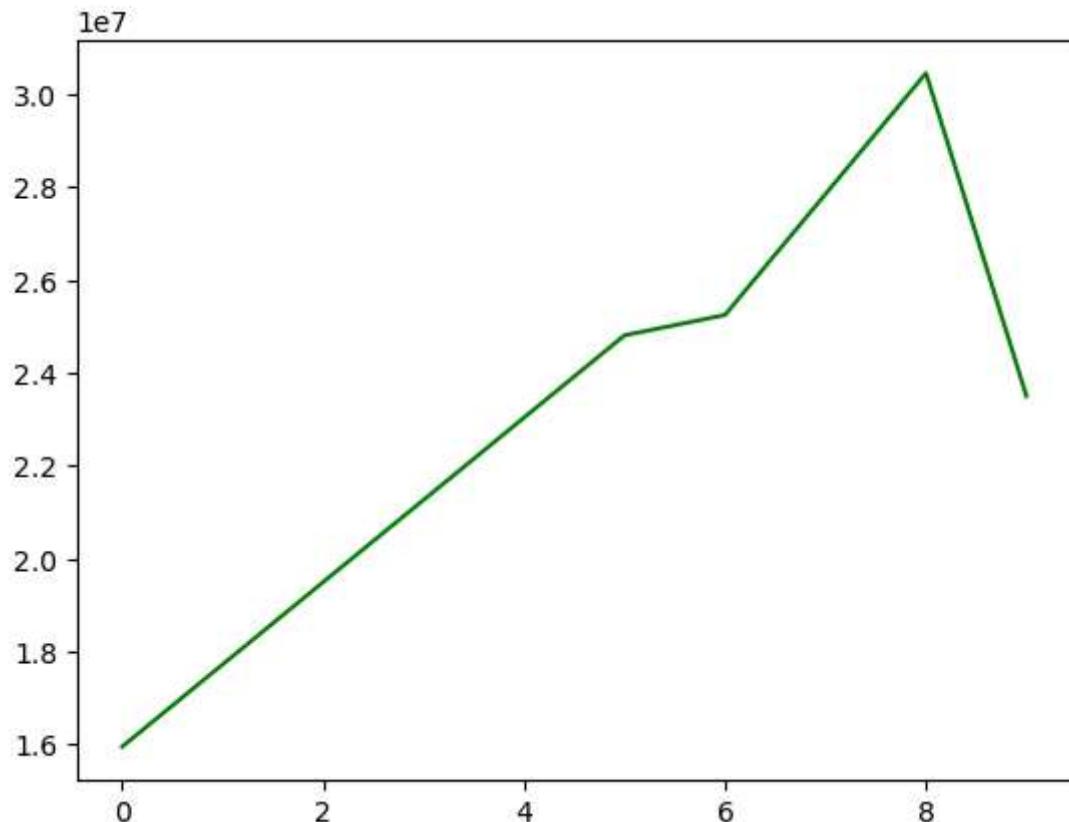
#As per the salary report of (Salary[0] person) i.e. Sachin salary has been increased

```
Out[38]: [<matplotlib.lines.Line2D at 0x160e9520c80>]
```



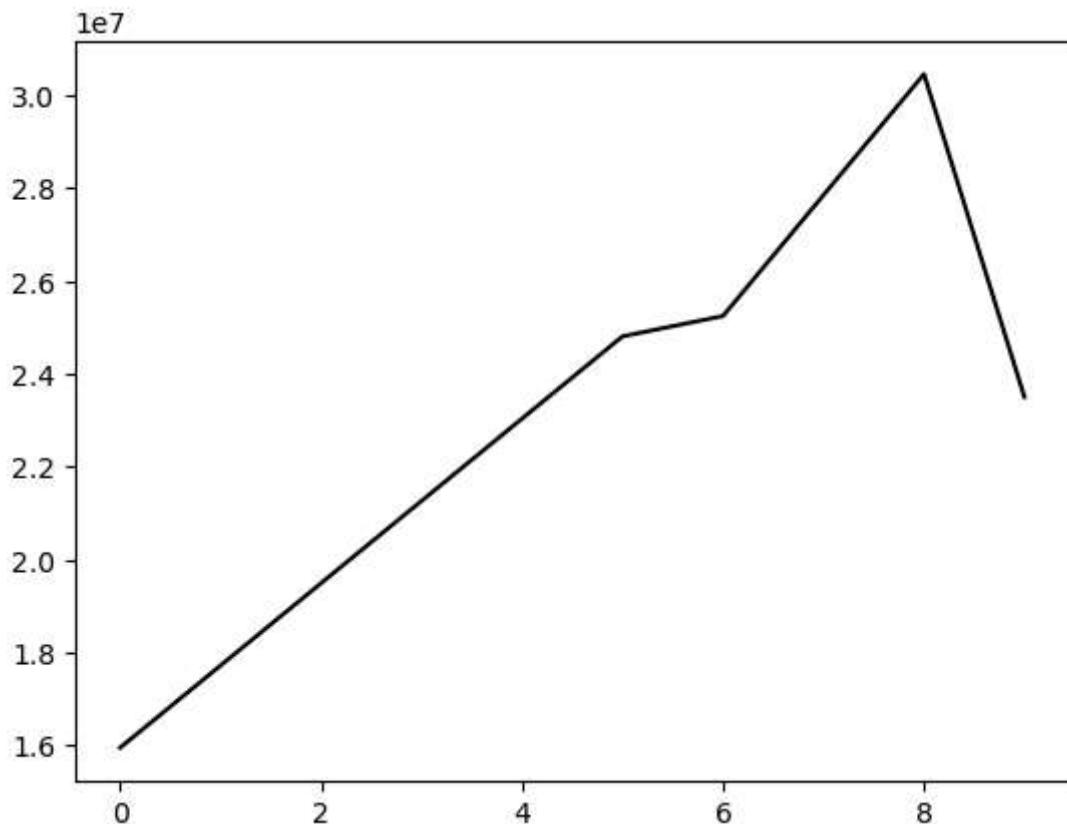
```
In [40]: plt.plot(Salary[0], c ='green')
```

```
Out[40]: [<matplotlib.lines.Line2D at 0x160ed196780>]
```



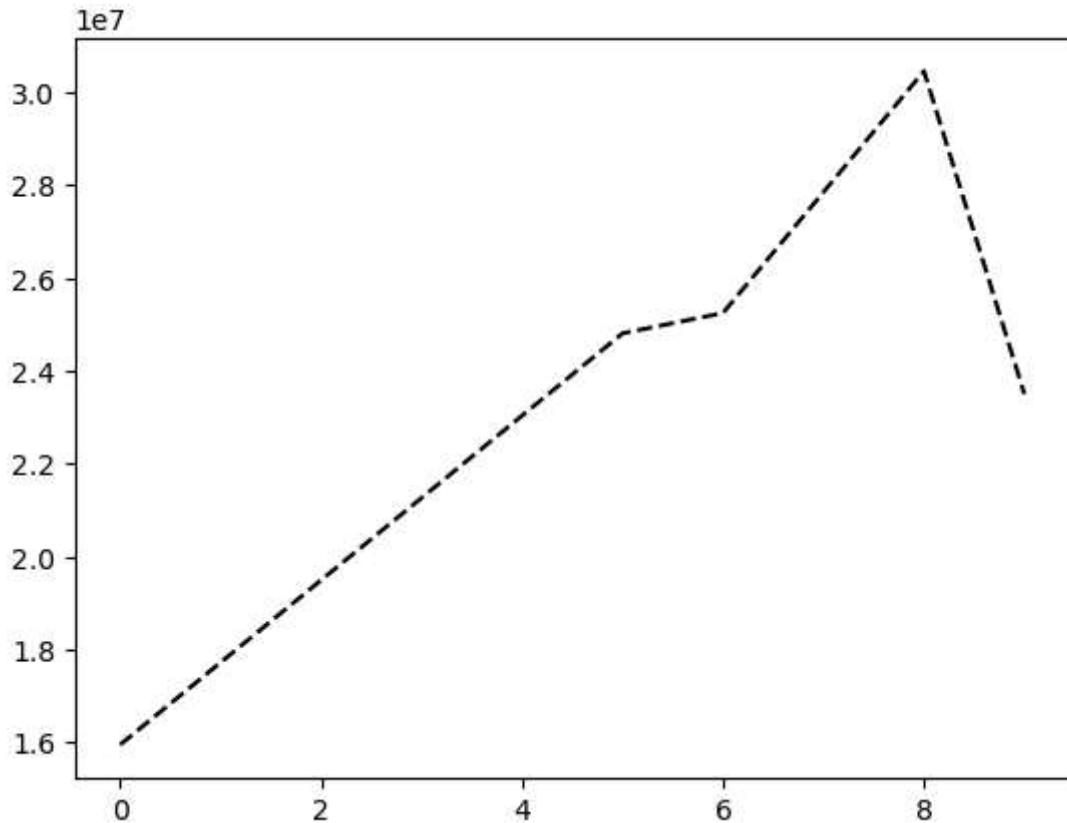
```
In [42]: plt.plot(Salary[0], color ='k')
```

```
Out[42]: [<matplotlib.lines.Line2D at 0x160ed212330>]
```



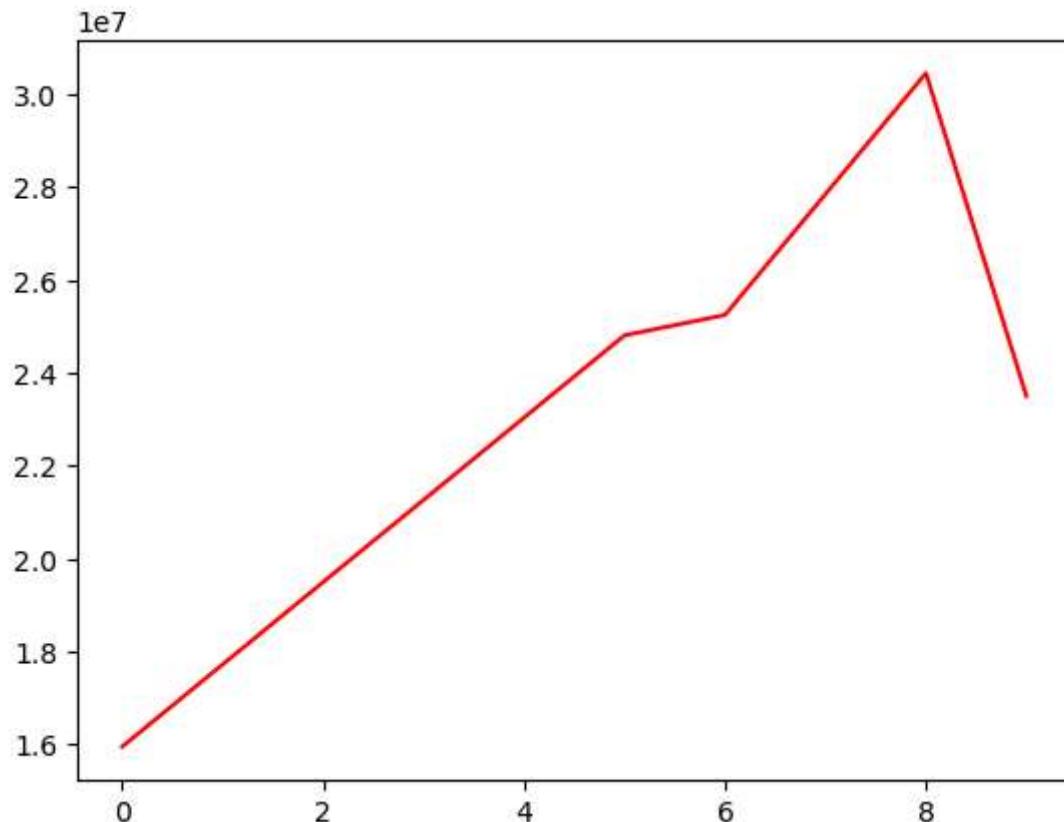
```
In [44]: plt.plot(Salary[0], color ='black', ls = '--')
# plt.plot(Salary[0], c ='k', ls = '--') #we can use this also
```

```
Out[44]: [<matplotlib.lines.Line2D at 0x160ed273290>]
```



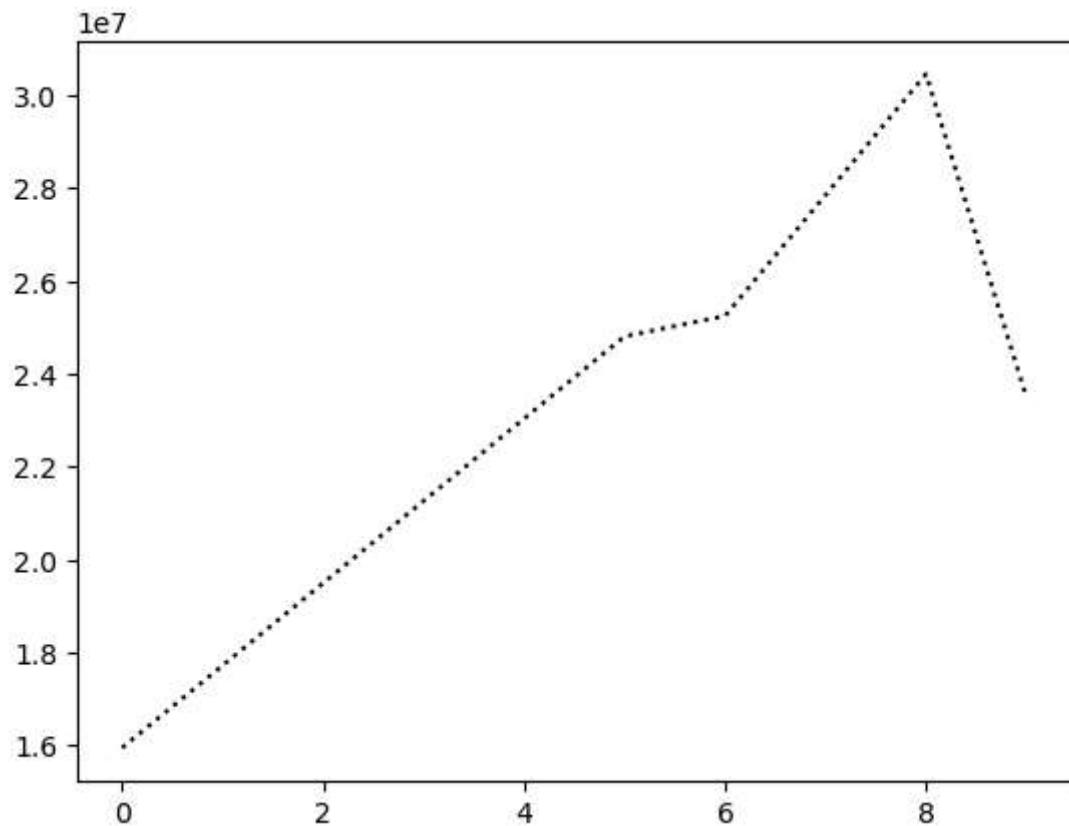
```
In [46]: plt.plot(Salary[0], c ='r') #the below pattern is inclined pattern
```

```
Out[46]: [<matplotlib.lines.Line2D at 0x160ed31c410>]
```



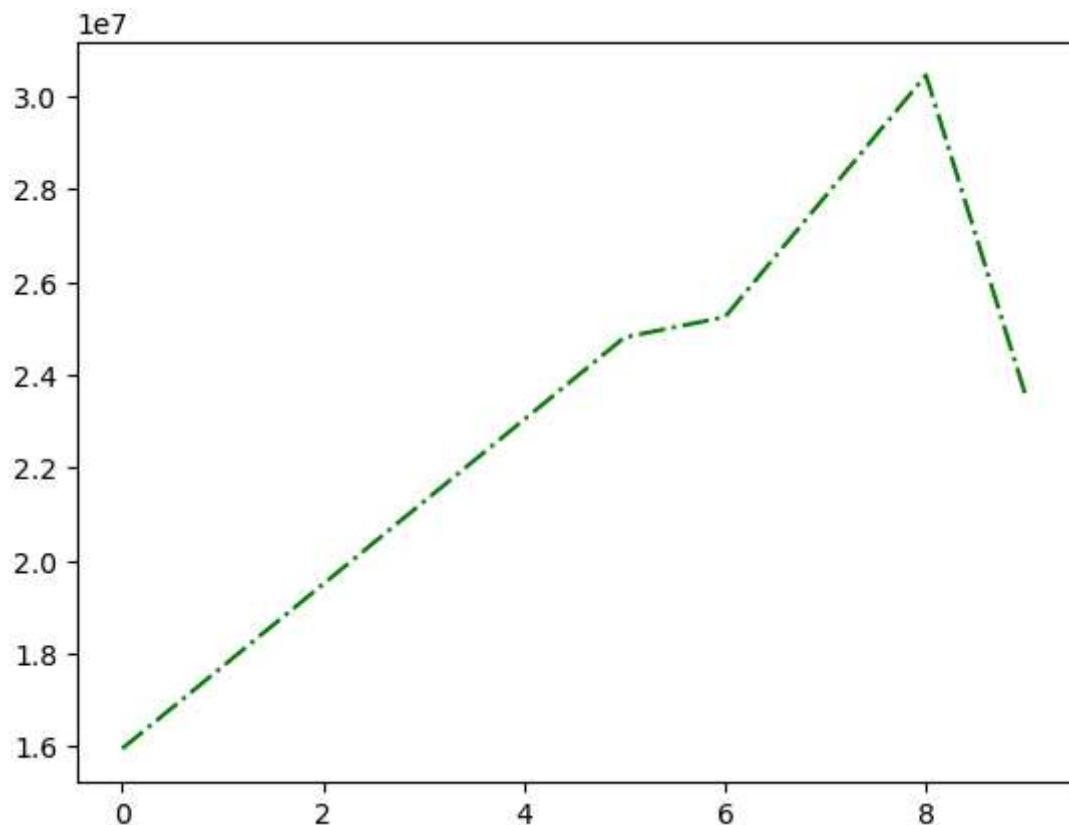
```
In [48]: plt.plot(Salary[0], color ='black', ls = ':')
```

```
Out[48]: [<matplotlib.lines.Line2D at 0x160ed37c6e0>]
```



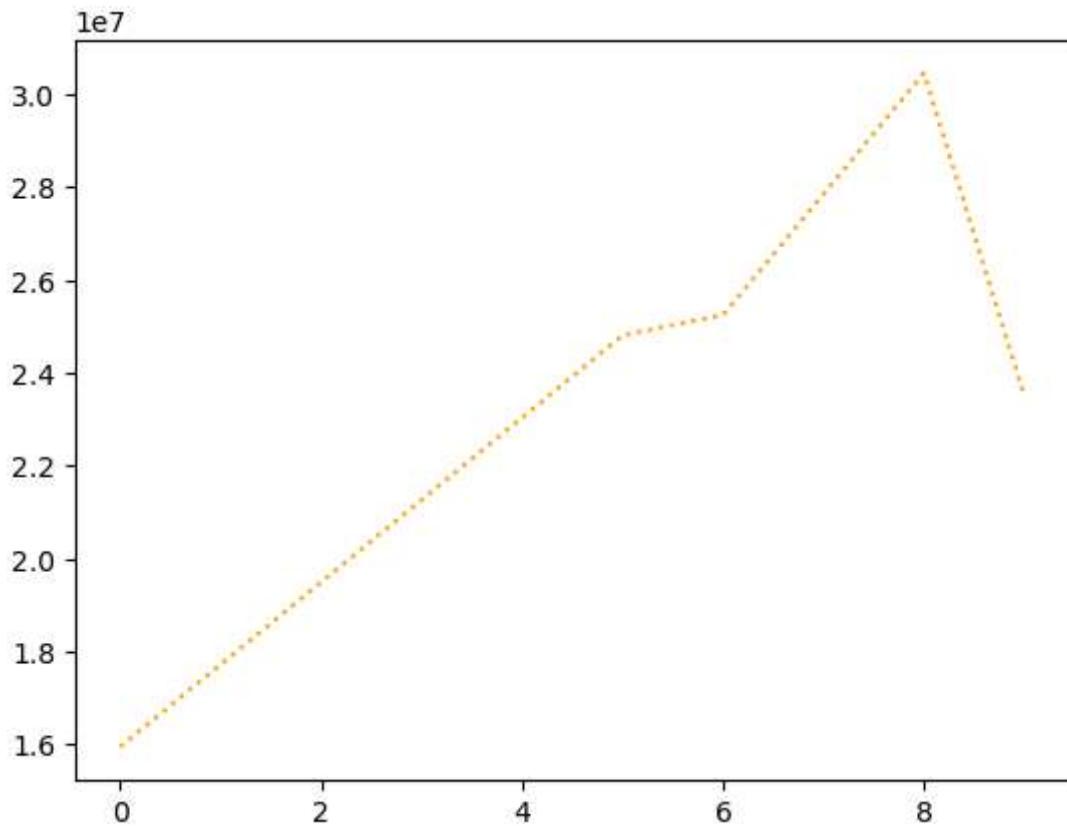
```
In [50]: plt.plot(Salary[0], color ='green', ls = '-.')
```

```
Out[50]: [<matplotlib.lines.Line2D at 0x160ed3ddbb0>]
```



```
In [52]: plt.plot(Salary[0], color ='orange', ls = 'dotted')
```

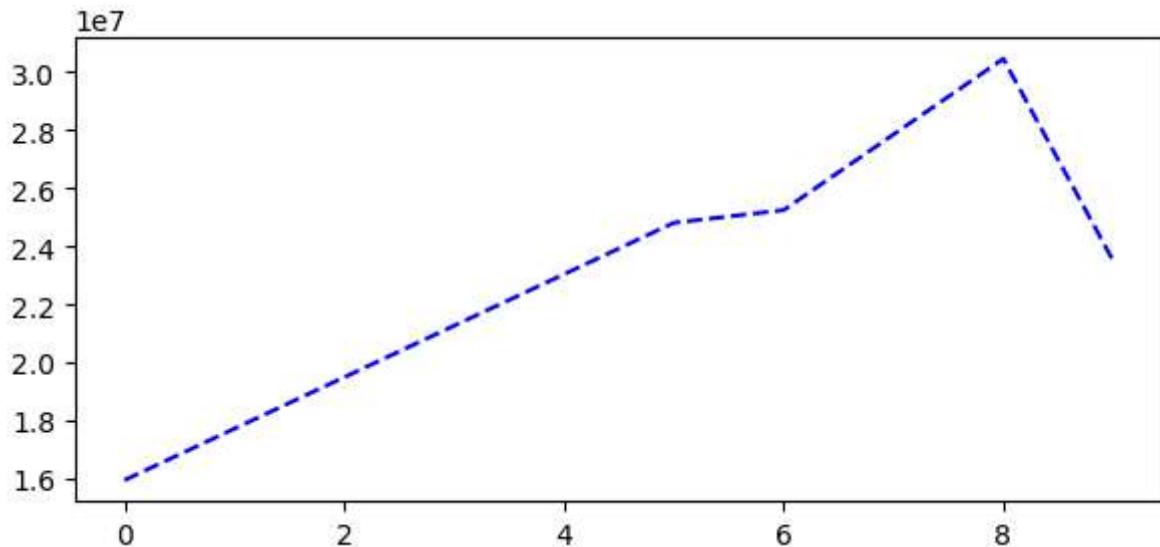
```
Out[52]: [
```



```
In [56]: #to reduce graph length  
%matplotlib inline  
plt.rcParams['figure.figsize'] = 7,3 #7 is width of the graph , 3 is height
```

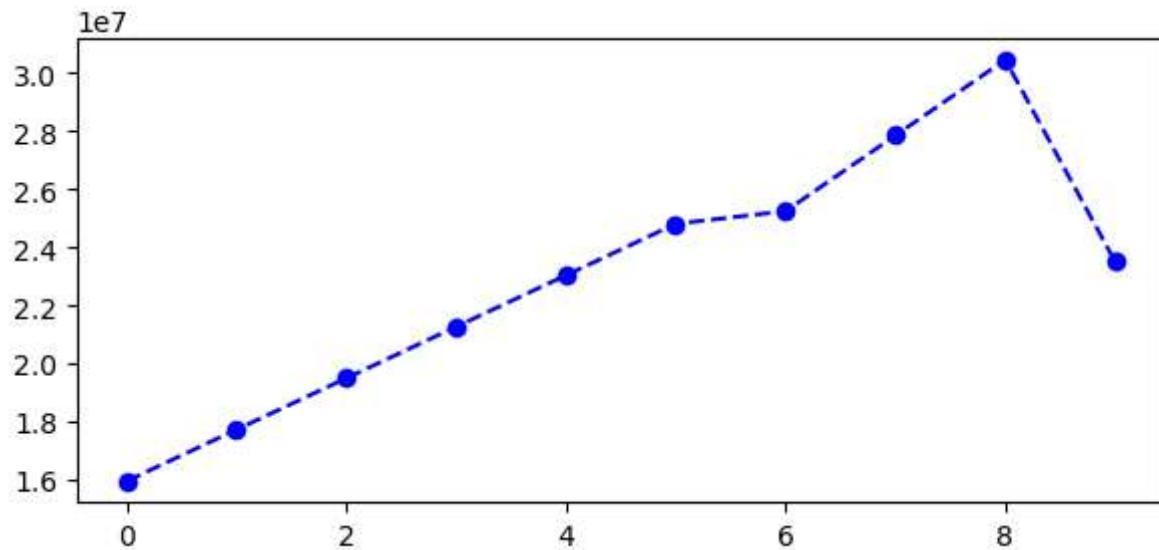
```
In [58]: plt.plot(Salary[0],c = 'Blue', ls = '--')
```

```
Out[58]: [
```



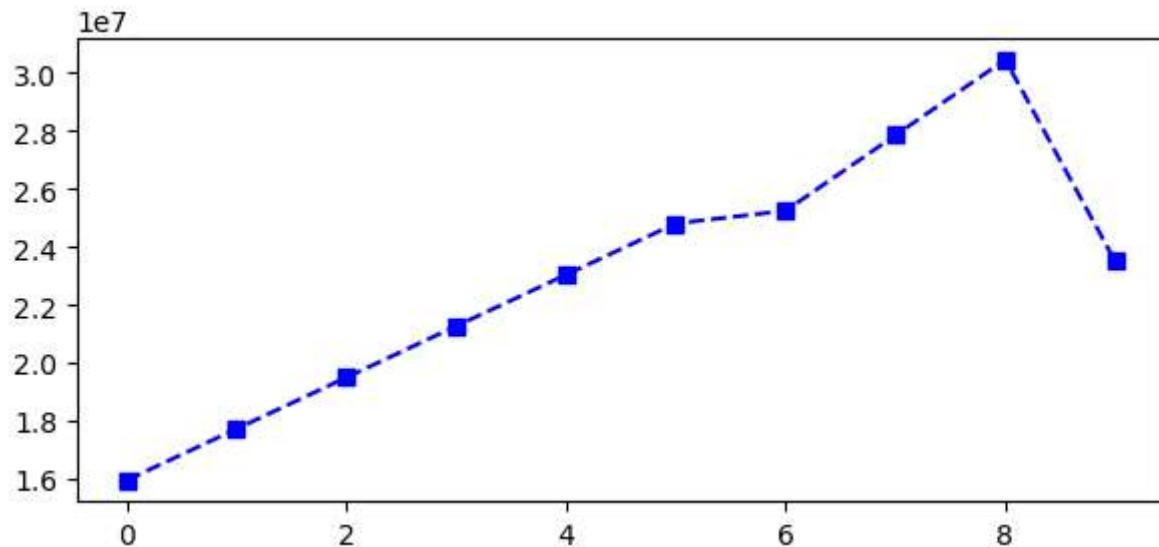
```
In [60]: plt.plot(Salary[0],c = 'Blue', ls = '--', marker = 'o') # D is square, o is circle
```

```
Out[60]: [<matplotlib.lines.Line2D at 0x160eca23740>]
```



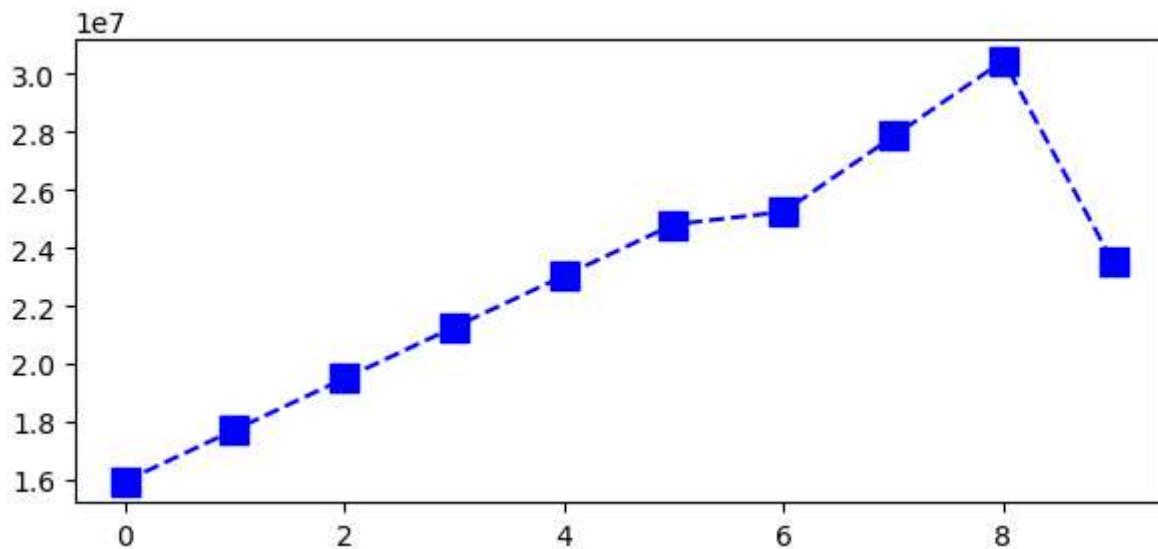
```
In [62]: plt.plot(Salary[0],c = 'Blue', ls = '--', marker = 's')
```

```
Out[62]: [<matplotlib.lines.Line2D at 0x160ee44be60>]
```



```
In [64]: plt.plot(Salary[0],c = 'Blue', ls = '--', marker = 's', ms = 10)
```

```
Out[64]: [<matplotlib.lines.Line2D at 0x160ee4dcbf0>]
```



```
In [66]: list(range(0,10))
```

```
Out[66]: [0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
```

```
In [68]: Sdict
```

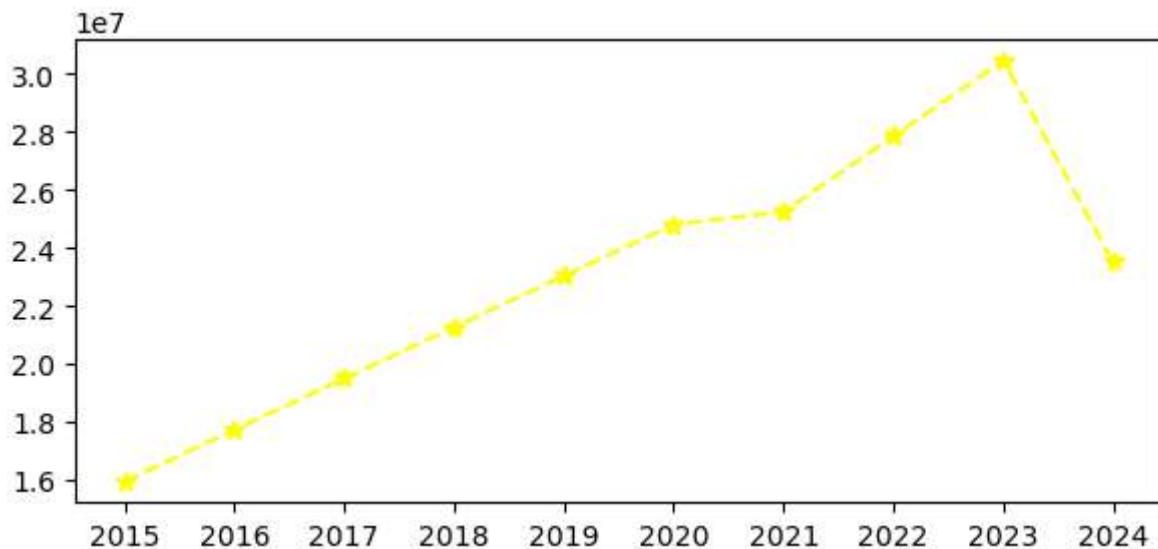
```
Out[68]: {'2015': 0,
          '2016': 1,
          '2017': 2,
          '2018': 3,
          '2019': 4,
          '2020': 5,
          '2021': 6,
          '2022': 7,
          '2023': 8,
          '2024': 9}
```

```
In [70]: Pdict
```

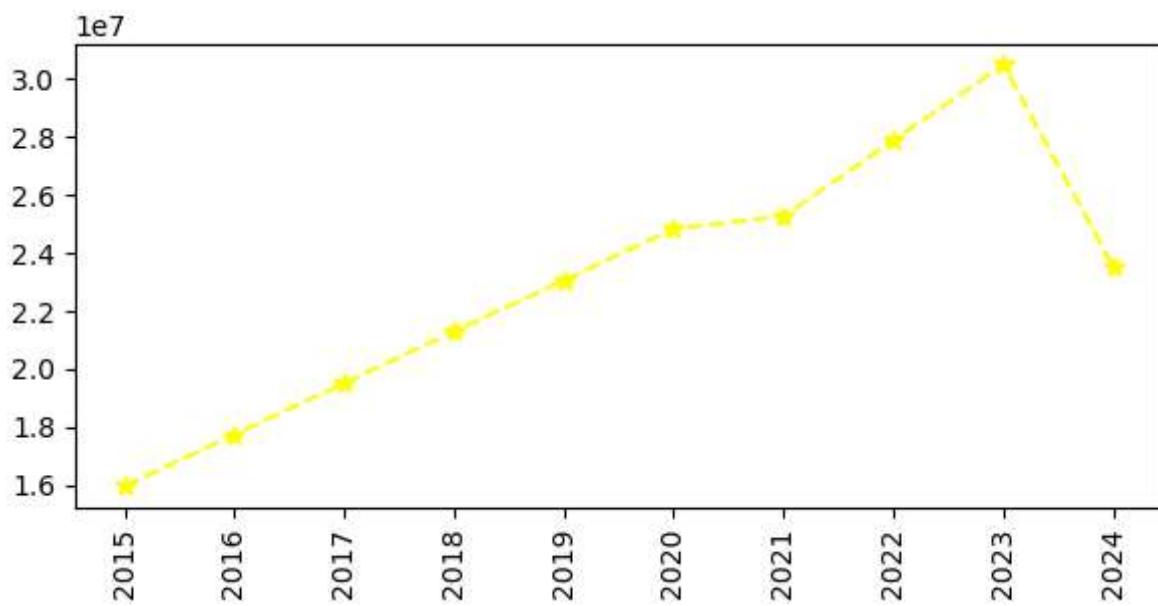
```
Out[70]: {'Sachin': 0,
          'Rahul': 1,
          'Smith': 2,
          'Sami': 3,
          'Pollard': 4,
          'Morris': 5,
          'Samson': 6,
          'Dhoni': 7,
          'Kohli': 8,
          'Sky': 9}
```

```
In [72]: plt.plot(Salary[0], c = 'Yellow', ls = '--', marker = '*', ms = 7)
plt.xticks(list(range(0,10)), Seasons)
plt.show()
```

#Here the insight of the below graph is that we are checking the salary of Sachin a



```
In [74]: plt.plot(Salary[0],c = 'Yellow', ls = '--', marker = '*', ms = 7, label = Players[0])
plt.xticks(list(range(0,10)),Seasons,rotation = 'vertical')
plt.show()
```

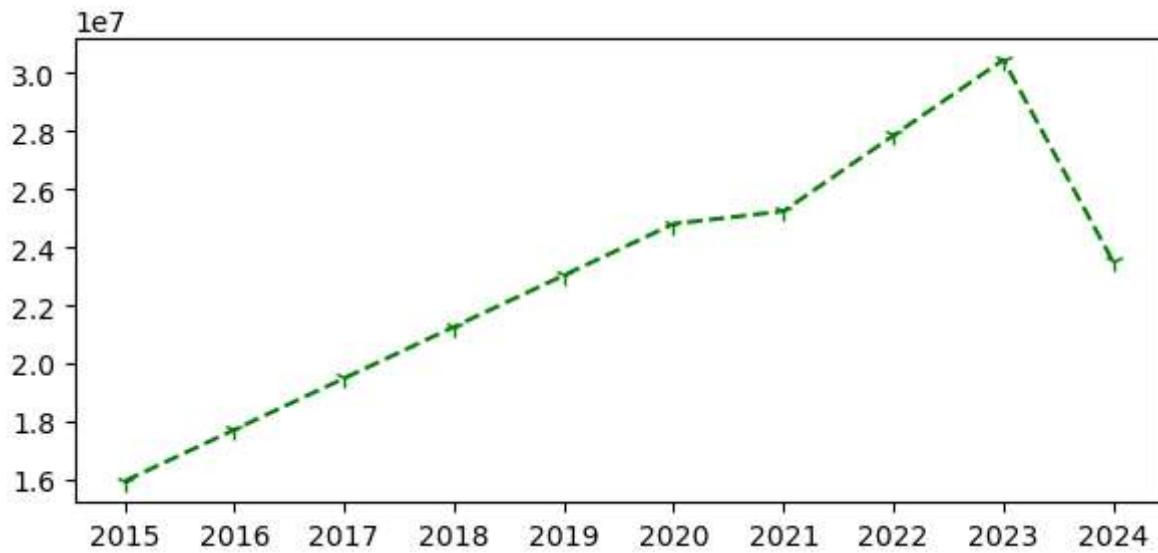


```
In [76]: Games
```

```
Out[76]: 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 [78]: plt.plot(Salary[0],c = 'Green', ls = '--', marker = '1', ms = 7, label = Players[0])
plt.xticks(list(range(0,10)),Seasons, rotation = 'horizontal')
```

```
plt.show()
```



```
In [80]: Salary[0]
```

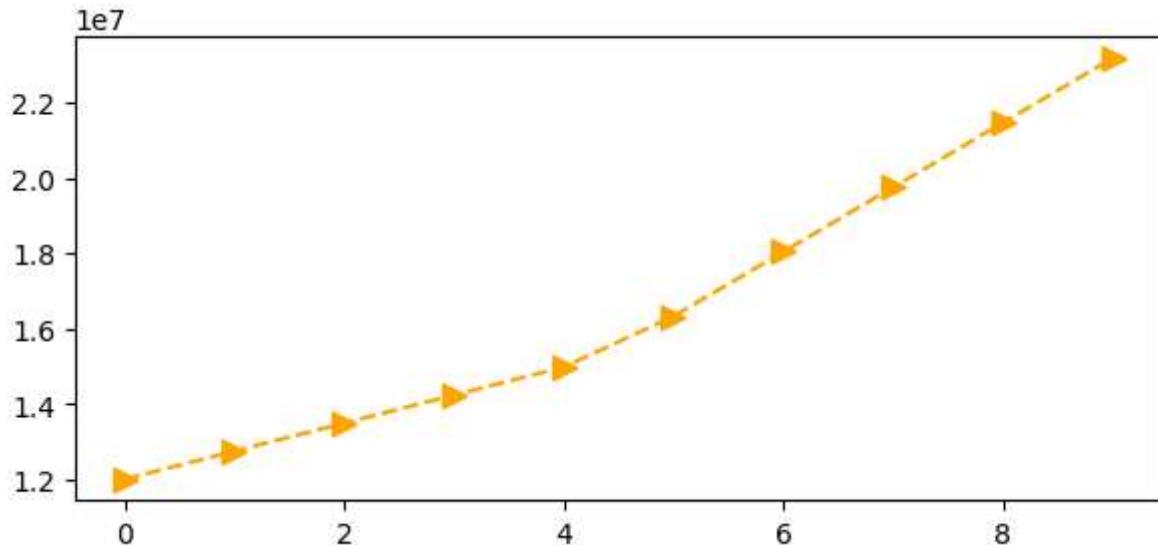
```
Out[80]: array([15946875, 17718750, 19490625, 21262500, 23034375, 24806250,  
 25244493, 27849149, 30453805, 23500000])
```

```
In [82]: Salary[1]
```

```
Out[82]: array([12000000, 12744189, 13488377, 14232567, 14976754, 16324500,  
 18038573, 19752645, 21466718, 23180790])
```

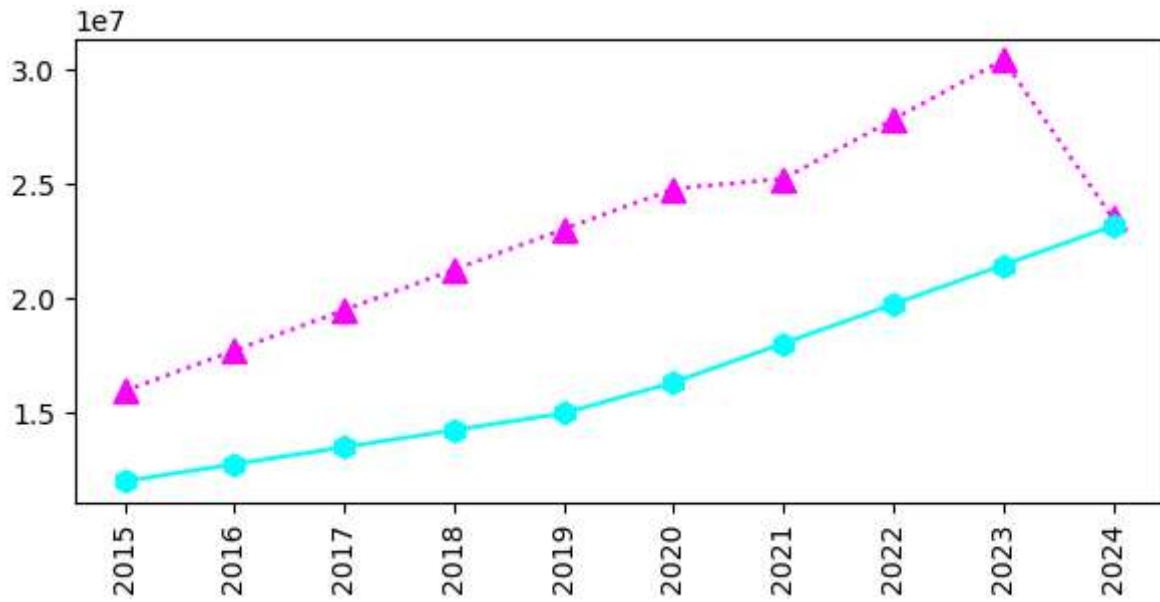
```
In [84]: plt.plot(Salary[1],c = 'Orange',ls = '--',marker = '>',ms = 9, label = Players[1])
```

```
Out[84]: [<matplotlib.lines.Line2D at 0x160ec997560>]
```

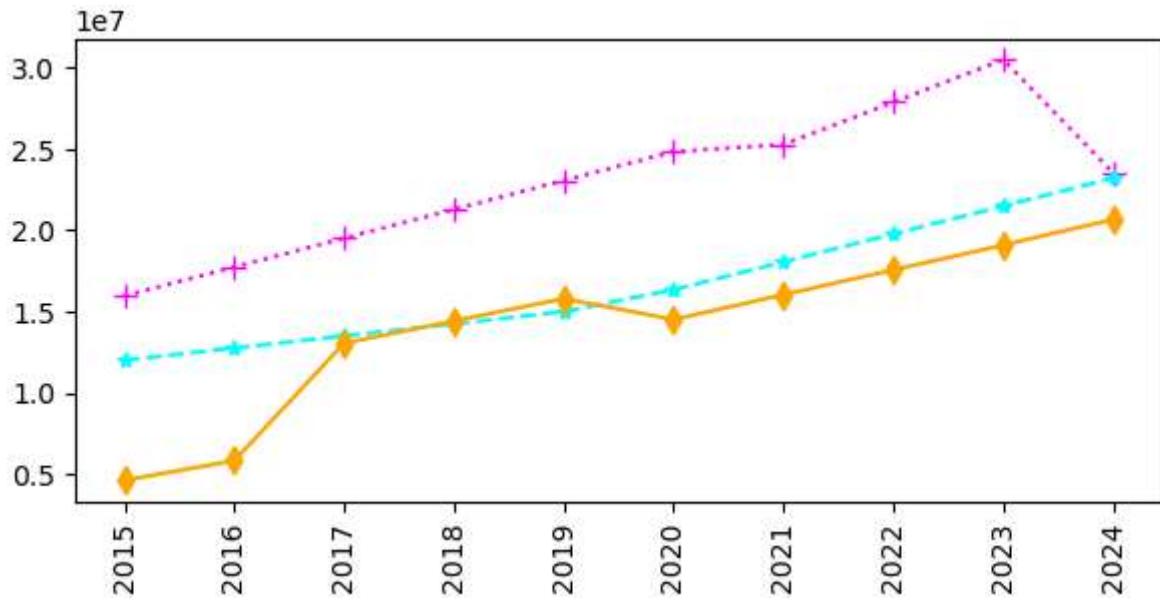


More Visualisations

```
In [149... plt.plot(Salary[0], c = 'magenta',ls = ':',marker = '^',ms = 8, label = Players[0])
plt.plot(Salary[1], c = 'cyan',ls = '-.',marker = 'h',ms = 8, label = Players[1])
plt.xticks(list(range(0,10)),Seasons,rotation = 'vertical')
plt.show()
```



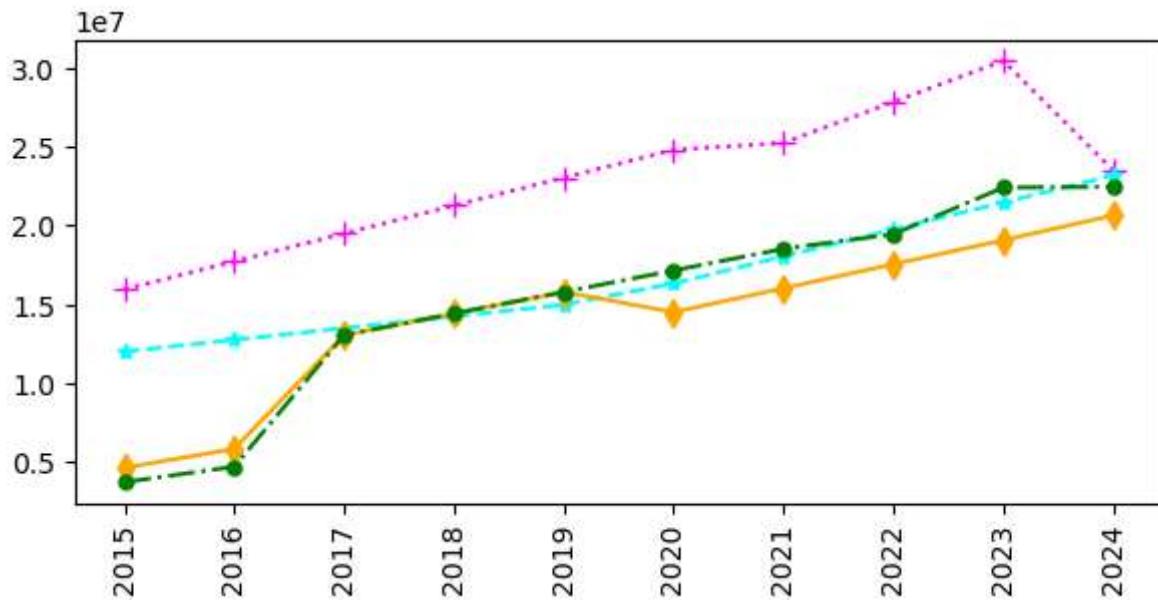
```
In [151... plt.plot(Salary[0], c = 'magenta',ls = ':',marker = '+',ms = 8, label = Players[0])
plt.plot(Salary[1], c = 'cyan',ls = '--',marker = '*',ms = 6, label = Players[1])
plt.plot(Salary[2], c = 'orange',ls = '-.',marker = 'd',ms = 7, label = Players[2])
plt.xticks(list(range(0,10)),Seasons,rotation = 'vertical')
plt.show()
```



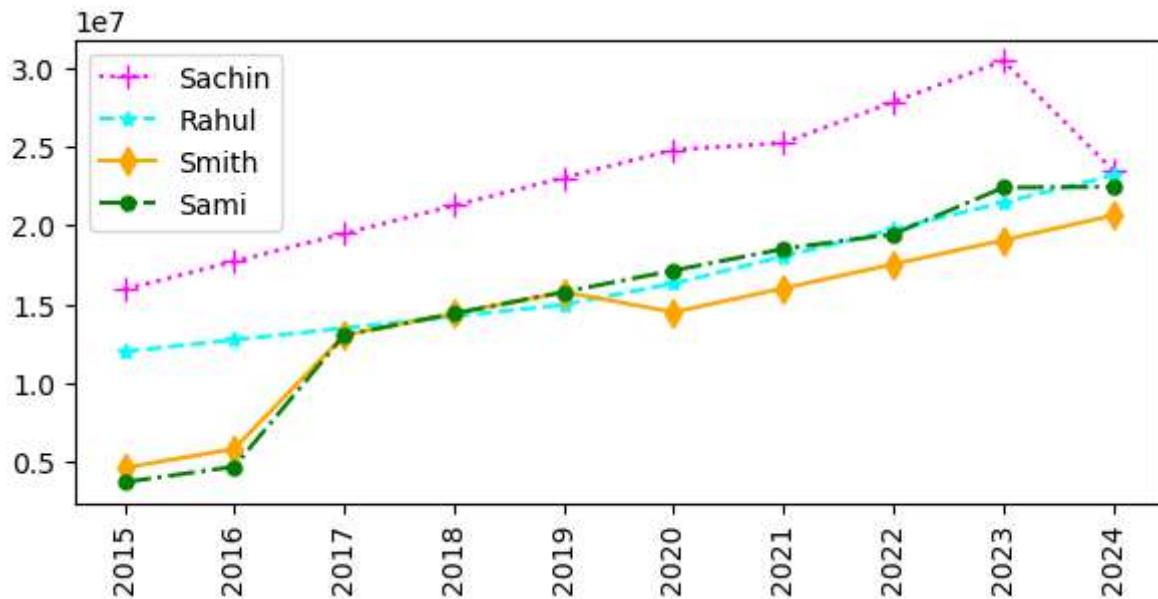
```
In [153... plt.plot(Salary[0], c = 'magenta',ls = ':',marker = '+',ms = 8, label = Players[0])
plt.plot(Salary[1], c = 'cyan',ls = '--',marker = '*',ms = 6, label = Players[1])
plt.plot(Salary[2], c = 'orange',ls = '-.',marker = 'd',ms = 7, label = Players[2])
plt.plot(Salary[3], c = 'Green',ls = '-.',marker = 'o',ms = 5, label = Players[3])

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

```
plt.show()
```



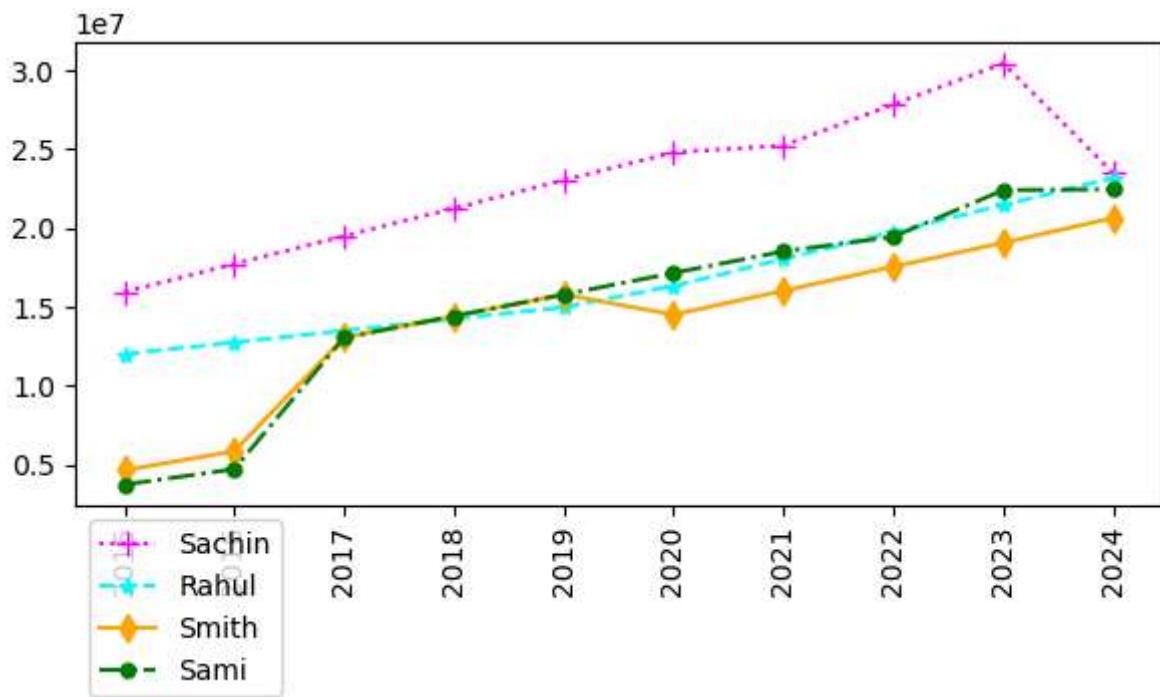
```
In [155]:  
plt.plot(Salary[0], c = 'magenta',ls = ':' ,marker = '+',ms = 8, label = Players[0])  
plt.plot(Salary[1], c = 'cyan',ls = '--',marker = '*',ms = 6, label = Players[1])  
plt.plot(Salary[2], c = 'orange',ls = '-.',marker = 'd',ms = 7, label = Players[2])  
plt.plot(Salary[3], c = 'Green',ls = '-.',marker = 'o',ms = 5, label = Players[3])  
plt.legend() #adding Legend  
plt.xticks(list(range(0,10)),Seasons,rotation = 'vertical')  
  
plt.show()
```



```
In [157]:  
plt.plot(Salary[0], c = 'magenta',ls = ':' ,marker = '+',ms = 8, label = Players[0])  
plt.plot(Salary[1], c = 'cyan',ls = '--',marker = '*',ms = 6, label = Players[1])  
plt.plot(Salary[2], c = 'orange',ls = '-.',marker = 'd',ms = 7, label = Players[2])  
plt.plot(Salary[3], c = 'Green',ls = '-.',marker = 'o',ms = 5, label = Players[3])  
#adding Legend with values  
plt.legend(loc = 'upper left', bbox_to_anchor=(0,0))
```

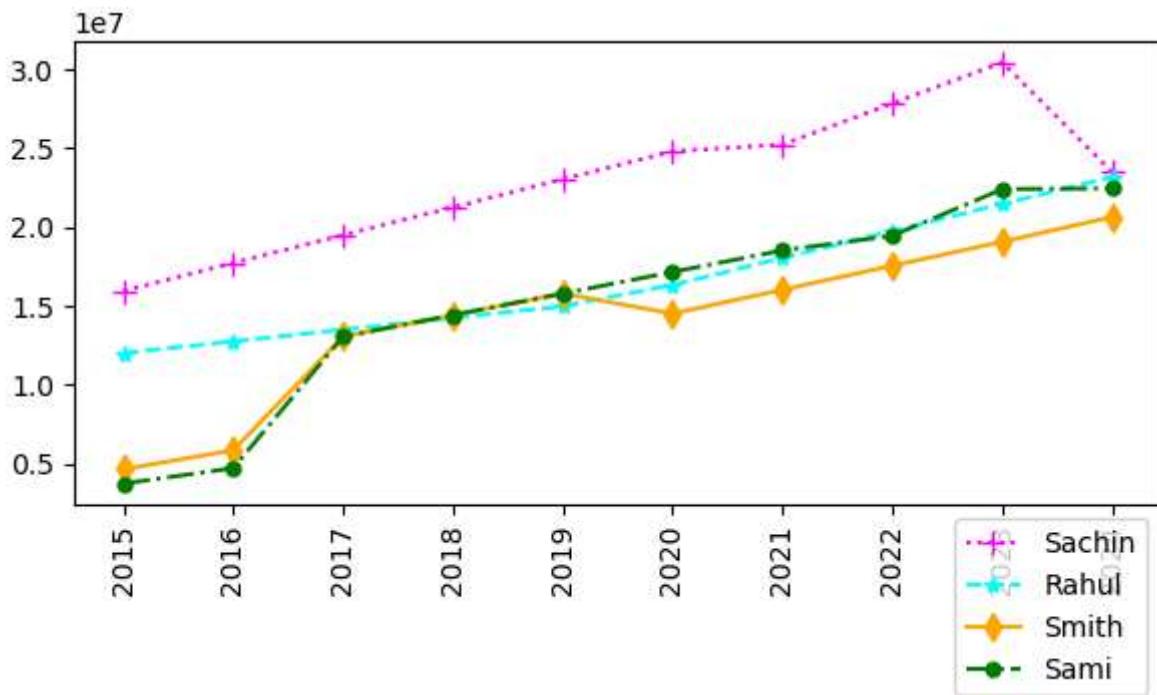
```
plt.xticks(list(range(0,10)),Seasons,rotation = 'vertical')

plt.show()
```



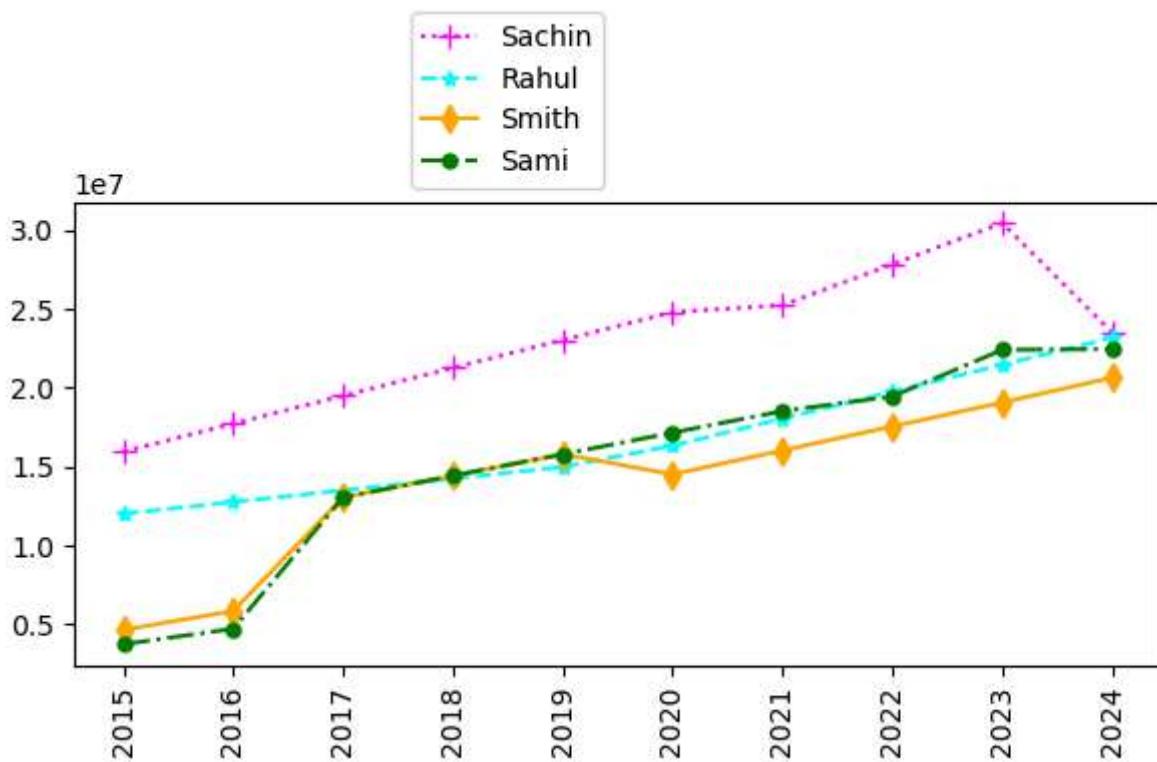
```
In [159...]
plt.plot(Salary[0], c = 'magenta',ls = ':' ,marker = '+',ms = 8, label = Players[0])
plt.plot(Salary[1], c = 'cyan',ls = '--' ,marker = '*',ms = 6, label = Players[1])
plt.plot(Salary[2], c = 'orange',ls = '-' ,marker = 'd',ms = 7, label = Players[2])
plt.plot(Salary[3], c = 'Green',ls = '-.' ,marker = 'o',ms = 5, label = Players[3])
#adding legend with values
plt.legend(loc = 'upper right', bbox_to_anchor=(1,0))
plt.xticks(list(range(0,10)),Seasons,rotation = 'vertical')

plt.show()
```



```
In [161]: plt.plot(Salary[0], c = 'magenta',ls = ':' ,marker = '+',ms = 8, label = Players[0])
plt.plot(Salary[1], c = 'cyan',ls = '--' ,marker = '*',ms = 6, label = Players[1])
plt.plot(Salary[2], c = 'orange',ls = '-' ,marker = 'd',ms = 7, label = Players[2])
plt.plot(Salary[3], c = 'Green',ls = '-.' ,marker = 'o',ms = 5, label = Players[3])
#adding Legend with values
plt.legend(loc = 'lower right', bbox_to_anchor=(0.5,1))
plt.xticks(list(range(0,10)),Seasons,rotation = 'vertical')

plt.show()
```



In [145...]

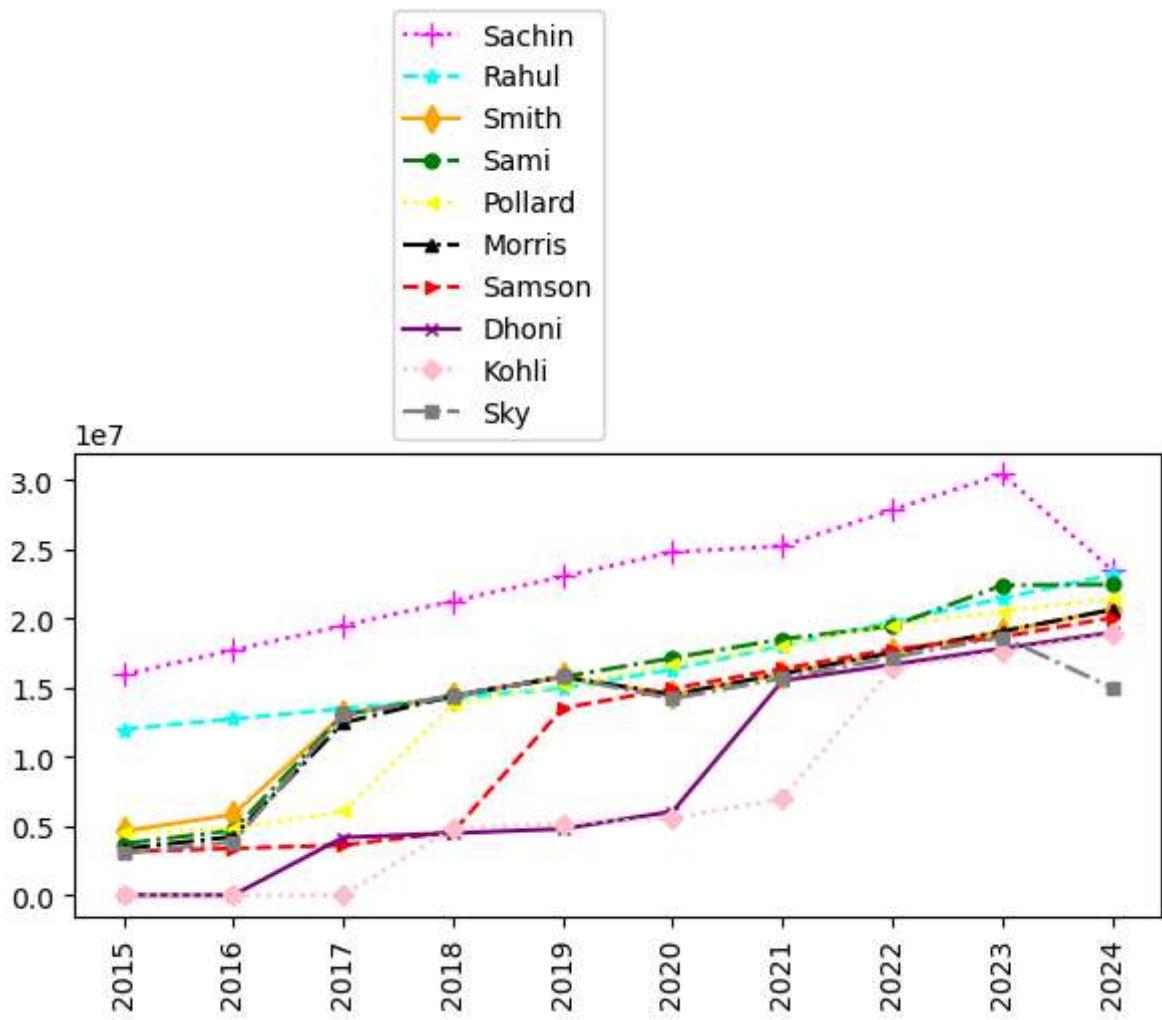
```

plt.plot(Salary[0], c = 'magenta',ls = ':' ,marker = '+',ms = 8, label = Players[0])
plt.plot(Salary[1], c = 'cyan',ls = '--',marker = '*',ms = 6, label = Players[1])
plt.plot(Salary[2], c = 'orange',ls = '-.',marker = 'd',ms = 7, label = Players[2])
plt.plot(Salary[3], c = 'Green',ls = '-.',marker = 'o',ms = 5, label = Players[3])
plt.plot(Salary[4], c = 'yellow',ls = ':' ,marker = '<',ms = 5, label = Players[4])
plt.plot(Salary[5], c = 'Black',ls = '-.',marker = '^',ms = 5, label = Players[5])
plt.plot(Salary[6], c = 'Red',ls = '--',marker = '>',ms = 5, label = Players[6])
plt.plot(Salary[7], c = 'Purple',ls = '-.',marker = 'x',ms = 5, label = Players[7])
plt.plot(Salary[8], c = 'Pink',ls = ':' ,marker = 'D',ms = 5, label = Players[8])
plt.plot(Salary[9], c = 'Grey',ls = '-.',marker = 's',ms = 5, label = Players[9])

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

plt.show()

```



In [163...]

```

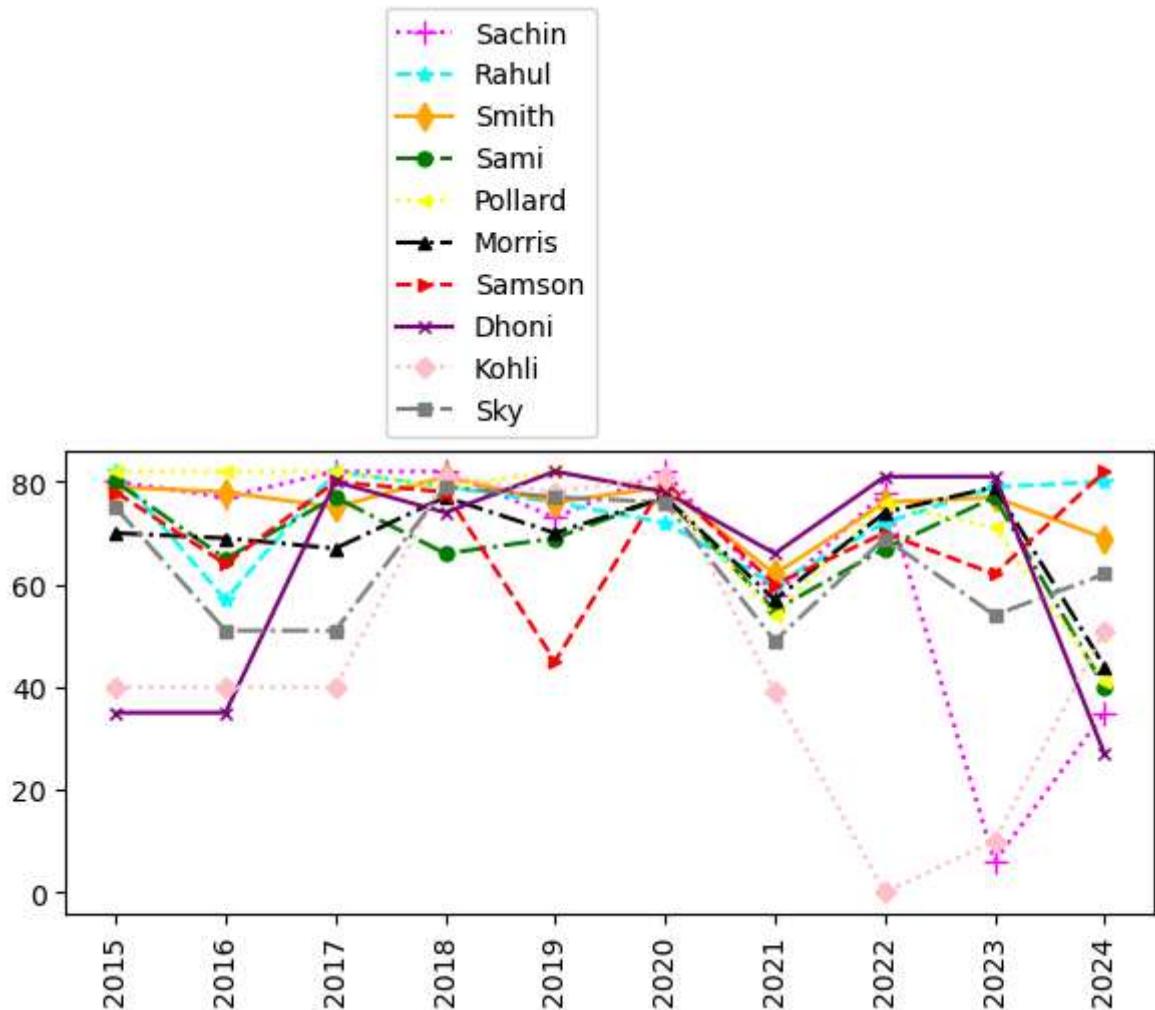
#We can visualise how many games are being played by a player
plt.plot(Games[0], c = 'magenta',ls = ':' ,marker = '+',ms = 8, label = Players[0])
plt.plot(Games[1], c = 'cyan',ls = '--',marker = '*',ms = 6, label = Players[1])
plt.plot(Games[2], c = 'orange',ls = '-.',marker = 'd',ms = 7, label = Players[2])
plt.plot(Games[3], c = 'Green',ls = '-.',marker = 'o',ms = 5, label = Players[3])
plt.plot(Games[4], c = 'yellow',ls = ':' ,marker = '<',ms = 5, label = Players[4])
plt.plot(Games[5], c = 'Black',ls = '-.',marker = '^',ms = 5, label = Players[5])

```

```
plt.plot(Games[6], c = 'Red',ls = '--',marker = '>',ms = 5, label = Players[6])
plt.plot(Games[7], c = 'Purple',ls = '-.',marker = 'x',ms = 5, label = Players[7])
plt.plot(Games[8], c = 'Pink',ls = ':',marker = 'D',ms = 5, label = Players[8])
plt.plot(Games[9], c = 'Grey',ls = '-.',marker = 's',ms = 5, label = Players[9])

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

plt.show()
```



In []:

In []: