```python
import numpy as np

#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,"2

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

#Salaries
Sachin_Salary = [15946875,17718750,19490625,21262500,23034375,24806250,25244493,278
Rahul_Salary = [12000000,12744189,13488377,14232567,14976754,16324500,18038573,1975
Smith_Salary = [4621800,5828090,13041250,14410581,15779912,14500000,16022500,175450
Sami_Salary = [3713640,4694041,13041250,14410581,15779912,17149243,18518574,1945000
Pollard_Salary = [4493160,4806720,6061274,13758000,15202590,16647180,18091770,19536
Morris_Salary = [3348000,4235220,12455000,14410581,15779912,14500000,16022500,17545
Samson_Salary = [3144240,3380160,3615960,4574189,13520500,14940153,16359805,1777945
Dhoni_Salary = [0,0,4171200,4484040,4796880,6053663,15506632,16669630,17832627,1899
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_

#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

#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_
```

```python
Seasons
```

```
Out[2]: ['2010',
         '2011',
         '2012',
         '2013',
         '2014',
         '2015',
         '2016',
         '2017',
         '2018',
         '2019']
```

```
In [3]: Salary
```

```
Out[3]: 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 [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]: Pdict
```

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

```
In [14]: Salary // Games
```

```
Out[14]: 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]])
```

```
In [13]: np.round(Salary//Games)
```

```
Out[13]:  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]])
```

In [12]:
```python
import warnings
warnings.filterwarnings('ignore')
```
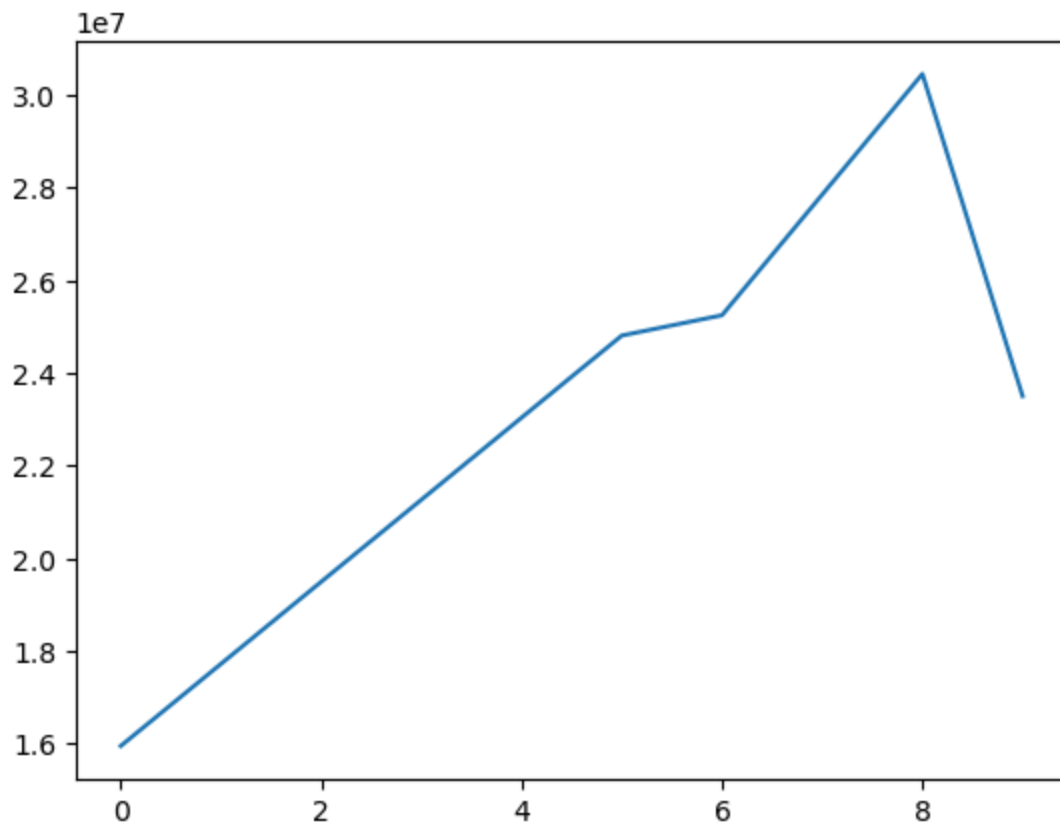
In [15]:
```python
import matplotlib.pyplot as plt # Library used for visualization
```

In [16]:
```python
Salary [0]
```

Out[16]:
```
array([15946875, 17718750, 19490625, 21262500, 23034375, 24806250,
       25244493, 27849149, 30453805, 23500000])
```
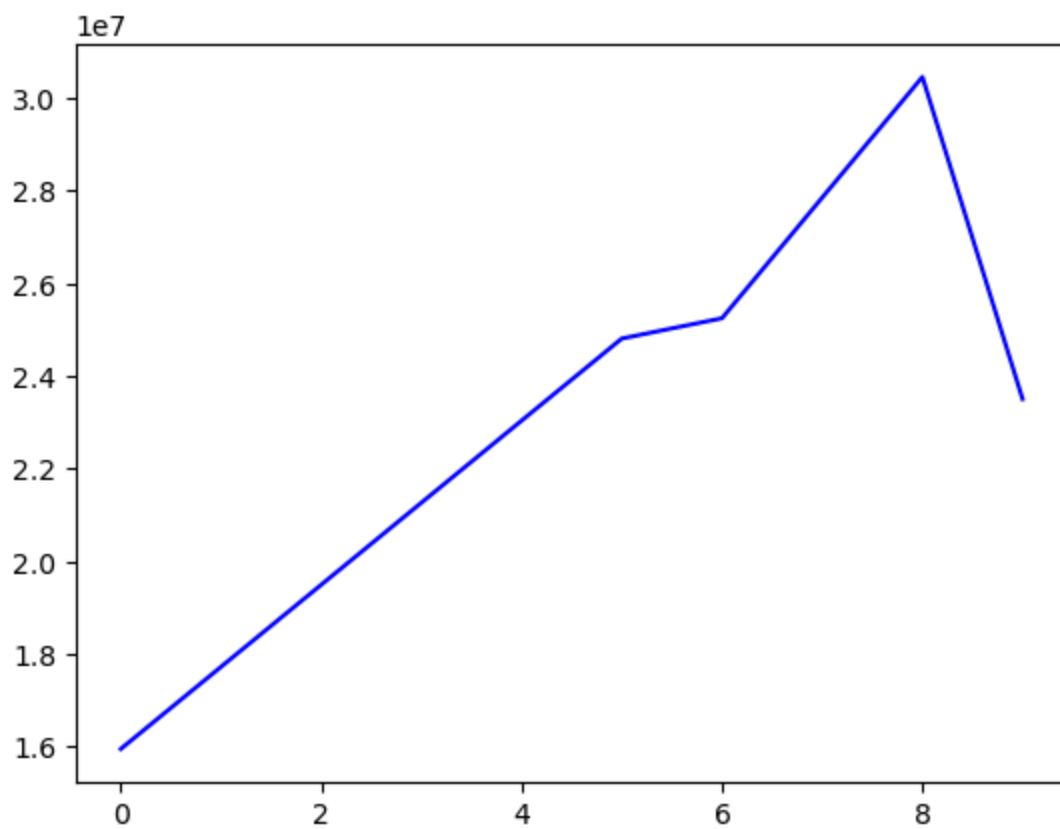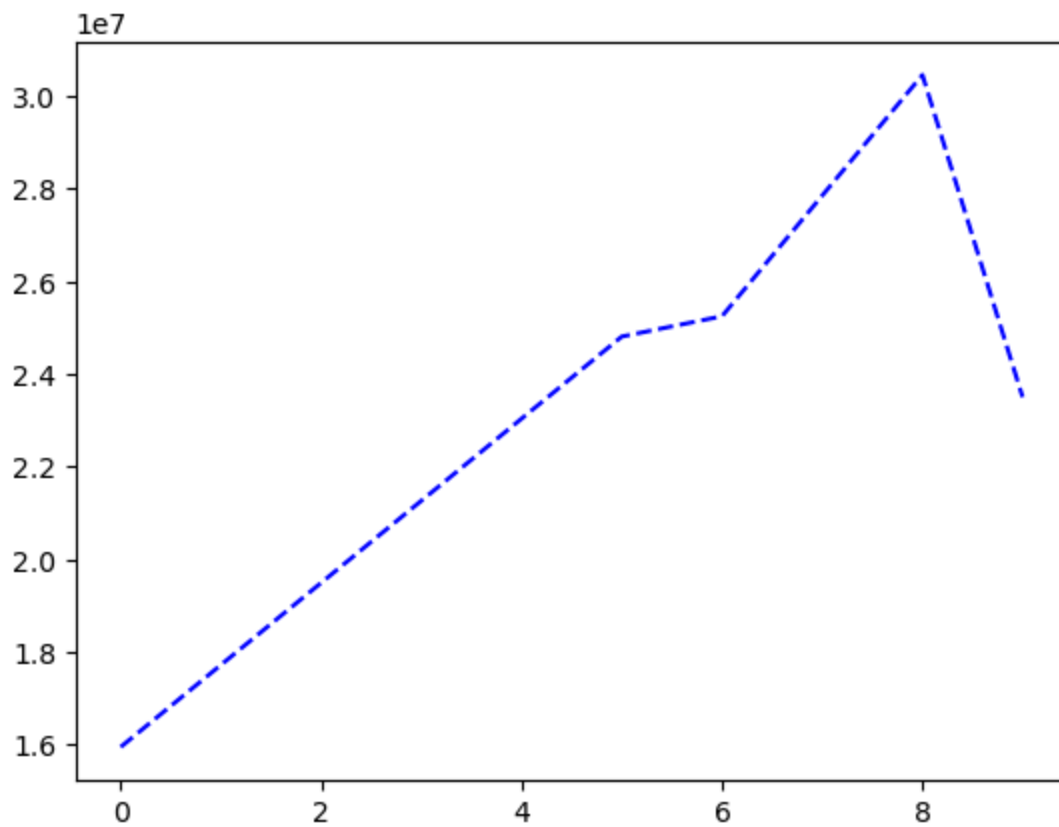
In [17]:
```python
plt.plot(Salary[0])
```

Out[17]: [<matplotlib.lines.Line2D at 0x2119546ccd0>]

In [24]: `plt.plot(Salary[0], c = 'b')`
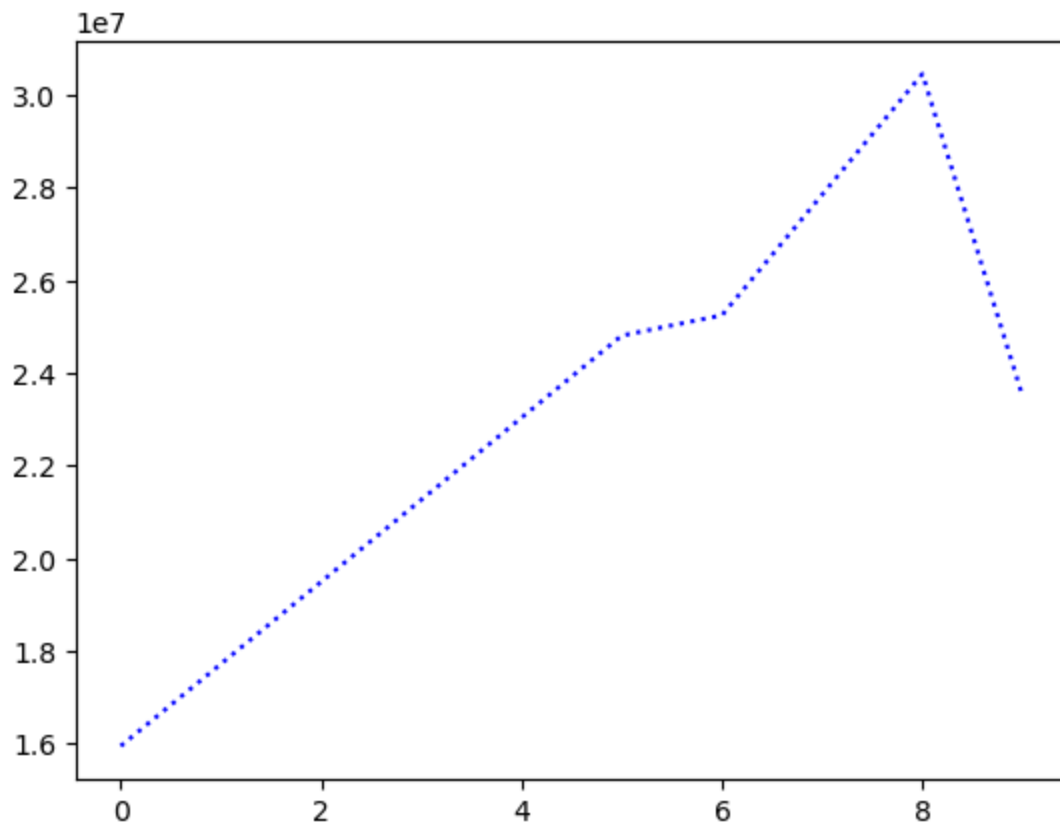
Out[24]: [<matplotlib.lines.Line2D at 0x21196ea7ed0>]

In [25]: `plt.plot(Salary[0], c = 'b', ls = '--')`

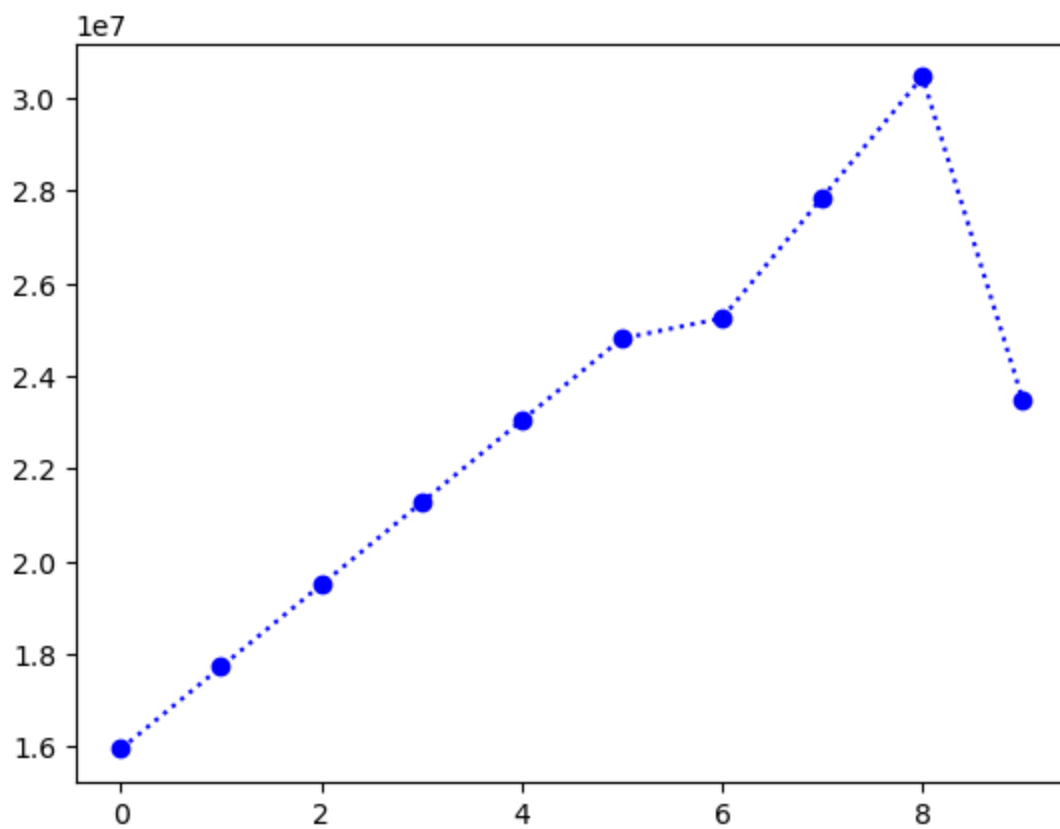Out[25]: `[<matplotlib.lines.Line2D at 0x21196f4e5d0>]`

In [27]: `plt.plot(Salary[0], c = 'b', ls = ':')`

Out[27]: `[<matplotlib.lines.Line2D at 0x2119717b250>]`

```
In [30]: plt.plot(Salary[0], c = 'b', ls = ':', marker = 'o')
```

Out[30]: [<matplotlib.lines.Line2D at 0x2119c693610>]

```
In [31]:  Games[0]
```

```
Out[31]:  array([80, 77, 82, 82, 73, 82, 58, 78,  6, 35])
```

```
In [35]:  %matplotlib inline
          plt.rcParams['figure.figsize'] = (7, 3)
```
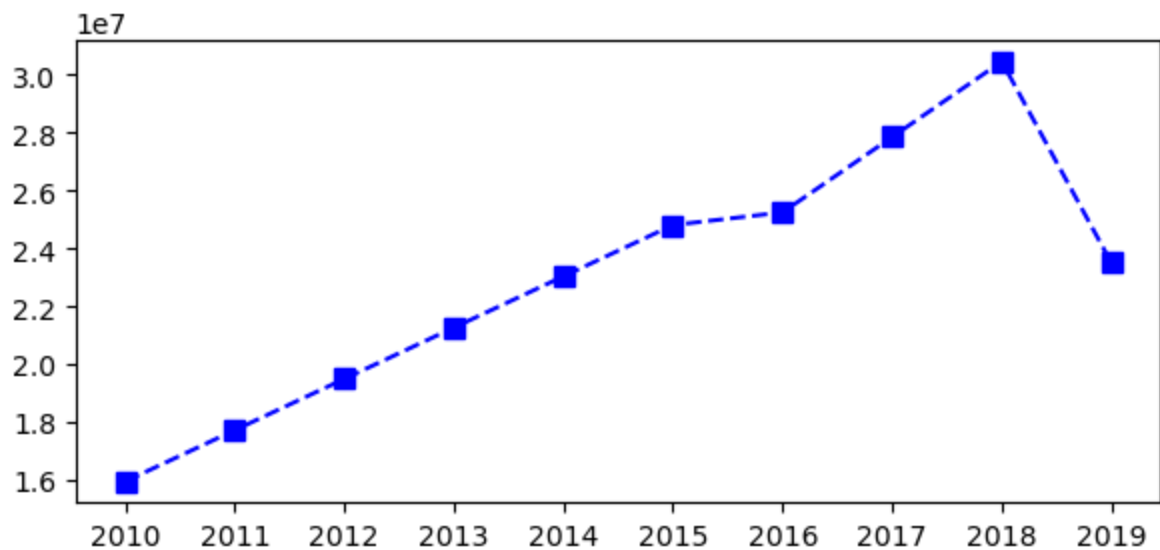
```
In [ ]:   plt.plot(Salary
```

```
In [ ]:
```

```
In [36]:  Sdict
```

```
Out[36]:  {'2010': 0,
           '2011': 1,
           '2012': 2,
           '2013': 3,
           '2014': 4,
           '2015': 5,
           '2016': 6,
           '2017': 7,
           '2018': 8,
           '2019': 9}
```

```
In [38]:  plt.plot(Salary[0], c='blue', ls='--', marker='s', ms=7)
          plt.xticks(list(range(0, 10)), Seasons)
          plt.show()
```
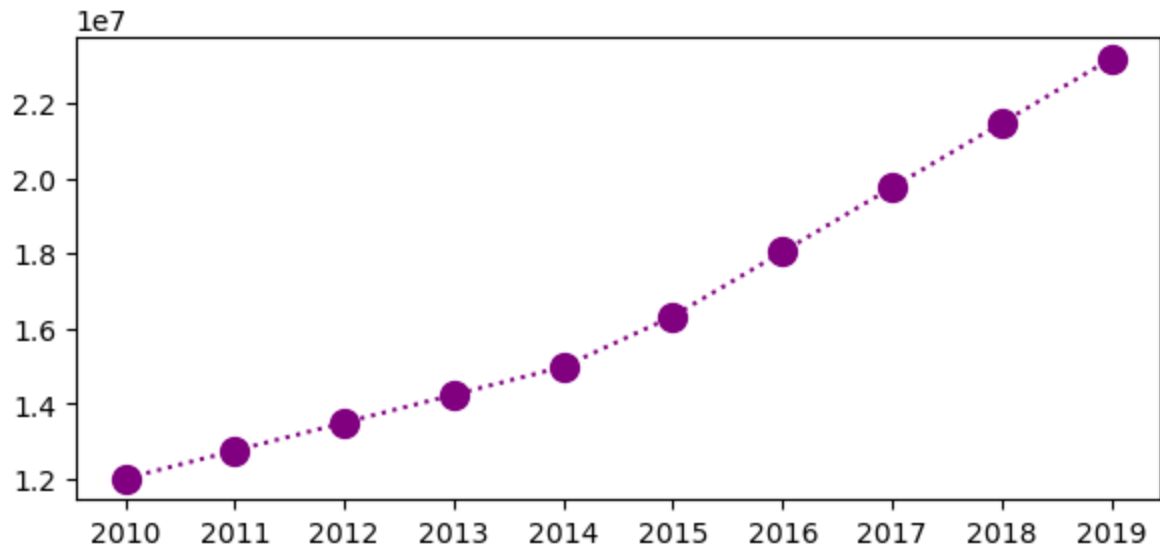


```
In [39]:  Salary[1]
```

```
Out[39]:  array([12000000, 12744189, 13488377, 14232567, 14976754, 16324500,
                 18038573, 19752645, 21466718, 23180790])
```

```
In [51]:  plt.plot(Salary[1], c='purple', ls = ':', marker = 'o', ms = 10, label = Players[0]
          plt.xticks(list(range(0,10)), Seasons,rotation='horizontal')
          plt.show()
```
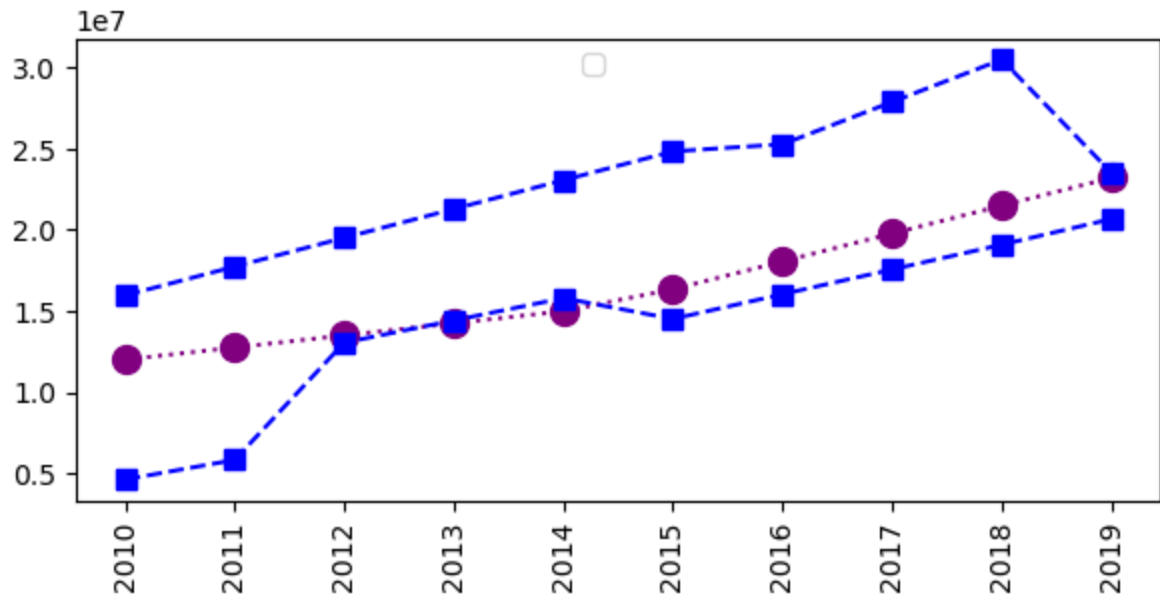
```
In [44]: plt.plot(Salary[1], c='purple', ls = ':', marker = 'o', ms = 10 )
         plt.plot(Salary[0], c='blue', ls='--', marker='s', ms=7)
         plt.plot(Salary[2], c='blue', ls='--', marker='s', ms=7)
         plt.xticks(list(range(0, 10)), Seasons)
         plt.show()
```
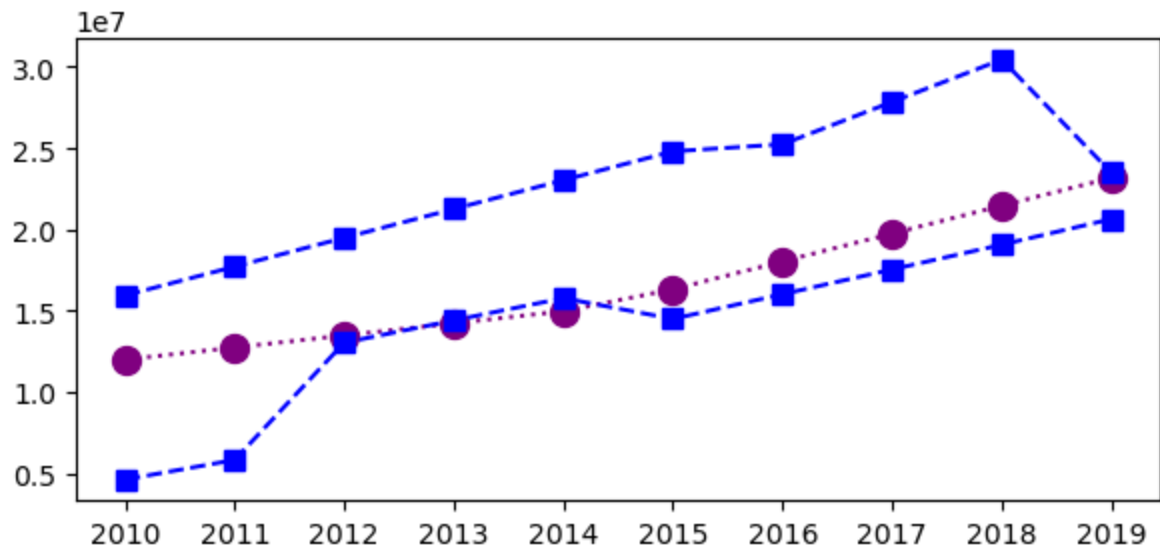


```
In [53]: plt.plot(Salary[1], c='purple', ls=':', marker='o', ms=10)
         plt.plot(Salary[0], c='blue', ls='--', marker='s', ms=7)
         plt.plot(Salary[2], c='blue', ls='--', marker='s', ms=7)

         plt.legend(bbox_to_anchor=(0.5, 1))    # legend positioning
         plt.xticks(list(range(len(Seasons))), Seasons, rotation='vertical')  # tick labels
         plt.show()
```
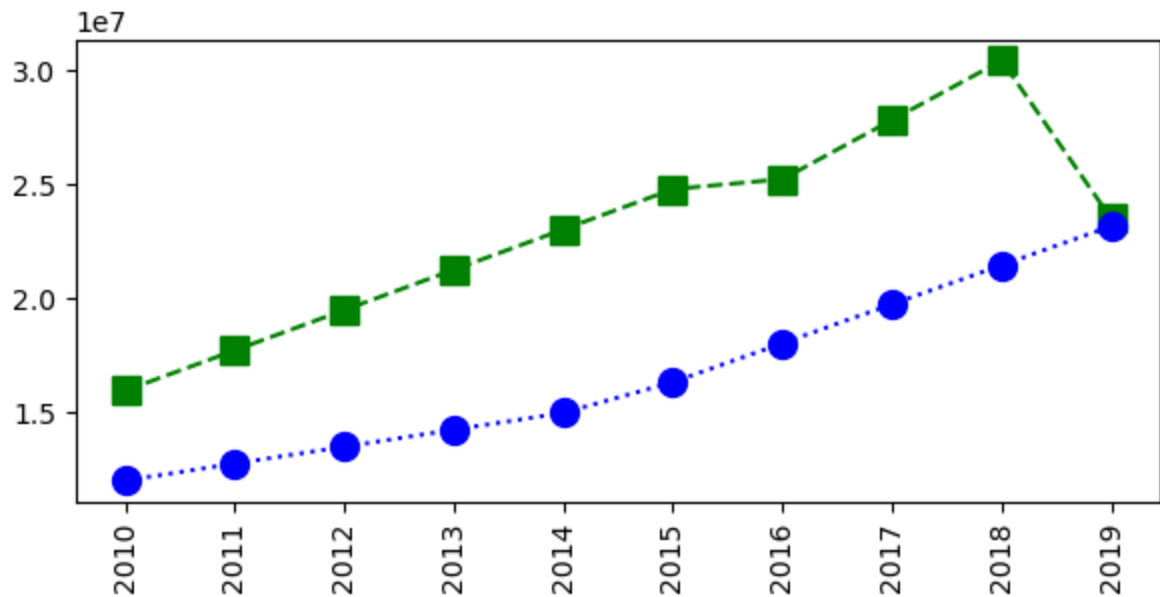
```
In [54]:  plt.plot(Salary[0], c='Green', ls = '--', marker = 's', ms = 10, label = Players[0]
          plt.plot(Salary[1], c='Blue', ls = ':', marker = 'o', ms = 10, label = Players[1])

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

          plt.show()
```
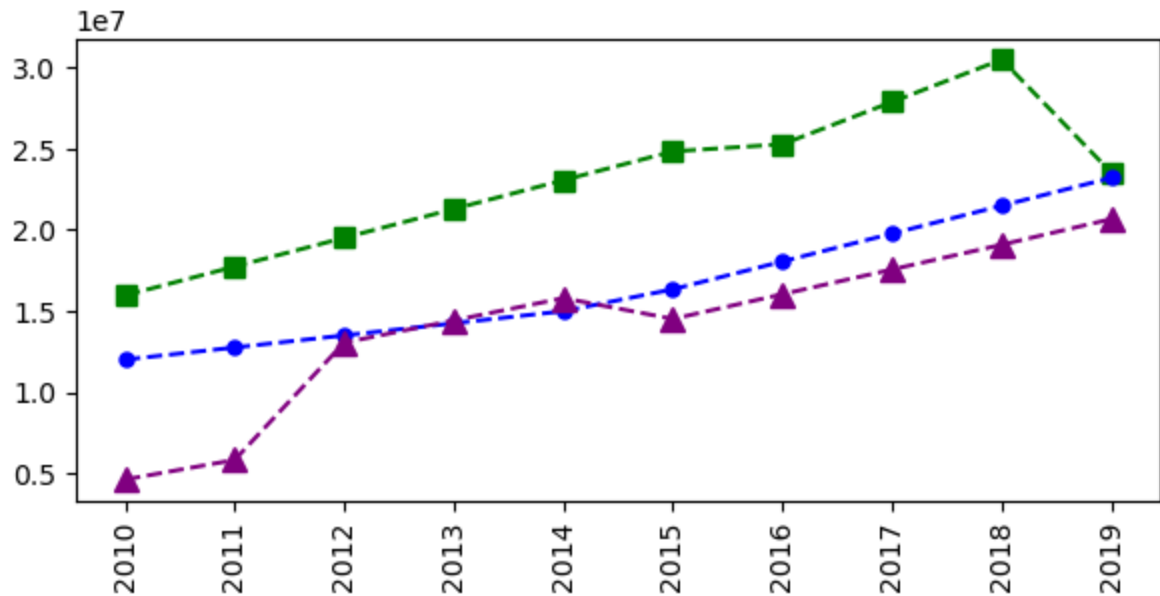


```
In [55]:  plt.plot(Salary[0], c='Green', ls = '--', marker = 's', ms = 7, label = Players[0])
          plt.plot(Salary[1], c='Blue', ls = '--', marker = 'o', ms = 5, label = Players[1])
          plt.plot(Salary[2], c='purple', ls = '--', marker = '^', ms = 8, label = Players[2]

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

          plt.show()
```
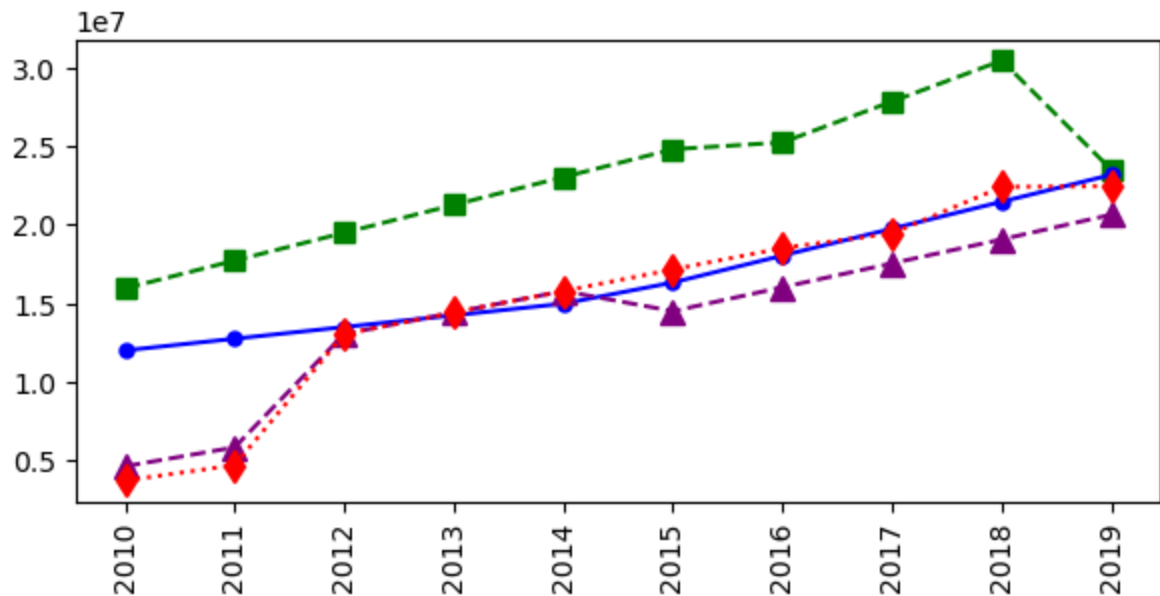
```
In [56]: plt.plot(Salary[0], c='Green', ls = '--', marker = 's', ms = 7, label = Players[0])
         plt.plot(Salary[1], c='Blue', ls = '-', marker = 'o', ms = 5, label = Players[1])
         plt.plot(Salary[2], c='purple', ls = '--', marker = '^', ms = 8, label = Players[2]
         plt.plot(Salary[3], c='Red', ls = ':', marker = 'd', ms = 8, label = Players[3])

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

         plt.show()
```
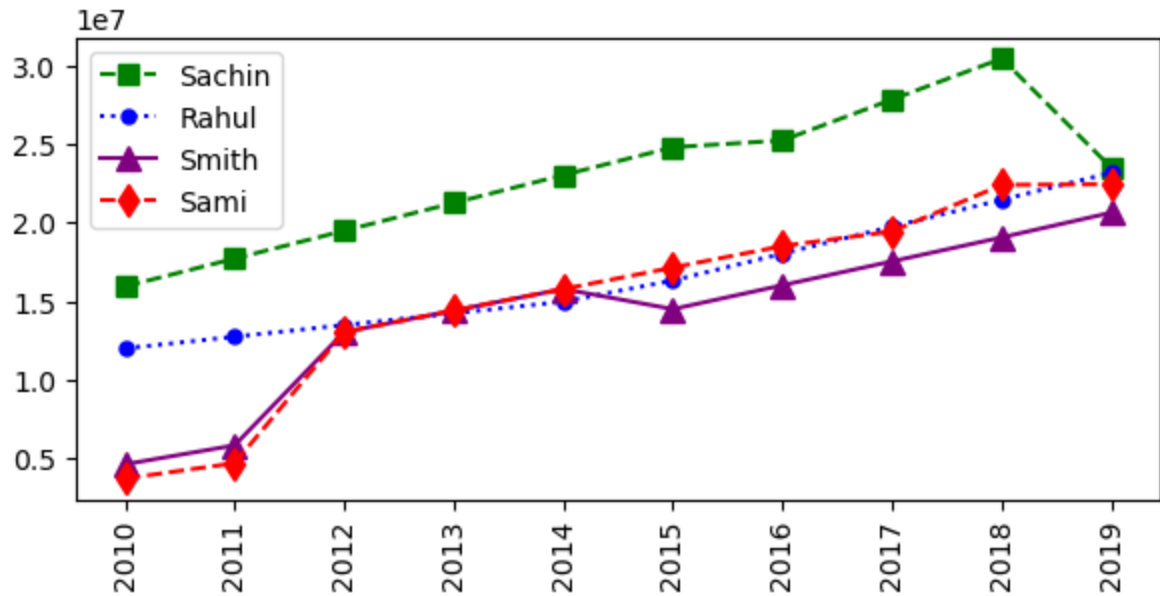


```
In [57]: # how to add legned in visualisation

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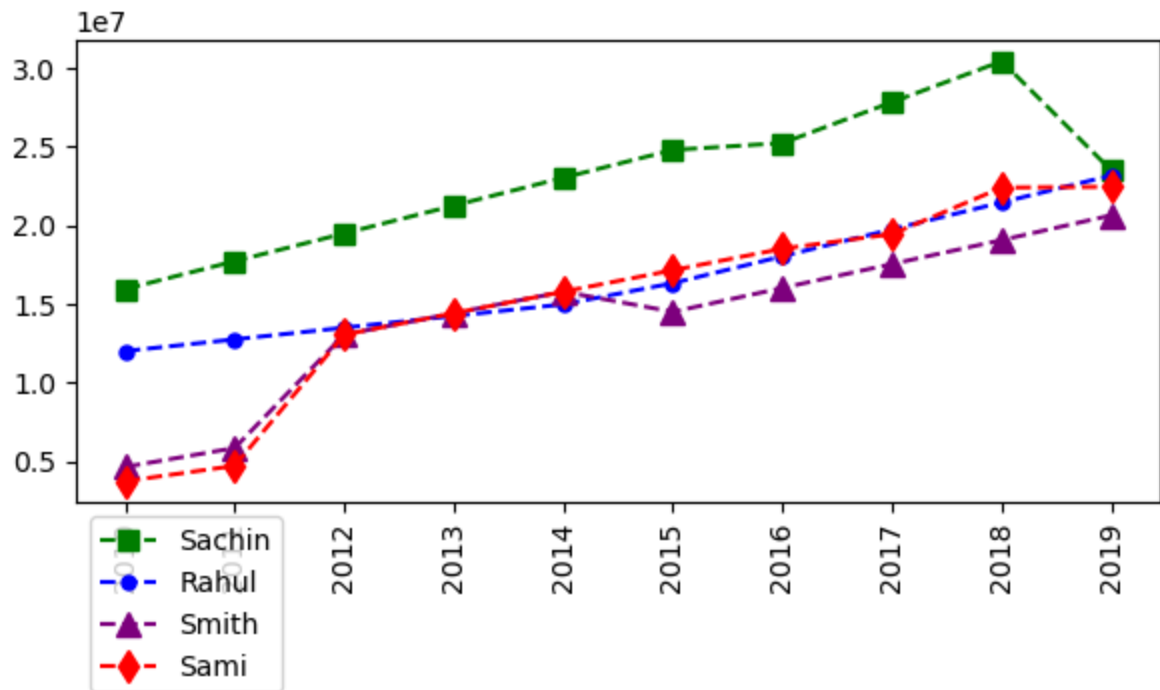
```
plt.show()
```

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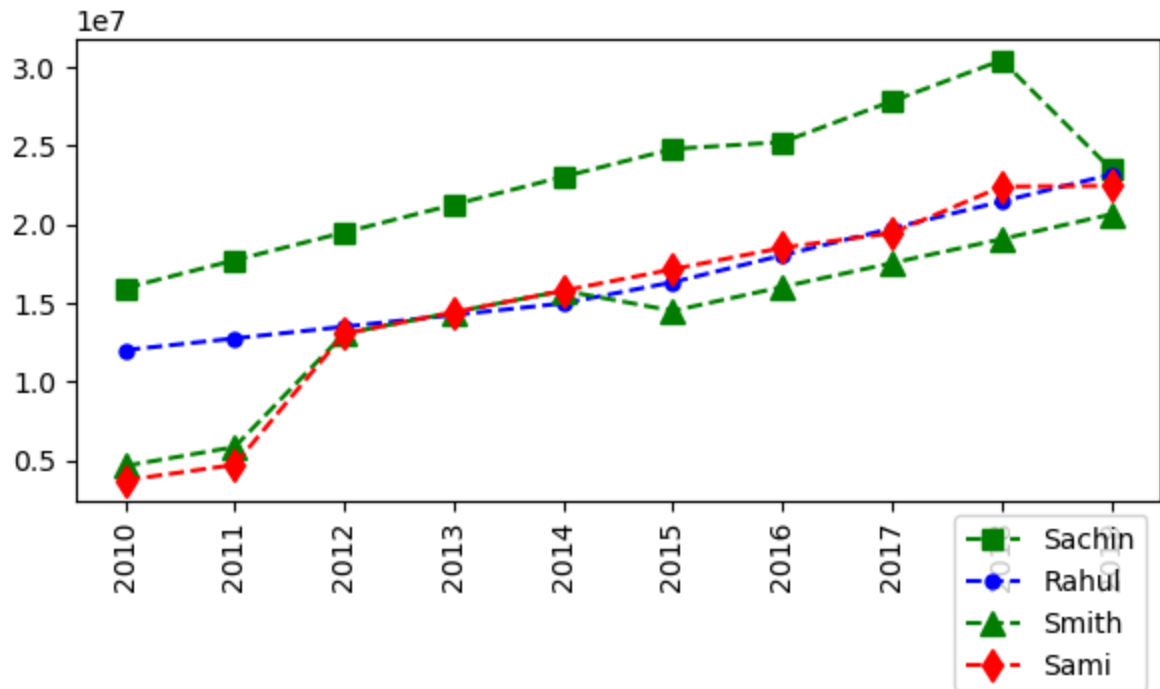
```
plt.plot(Salary[0], c='Green', ls = '--', marker = 's', ms = 7, label = Players[0])
plt.plot(Salary[1], c='Blue', ls = '--', marker = 'o', ms = 5, label = Players[1])
plt.plot(Salary[2], c='Green', ls = '--', marker = '^', ms = 8, label = Players[2])
```

```
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()
```

```
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()
```
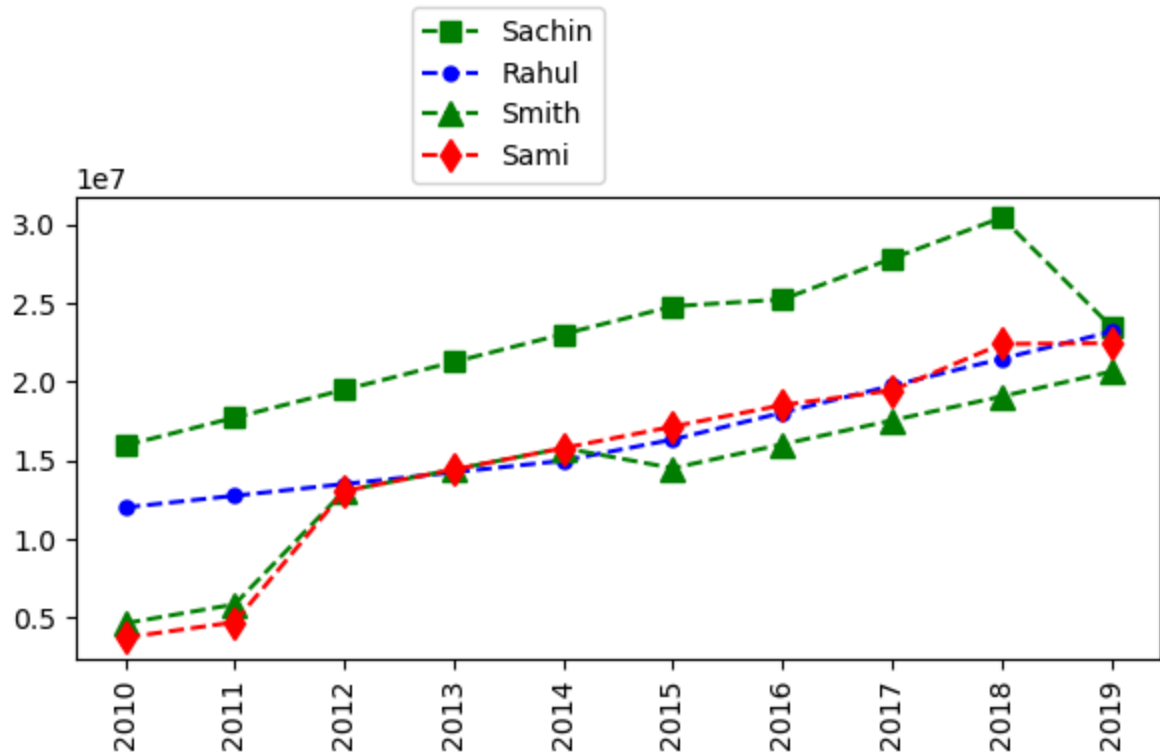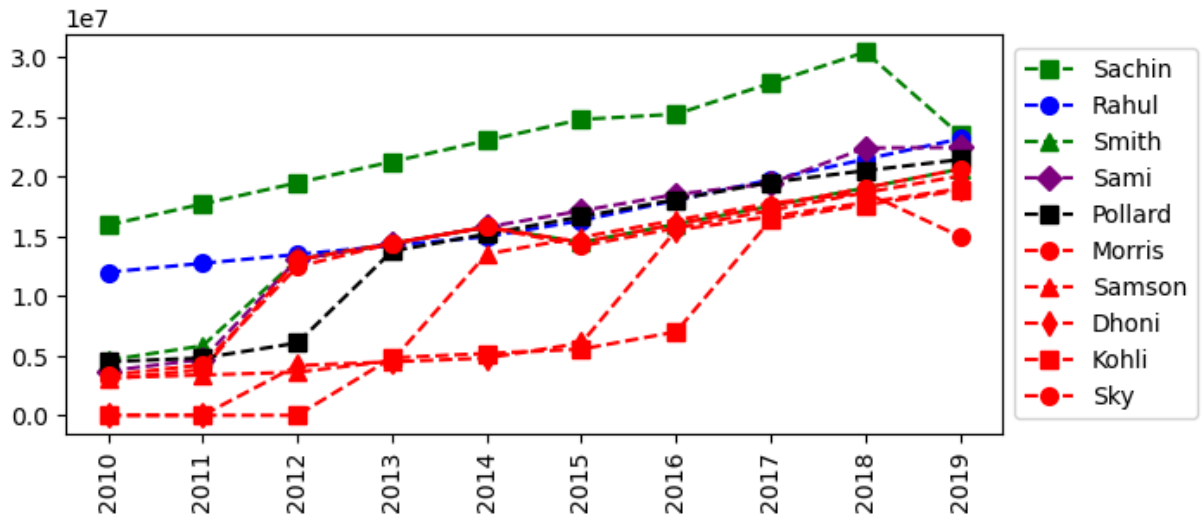
```
In [66]: plt.plot(Salary[0], c='Green', ls='--', marker='s', ms=7, label=Players[0])
         plt.plot(Salary[1], c='Blue', ls='--', marker='o', ms=7, label=Players[1])
         plt.plot(Salary[2], c='Green', ls='--', marker='^', ms=7, label=Players[2])
         plt.plot(Salary[3], c='Purple', ls='--', 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=(1, 0))

         plt.legend(loc='center left', bbox_to_anchor=(1, 0.5)) # this is better option as t

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

```
In [68]:  # 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.legend(loc='center left', bbox_to_anchor=(1, 0.5))

plt.show()
```
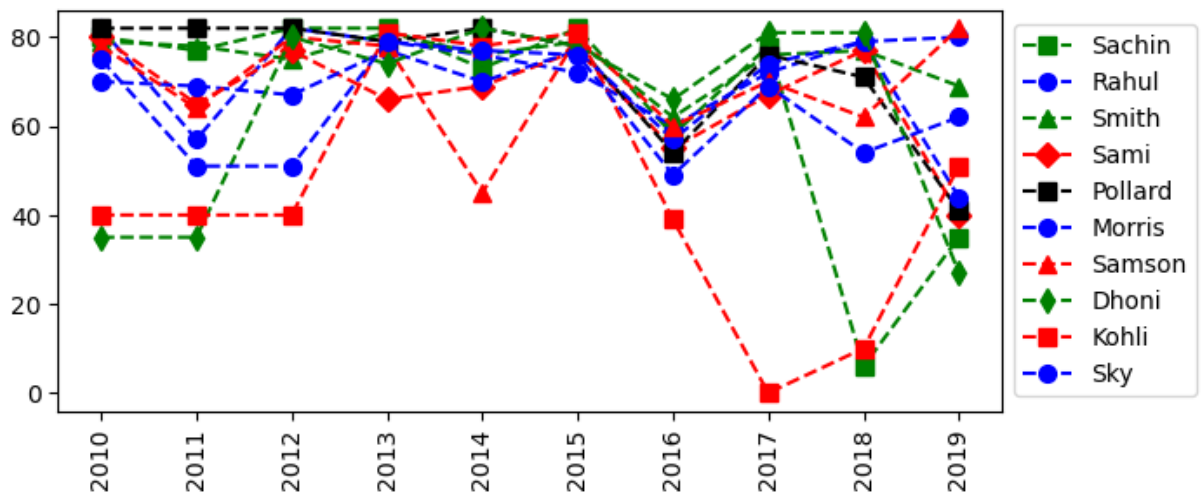

In this section we learned - 1>Matrices
2>Building matrices - np.reshape

3>Dictionaried in python (order doesnot mater) (keys & values)

4>visualizaing using pyplot 5>Basket ball analysis