

```

In [5]: import numpy as np

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

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

#Salaries
Sachin_Salary = [15946875, 17718750, 19490625, 21262500, 23034375, 24806250, 25244
Rahul_Salary = [12000000, 12744189, 13488377, 14232567, 14976754, 16324500, 18038573
Smith_Salary = [4621800, 5828090, 13041250, 14410581, 15779912, 14500000, 1602250
Sami_Salary = [3713640, 4694041, 13041250, 14410581, 15779912, 17149243, 18518574, 1
Pollard_Salary = [4493160, 4806720, 6061274, 13758000, 15202590, 16647180, 18091770
Morris_Salary = [3348000, 4235220, 12455000, 14410581, 15779912, 14500000, 1602250
Samson_Salary = [3144240, 3380160, 3615960, 4574189, 13520500, 14940153, 16359805, 1
Dhoni_Salary = [0, 0, 4171200, 4484040, 4796880, 6053663, 15506632, 16669630, 178326
Kohli_Salary = [0, 0, 0, 4822800, 5184480, 5546160, 6993708, 16402500, 17632688, 188
Sky_Salary = [3031920, 3841443, 13041250, 14410581, 15779912, 14200000, 15691000,

#Matrix
Salary = np.array([Sachin_Salary, Rahul_Salary, Smith_Salary, Sami_Salary, Pollar

#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 [7]: Salary

```

```
Out[7]: 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 [8]: Games

```
Out[8]: 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 [9]: np.round(Salary/Games)

C:\Users\Acer\AppData\Local\Temp\ipykernel\_5460\3232172828.py:1: RuntimeWarning:  
divide by zero encountered in divide  
np.round(Salary/Games)

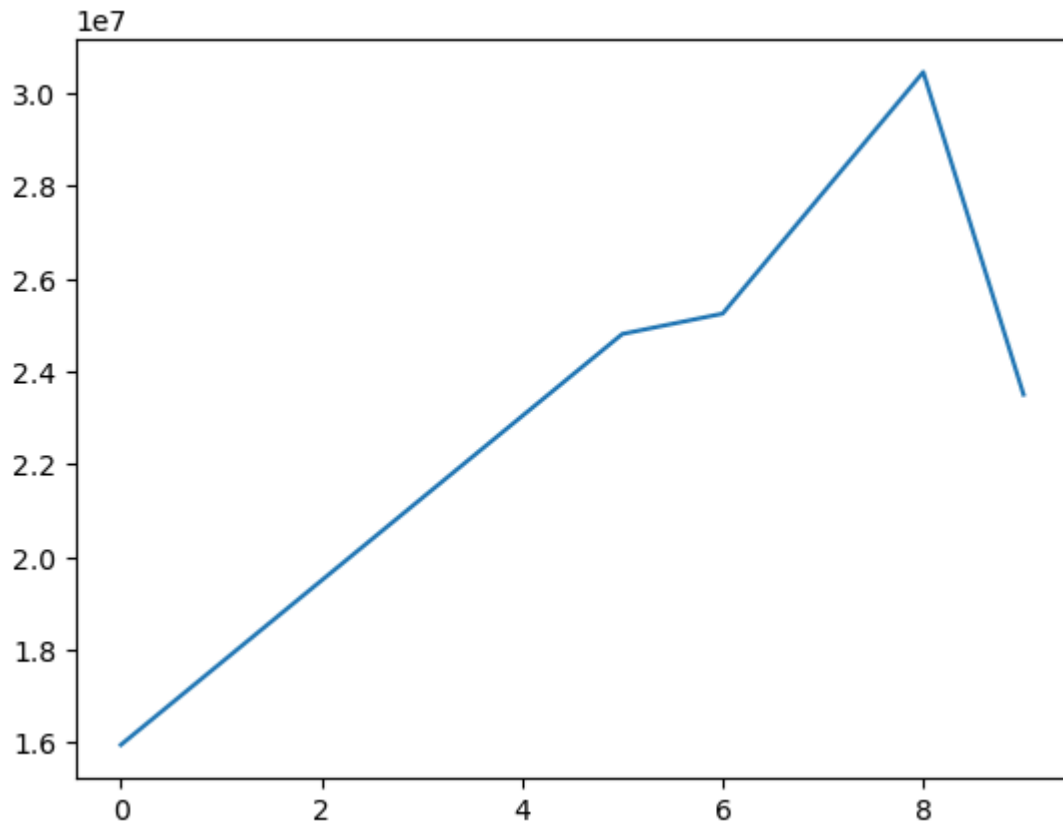
```
Out[9]: array([[ 199336.,  230114.,  237691.,  259299.,  315539.,  302515.,
                 435250.,  357040.,  5075634.,  671429.],
               [ 146341.,  223582.,  164492.,  180159.,  197063.,  226729.,
                 300643.,  274342.,  271731.,  289760.],
               [  58504.,  74719.,  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 [10]: import warnings  
warnings.filterwarnings('ignore')
```

```
In [11]: import matplotlib.pyplot as PLT
```

```
In [12]: PLT.plot(Salary[0])
```

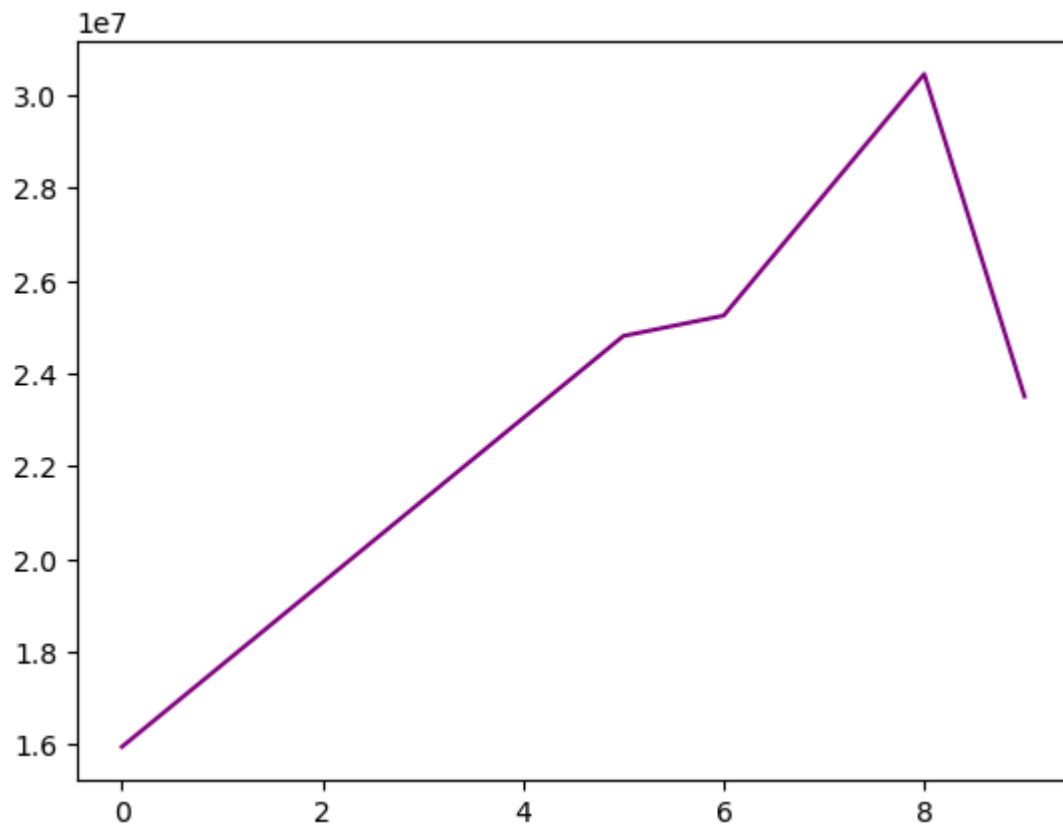
```
Out[12]: [<matplotlib.lines.Line2D at 0x18756e07750>]
```



```
In [13]: # In the above graph x axis data is displayed automatically by matplotlib, this
```

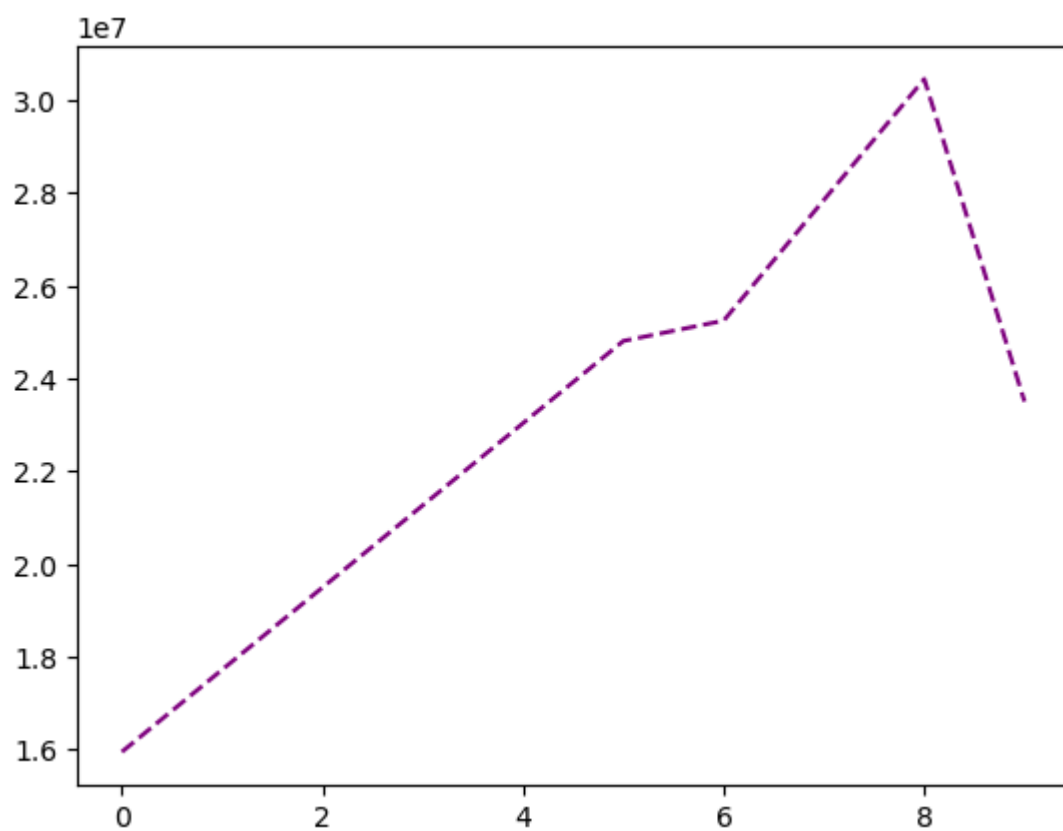
```
In [14]: PLT.plot(Salary[0], color = 'purple')
```

```
Out[14]: [<matplotlib.lines.Line2D at 0x18756ebf9d0>]
```



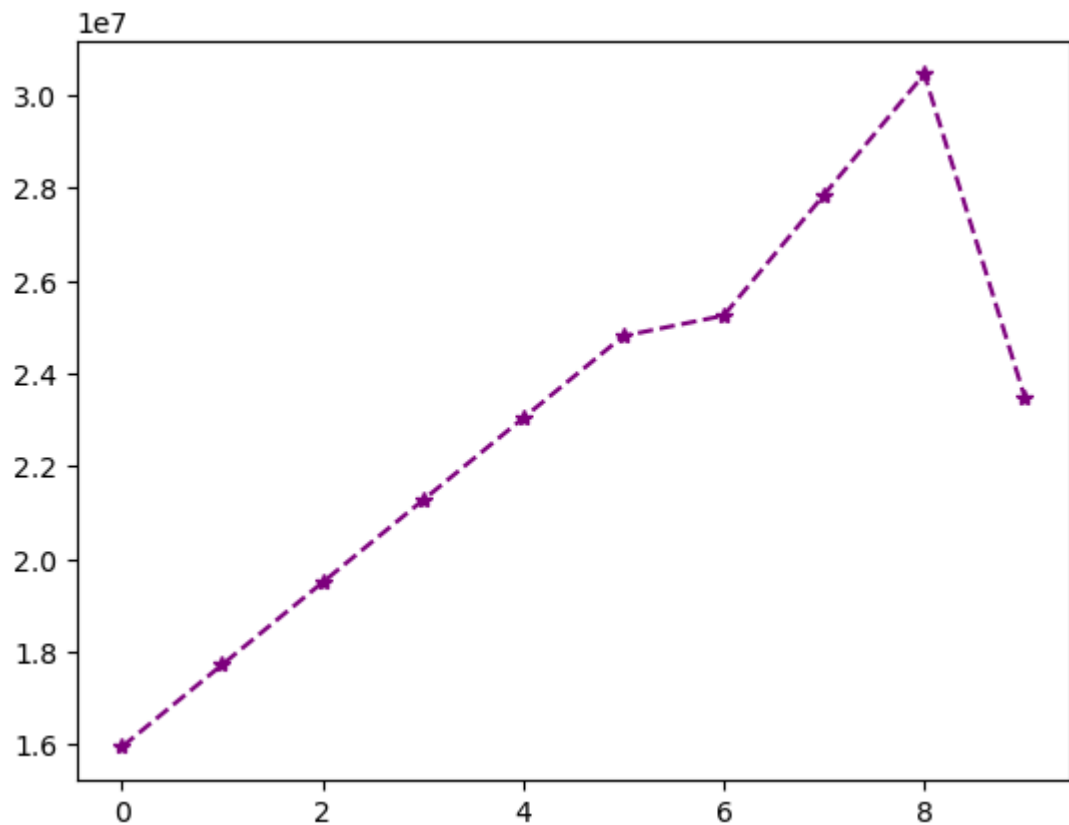
```
In [15]: PLT.plot(Salary[0], c = 'purple', ls = '--') #we can use c as shorthand for color
```

```
Out[15]: [<matplotlib.lines.Line2D at 0x187570ce990>]
```



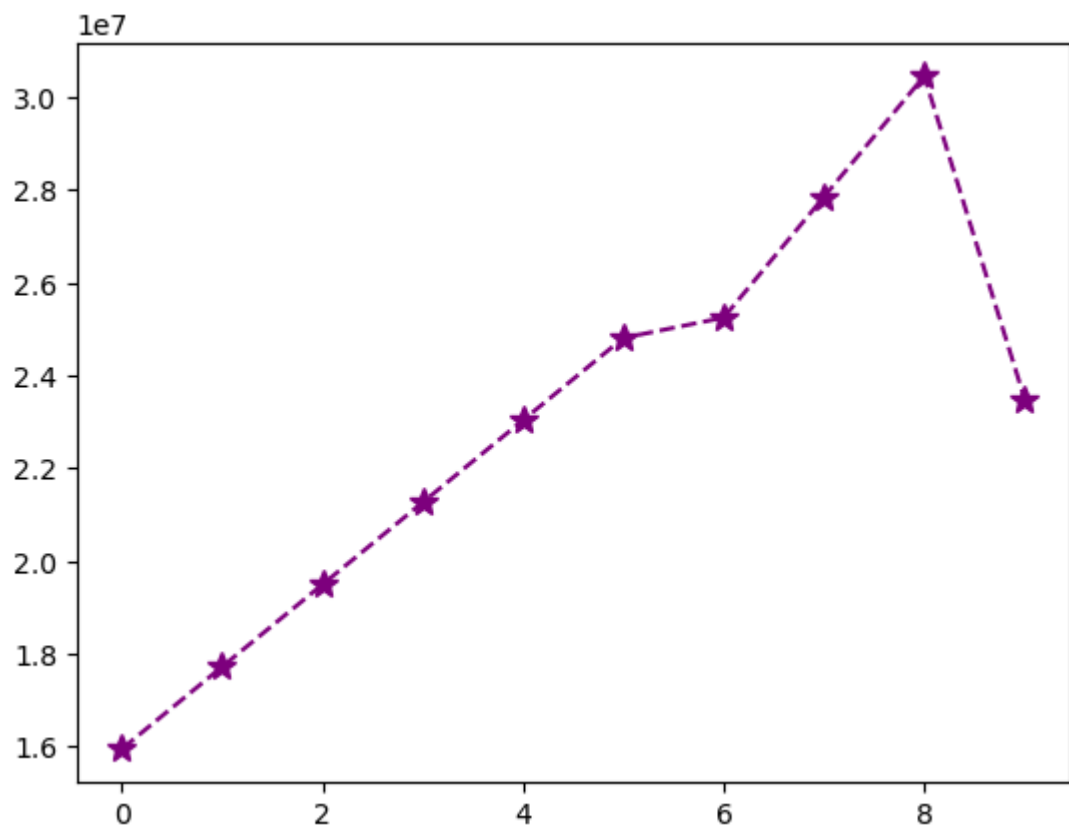
```
In [16]: PLT.plot(Salary[0], c = 'purple', ls = '--', marker = '*')
```

```
Out[16]: [<matplotlib.lines.Line2D at 0x1875733d450>]
```



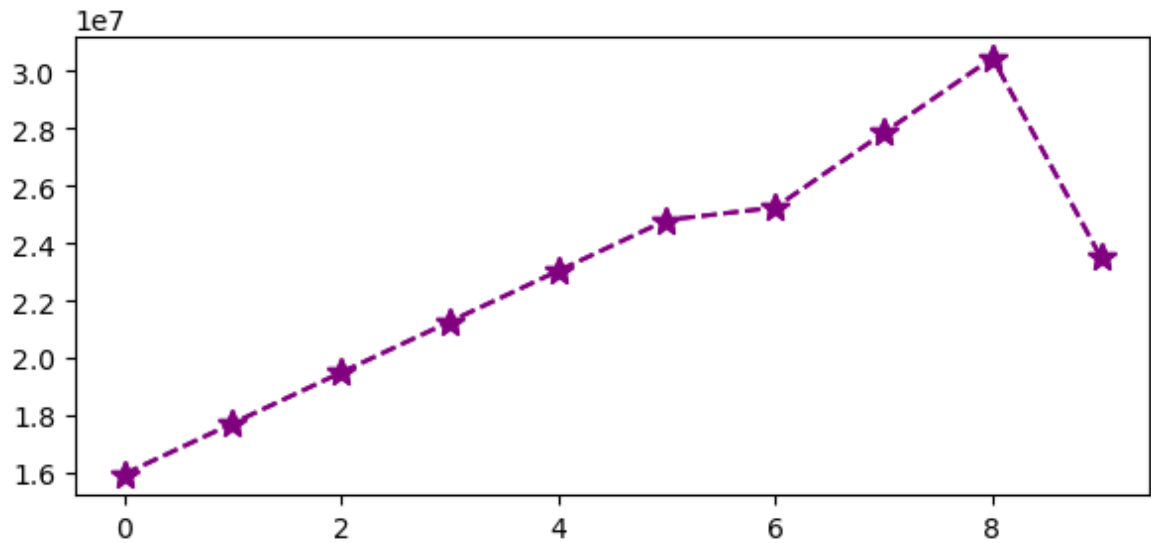
```
In [18]: PLT.plot(Salary[0], c = 'purple', ls = '--', marker = '*', ms = 10) #ms means ma
```

```
Out[18]: [<matplotlib.lines.Line2D at 0x18757ac9f90>]
```



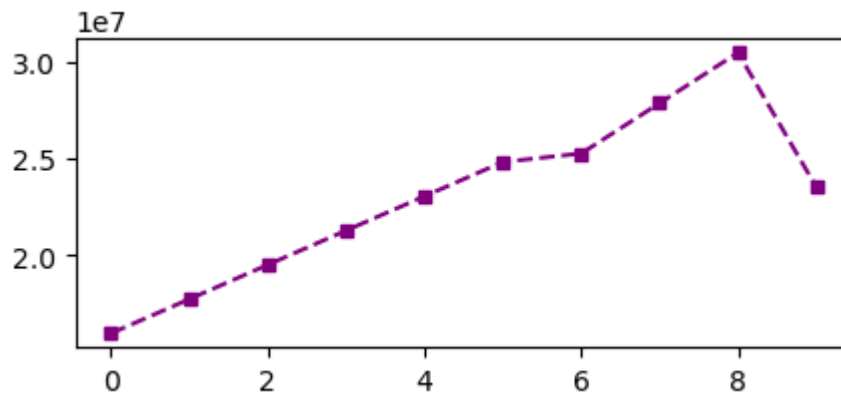
```
In [19]: %matplotlib inline
          PLT.rcParams['figure.figsize'] = 7,3 #7 = Height, 3 = Width
```

```
In [21]: PLT.plot(Salary[0], c = 'purple', ls = '--', marker = '*', ms = 10)
         PLT.show()
```



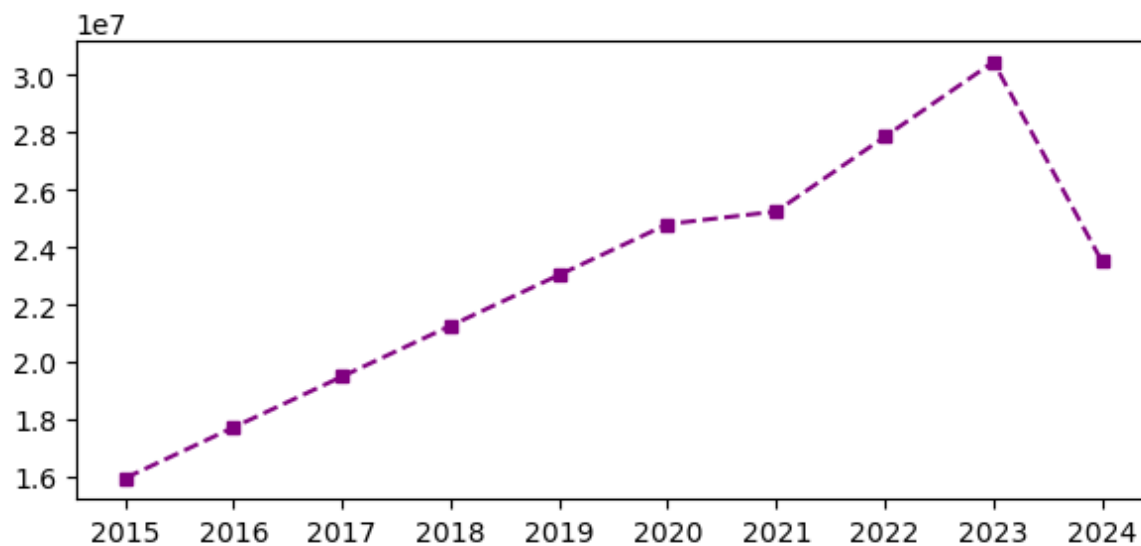
```
In [22]: %matplotlib inline
         PLT.rcParams['figure.figsize'] = 5,2 #5 = Height, 2 = Width
```

```
In [25]: PLT.plot(Salary[0], c = 'purple', ls = '--', marker = 's', ms = 5)
         PLT.show()
```

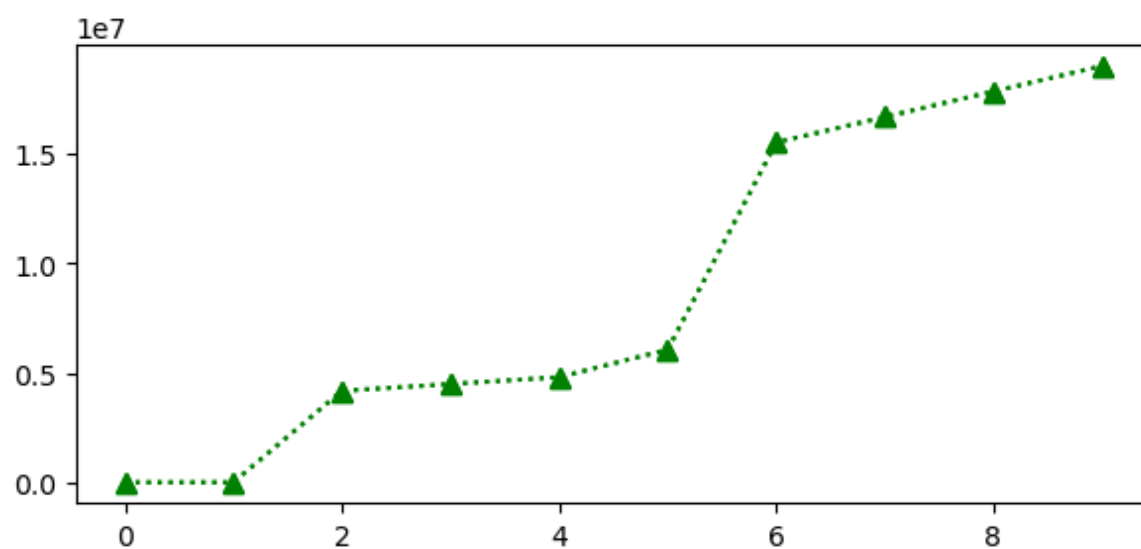


```
In [26]: %matplotlib inline
         PLT.rcParams['figure.figsize'] = 7,3
```

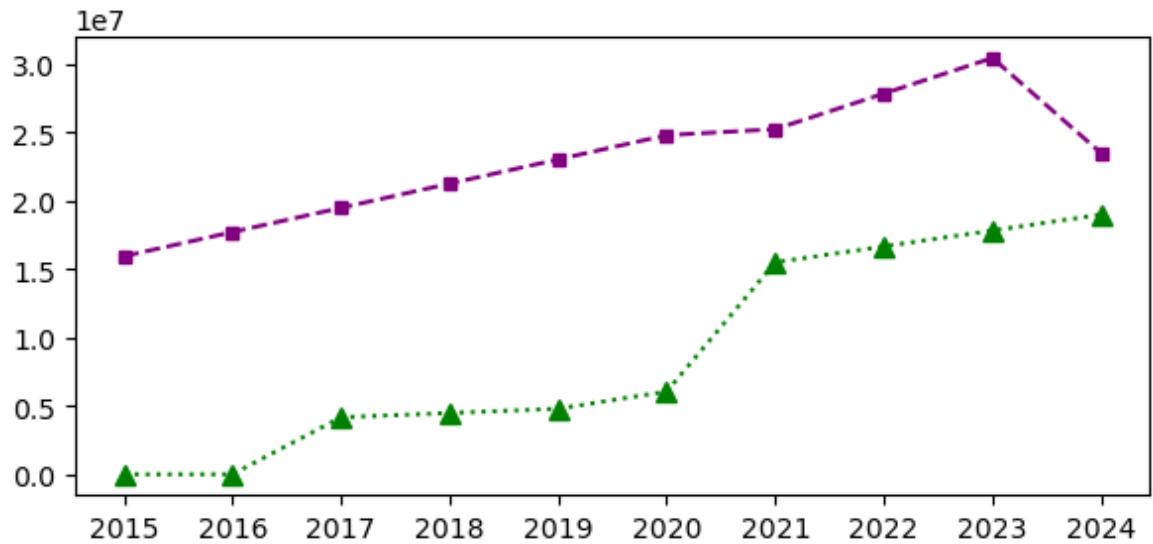
```
In [27]: PLT.plot(Salary[0], c = 'purple', ls = '--', marker = 's', ms = 5)
         PLT.xticks(list(range(0,10)),Seasons)
         PLT.show()
```



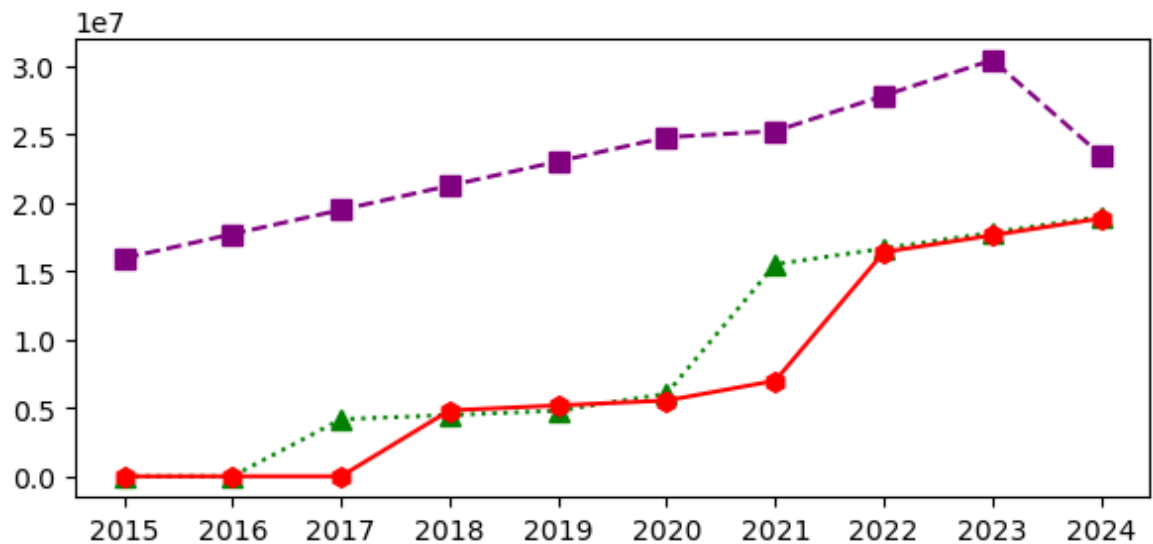
```
In [29]: PLT.plot(Salary[7], c = 'green', ls = ':', marker = '^', ms = 7)
          PLT.show()
```



```
In [30]: PLT.plot(Salary[0], c = 'purple', ls = '--', marker = 's', ms = 5)
          PLT.plot(Salary[7], c = 'green', ls = ':', marker = '^', ms = 7)
          PLT.xticks(list(range(0,10)),Seasons)
          PLT.show()
```



```
In [31]: PLT.plot(Salary[0], c = 'purple', ls = '--', marker = 's', ms = 7)
PLT.plot(Salary[7], c = 'green', ls = ':', marker = '^', ms = 7)
PLT.plot(Salary[8], c = 'red', ls = '-', marker = 'h', ms = 7)
PLT.xticks(list(range(0,10)),Seasons)
PLT.show()
```

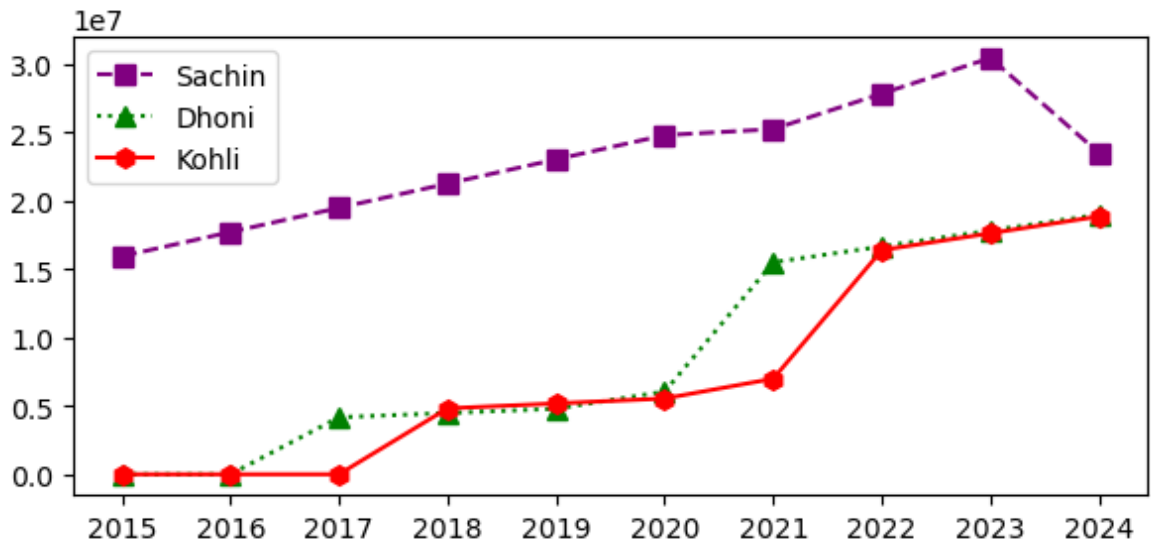


```
In [33]: PLT.plot(Salary[0], c = 'purple', ls = '--', marker = 's', ms = 7, label = Player)
PLT.plot(Salary[7], c = 'green', ls = ':', marker = '^', ms = 7, label = Players[7])
PLT.plot(Salary[8], c = 'red', ls = '-', marker = 'h', ms = 7, label = Players[8])

PLT.legend() #Is used to display the label, Legend is directly proportional to label

PLT.xticks(list(range(0,10)),Seasons)
PLT.show()
```



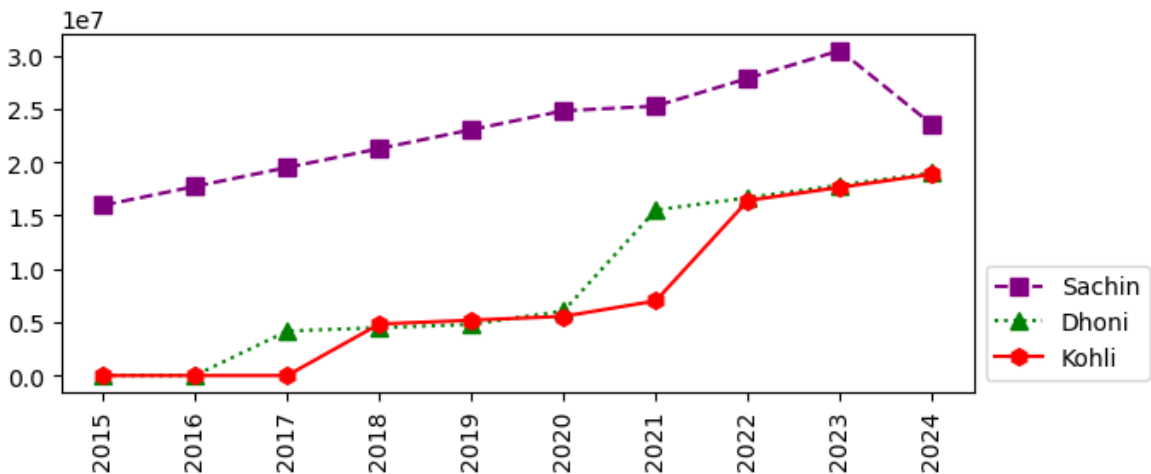


```
In [58]: PLT.plot(Salary[0], c='purple', ls='--', marker='s', ms=7, label=Players[0])
          PLT.plot(Salary[7], c='green', ls=':', marker='^', ms=7, label=Players[7])
          PLT.plot(Salary[8], c='red', ls='-', marker='h', ms=7, label=Players[8])

          # Position the Legend outside the plot at the top left
          PLT.legend(loc="lower left", bbox_to_anchor=(1, 0)) #mode="expand", borderaxespa

          #anchor position 0,0 = axis intersection/bottom left, 0,1 = top left, 1,1 = top

          PLT.xticks(list(range(0, 10)), Seasons, rotation = 'vertical') #rotation for Lab
          #PLT.tight_layout() # Adjust layout to make room for the Legend
          #PLT.subplots_adjust(top=1) # Make space above the plot for the Legend
          PLT.show()
```



In [ ]: