

```

In [17]: 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

#Players
Players = ["Sachin", "Rahul", "Smith", "Sami", "Pollard", "Morris", "Samson", "Dhoni", "
Pdct = {"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 [18]: Salary

```

```
Out[18]: 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 [19]: Games

```
Out[19]: 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 [20]: Points

```
Out[20]: 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 [21]: Games[1]

```
Out[21]: array([82, 57, 82, 79, 76, 72, 60, 72, 79, 80])
```

In [22]: Games[0:6]

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

```
In [23]: Games[0,6]
```

```
Out[23]: 58
```

## NUMPY+MATPLOTLIB=VISUALIZATION

```
In [24]: Salary/Games
```

```
Out[24]: 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]])
```

```
In [25]: np.round(Salary//Games)
```

```
Out[25]: 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 [26]: import warnings
warnings.filterwarnings('ignore')
```

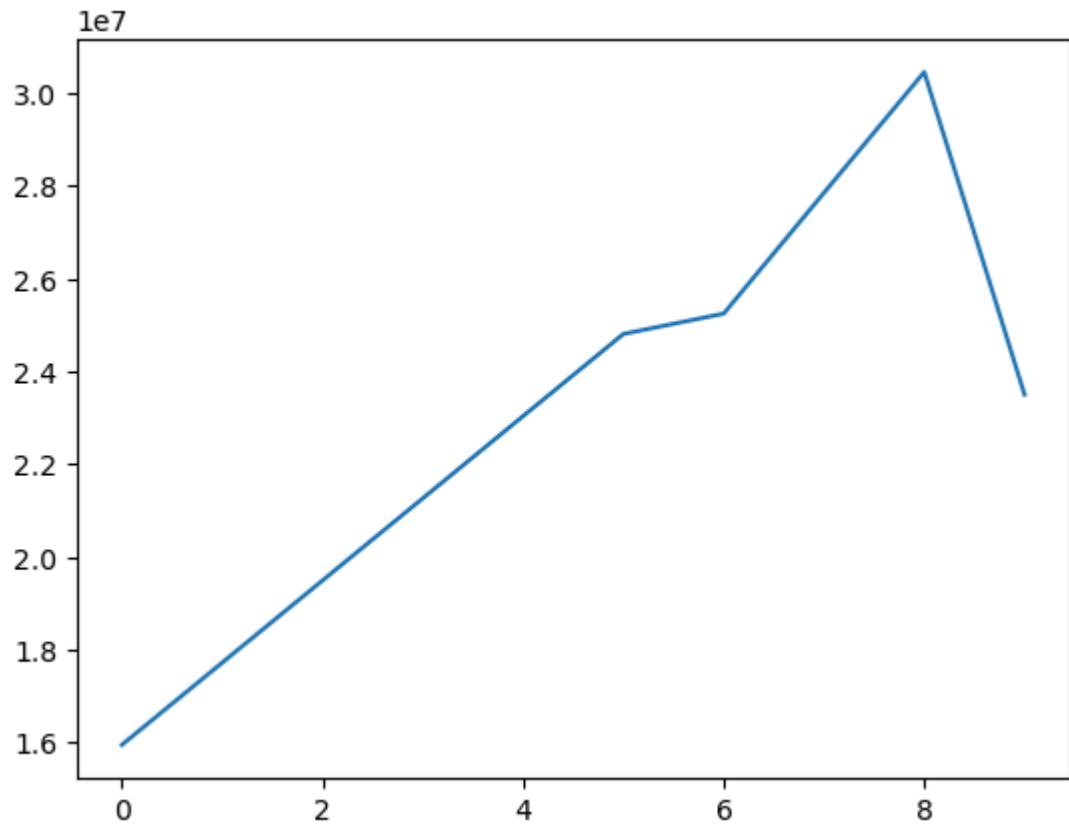
```
In [27]: import matplotlib.pyplot as plt
import numpy as np
```

```
In [28]: Salary[0]
```

```
Out[28]: array([15946875, 17718750, 19490625, 21262500, 23034375, 24806250,
                25244493, 27849149, 30453805, 23500000])
```

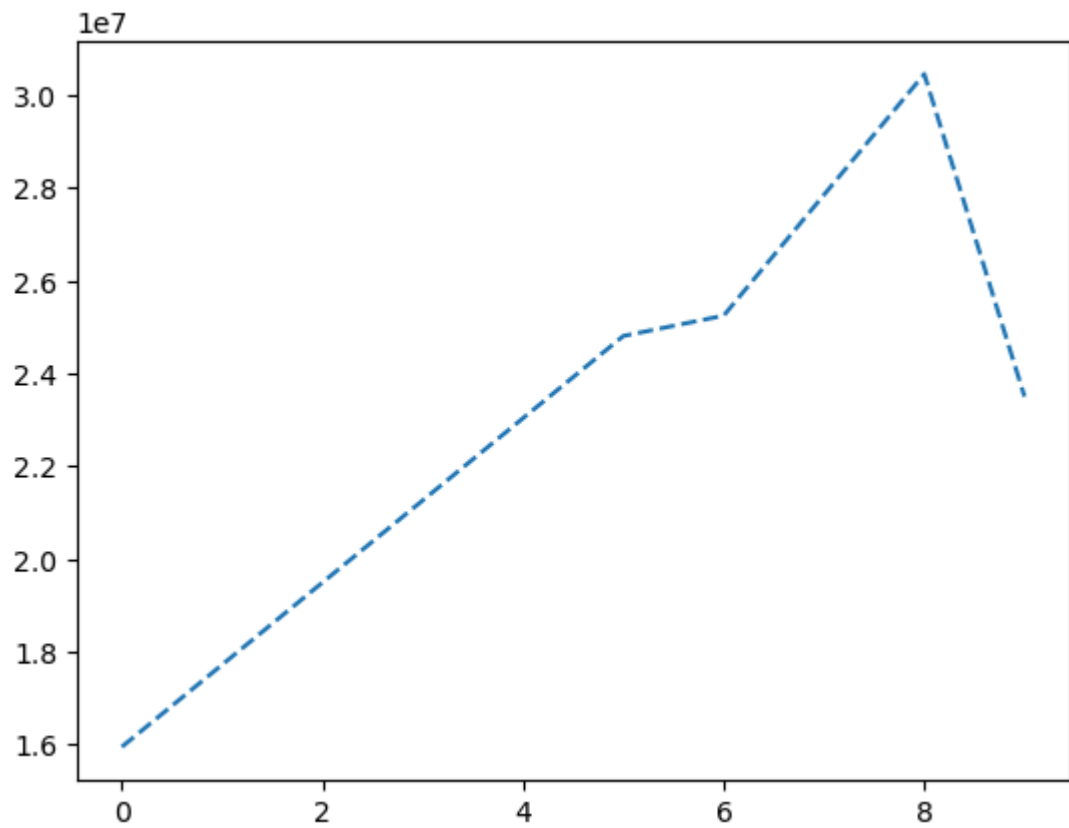
```
In [29]: plt.plot(Salary[0])
```

```
Out[29]: [<matplotlib.lines.Line2D at 0x163c5fd2930>]
```



```
In [30]: plt.plot(Salary[0], ls = '--') # use shift tab for more clarity
```

```
Out[30]: [<matplotlib.lines.Line2D at 0x163c60ca810>]
```



```
In [31]: plt.plot(Salary[0], ls = '.-')
```

```

-----
ValueError                                Traceback (most recent call last)
Cell In[31], line 1
----> 1 plt.plot(Salary[0], ls = '-.-')

File ~\anaconda3\Lib\site-packages\matplotlib\pyplot.py:3794, in plot(scalex, scaley, data, *args, **kwargs)
    3786 @_copy_docstring_and_deprecators(Axes.plot)
    3787 def plot(
    3788     *args: float | ArrayLike | str,
    (...)
    3792     **kwargs,
    3793 ) -> list[Line2D]:
-> 3794     return gca().plot(
    3795         *args,
    3796         scalex=scalex,
    3797         scaley=scaley,
    3798         **({"data": data} if data is not None else {}),
    3799         **kwargs,
    3800     )

File ~\anaconda3\Lib\site-packages\matplotlib\axes\_axes.py:1779, in Axes.plot(self, scalex, scaley, data, *args, **kwargs)
    1536 """
    1537 Plot y versus x as lines and/or markers.
    1538 (...)
    1776 (``'green'``) or hex strings (``'#008000'``).
    1777 """
    1778 kwargs = cbook.normalize_kwargs(kwargs, mlines.Line2D)
-> 1779 lines = [*self._get_lines(self, *args, data=data, **kwargs)]
    1780 for line in lines:
    1781     self.add_line(line)

File ~\anaconda3\Lib\site-packages\matplotlib\axes\_base.py:296, in _process_plot_var_args.__call__(self, axes, data, *args, **kwargs)
    294     this += args[0],
    295     args = args[1:]
--> 296 yield from self._plot_args(
    297     axes, this, kwargs, ambiguous_fmt_datakey=ambiguous_fmt_datakey)

File ~\anaconda3\Lib\site-packages\matplotlib\axes\_base.py:534, in _process_plot_var_args._plot_args(self, axes, tup, kwargs, return_kwargs, ambiguous_fmt_datakey)
    532     return list(result)
    533 else:
--> 534     return [l[0] for l in result]

File ~\anaconda3\Lib\site-packages\matplotlib\axes\_base.py:527, in <genexpr>(.0)
    522 else:
    523     raise ValueError(
    524         f"label must be scalar or have the same length as the input "
    525         f"data, but found {len(label)} for {n_datasets} datasets.")
--> 527 result = (make_artist(axes, x[:, j % ncx], y[:, j % ncy], kw,
    528                     **kwargs, 'label': label))
    529     for j, label in enumerate(labels))
    531 if return_kwargs:
    532     return list(result)

File ~\anaconda3\Lib\site-packages\matplotlib\axes\_base.py:335, in _process_plot

```

```

_var_args._makeline(self, axes, x, y, kw, kwargs)
    333 kw = {**kw, **kwargs} # Don't modify the original kw.
    334 self._setdefaults(self._getdefaults(kw), kw)
--> 335 seg = mlines.Line2D(x, y, **kw)
    336 return seg, kw

```

```

File ~\anaconda3\Lib\site-packages\matplotlib\lines.py:372, in Line2D.__init__(self, xdata, ydata, linewidth, linestyle, color, gapcolor, marker, markersize, markeredgewidth, markeredgewidth, markerfacecolor, markerfacecoloralt, fillstyle, antialiased, dash_capstyle, solid_capstyle, dash_joinstyle, solid_joinstyle, pickradius, drawstyle, markevery, **kwargs)
    369 self._dash_pattern = (0, None) # offset, dash (scaled by linewidth)
    371 self.set_linewidth(linewidth)
--> 372 self.set_linestyle(linestyle)
    373 self.set_drawstyle(drawstyle)
    375 self._color = None

```

```

File ~\anaconda3\Lib\site-packages\matplotlib\lines.py:1177, in Line2D.set_linestyle(self, ls)
    1175 if ls in [' ', '', 'none']:
    1176     ls = 'None'
-> 1177 _api.check_in_list([*self._lineStyles, *ls_mapper_r], ls=ls)
    1178 if ls not in self._lineStyles:
    1179     ls = ls_mapper_r[ls]

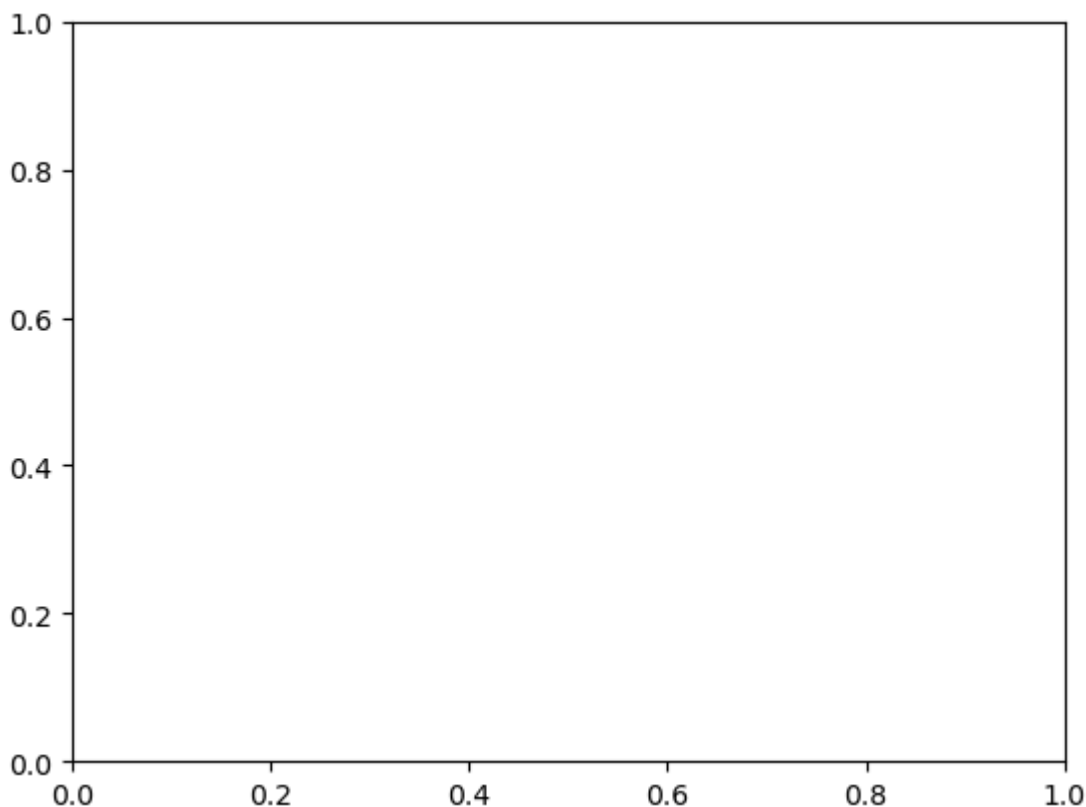
```

```

File ~\anaconda3\Lib\site-packages\matplotlib\_api\__init__.py:129, in check_in_list(values, _print_supported_values, **kwargs)
    127 if _print_supported_values:
    128     msg += f"; supported values are {'', '.join(map(repr, values))}"
--> 129 raise ValueError(msg)

```

**ValueError:** '.-' is not a valid value for ls; supported values are '-', '--', '-.', ':', 'None', ' ', '', 'solid', 'dashed', 'dashdot', 'dotted'



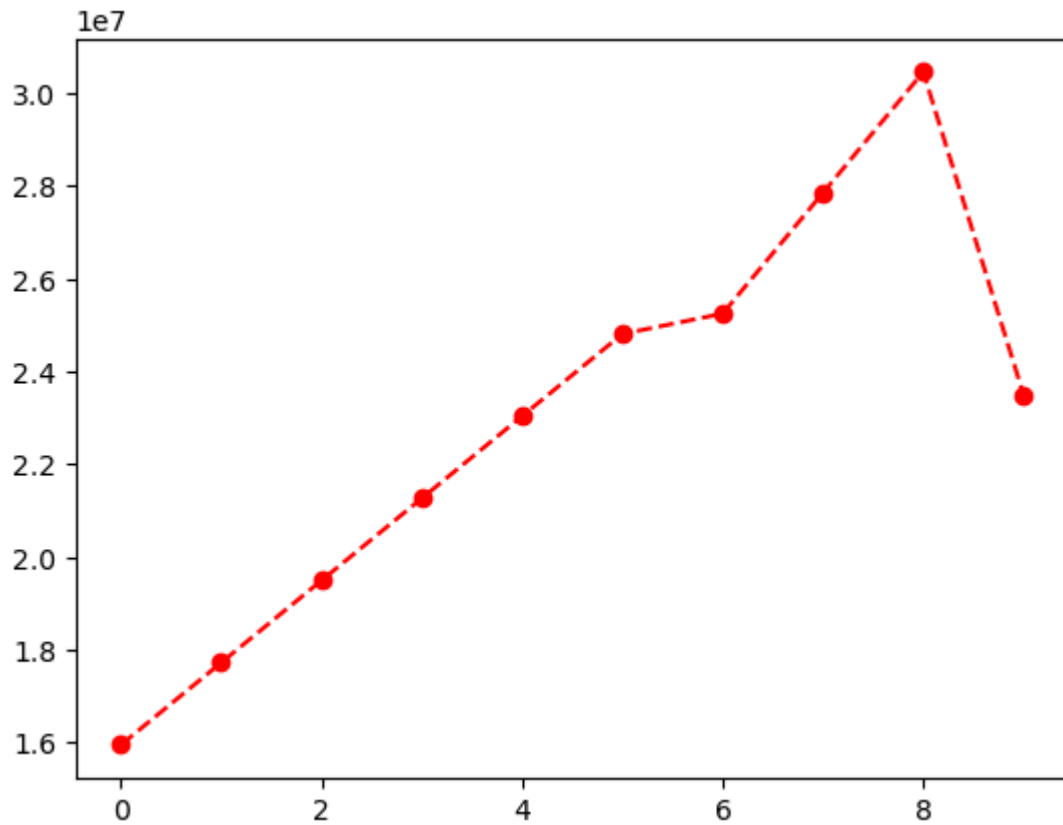
```
In [ ]: plt.plot(Salary[0], ls = '-.')
```

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

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

```
In [32]: plt.plot(Salary[0], ls = '--', color = 'red', marker= 'o')
```

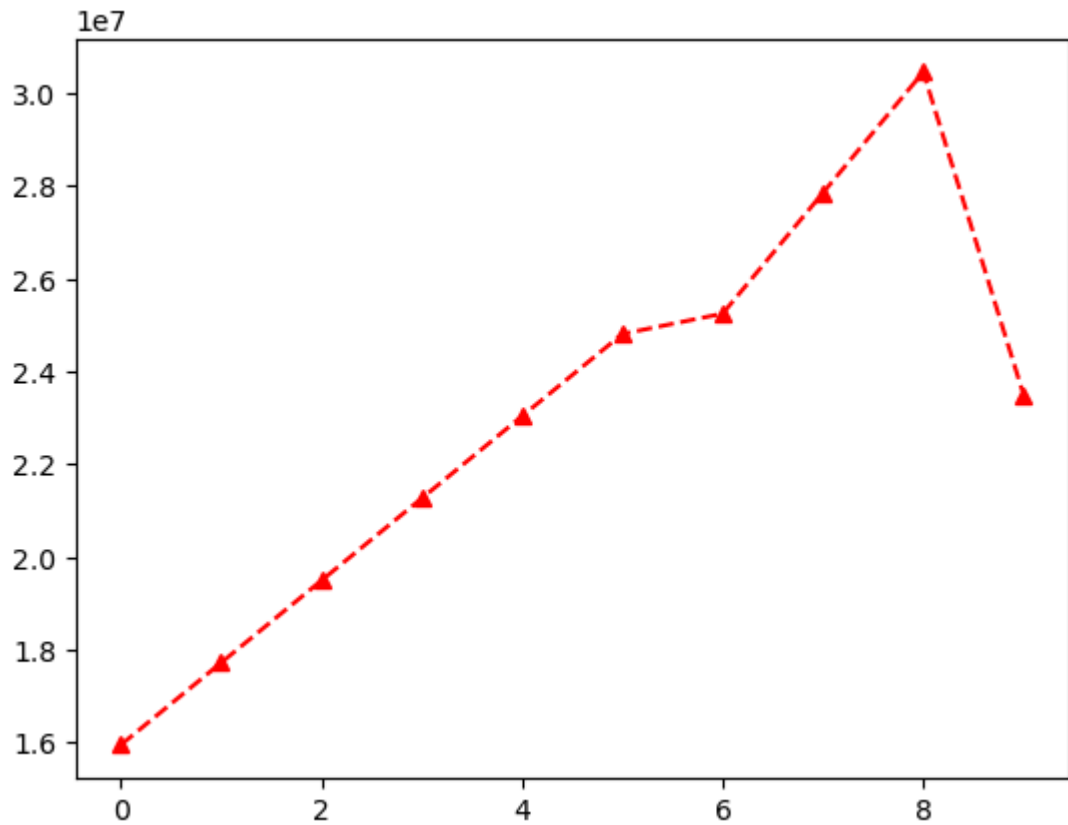
```
Out[32]: [<matplotlib.lines.Line2D at 0x163c7229f40>]
```



```
In [33]: plt.plot(Salary[0], ls = '--', color = 'red', marker= '^')
```

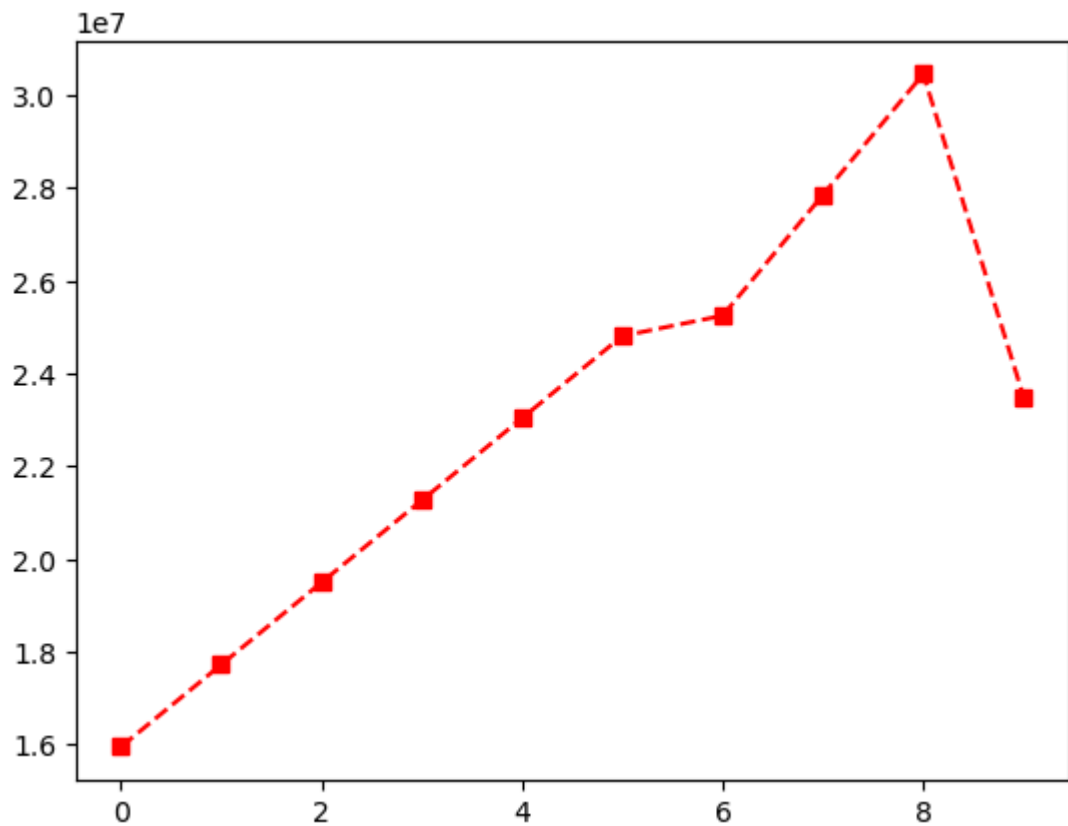
```
Out[33]: [<matplotlib.lines.Line2D at 0x163c7286c30>]
```





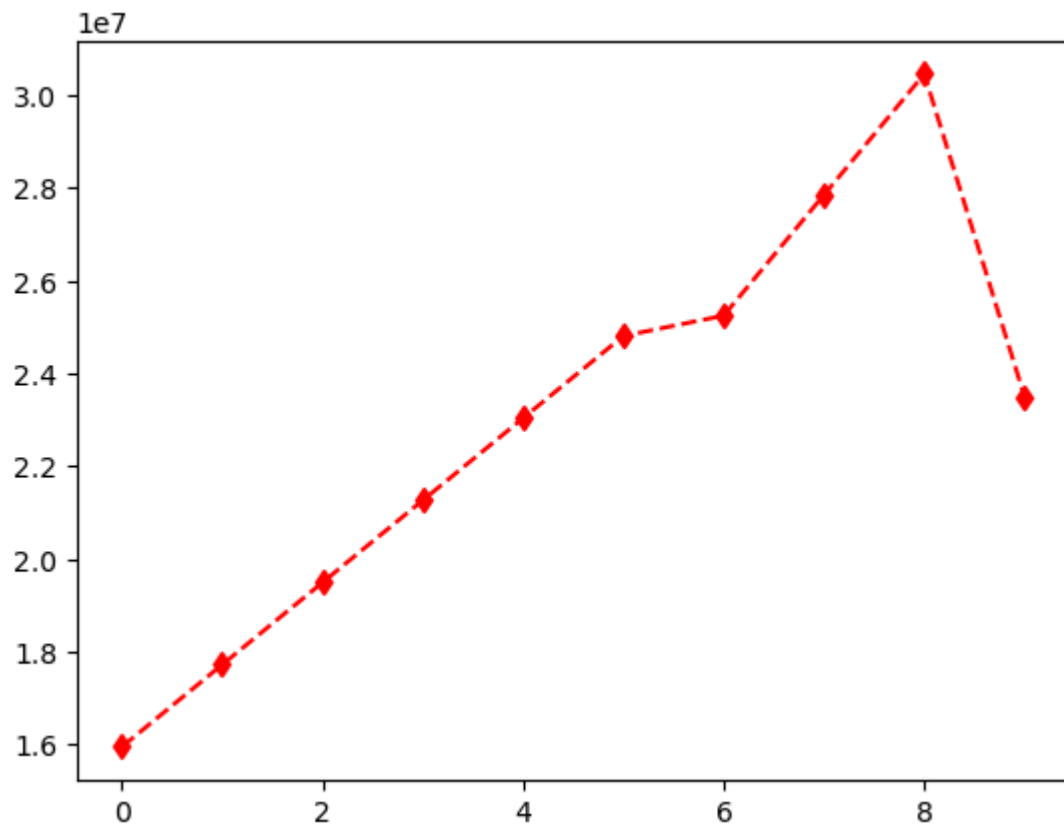
```
In [34]: plt.plot(Salary[0], ls = '--', color = 'red', marker= 's')
```

```
Out[34]: [<matplotlib.lines.Line2D at 0x163c72e3770>]
```



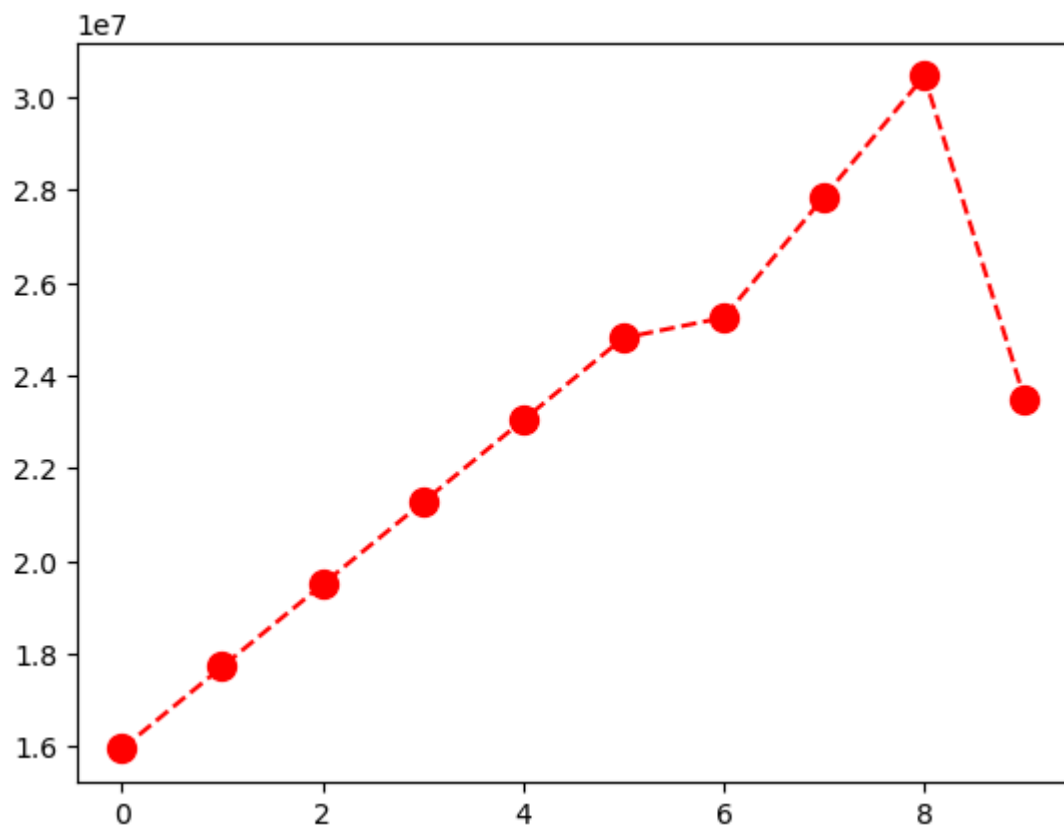
```
In [35]: plt.plot(Salary[0], ls = '--', color = 'red', marker= 'd')
```

```
Out[35]: [<matplotlib.lines.Line2D at 0x163c73377a0>]
```



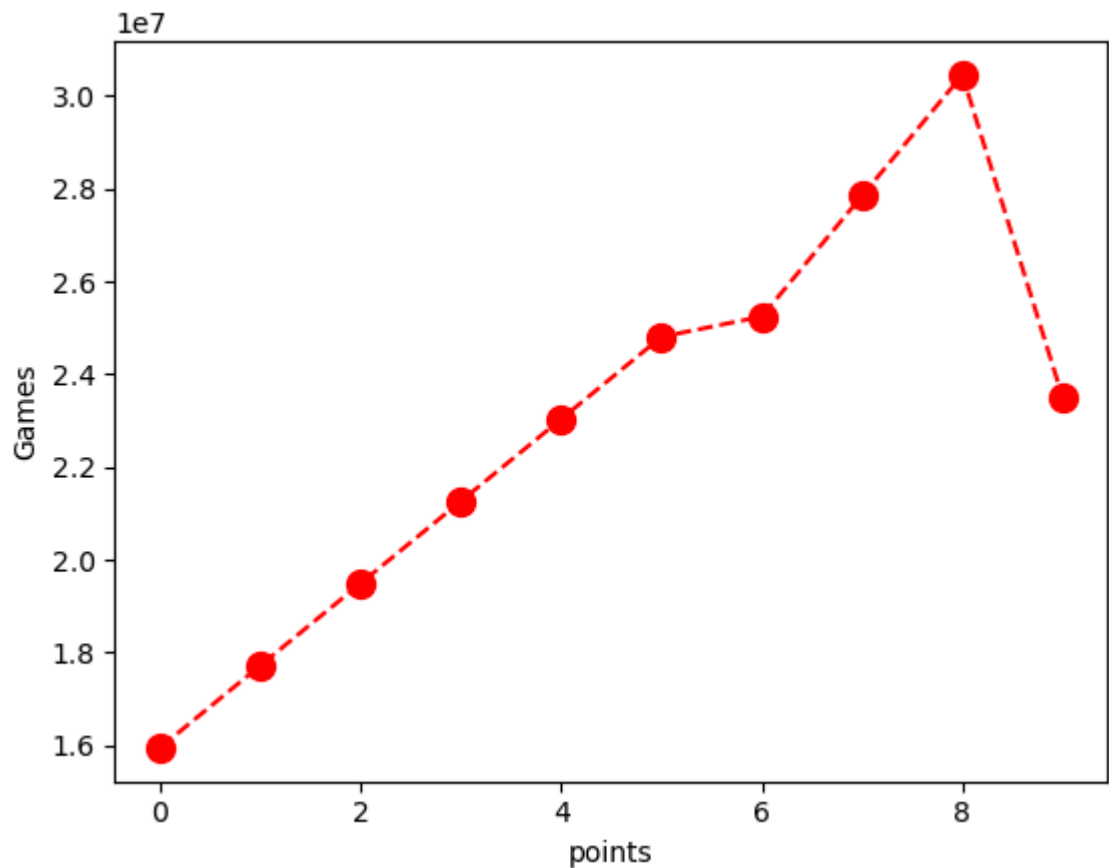
```
In [36]: plt.plot(Salary[0], ls = '--', color = 'red', marker= 'o', ms=10)
```

```
Out[36]: [<matplotlib.lines.Line2D at 0x163c73b6c00>]
```



```
In [37]: plt.plot(Salary[0], ls = '--', color = 'red', marker= 'o', ms=10)
plt.xlabel('points')
plt.ylabel('Games')
```

Out[37]: Text(0, 0.5, 'Games')



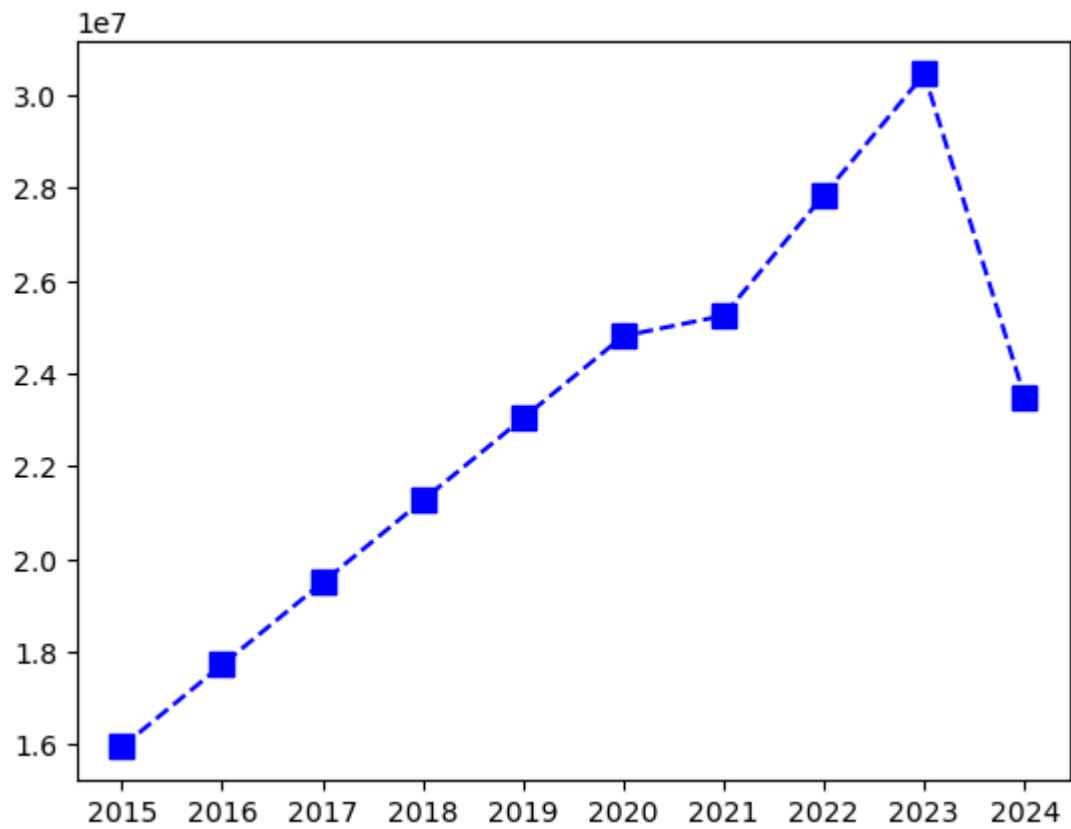
In [39]: Sdict

```
Out[39]: {'2015': 0,
          '2016': 1,
          '2017': 2,
          '2018': 3,
          '2019': 4,
          '2020': 5,
          '2021': 6,
          '2022': 7,
          '2023': 8,
          '2024': 9}
```

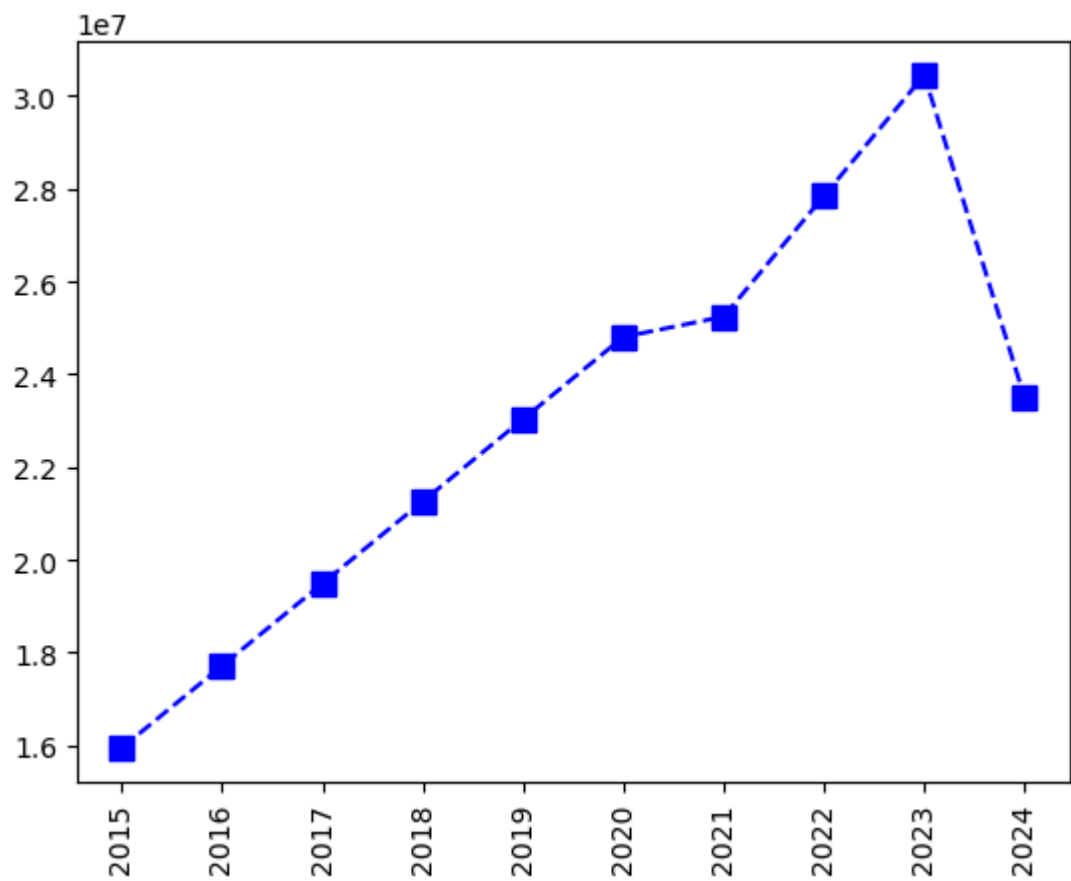
In [40]: Pdict

```
Out[40]: {'Sachin': 0,
          'Rahul': 1,
          'Smith': 2,
          'Sami': 3,
          'Pollard': 4,
          'Morris': 5,
          'Samson': 6,
          'Dhoni': 7,
          'Kohli': 8,
          'Sky': 9}
```

```
In [43]: plt.plot(Salary[0], c = 'blue', ls='--', marker= 's', ms=8)
plt.xticks(list(range(0,10)), Seasons) # xticks ->x axis
plt.show()
```



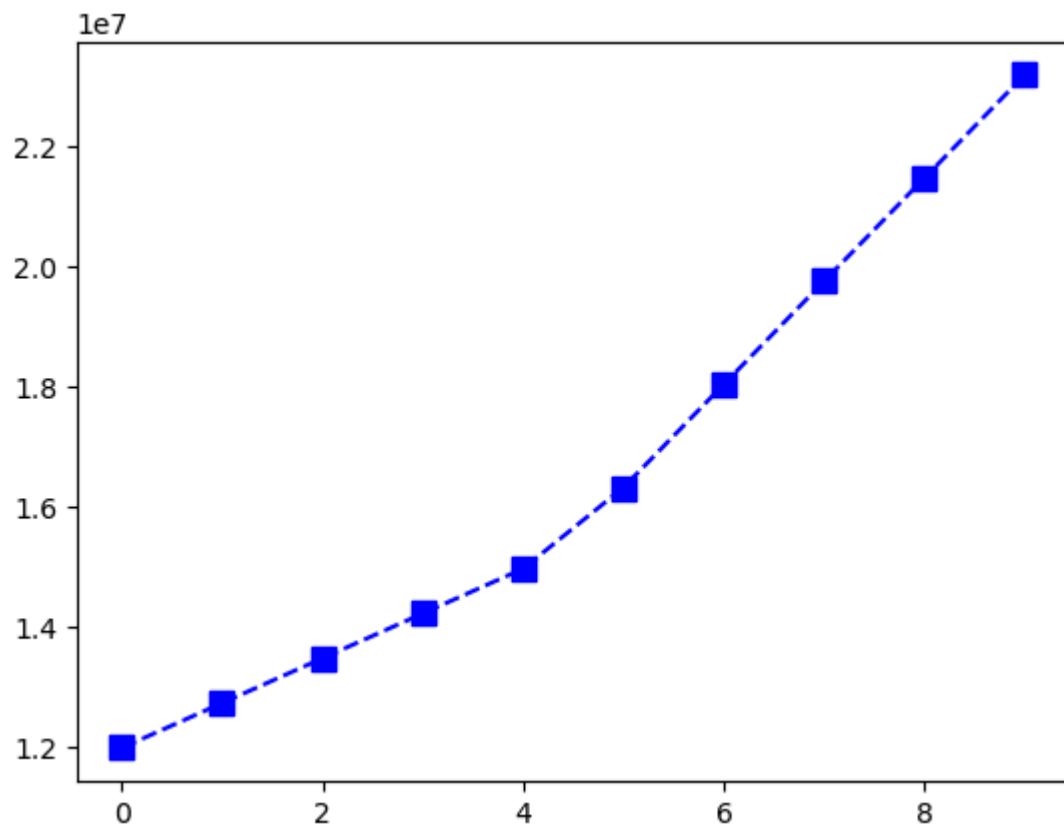
```
In [44]: plt.plot(Salary[0], c = 'blue', ls='--', marker= 's', ms=8)
plt.xticks(list(range(0,10)), Seasons, rotation='vertical') # rotation='vertical'
plt.show()
```



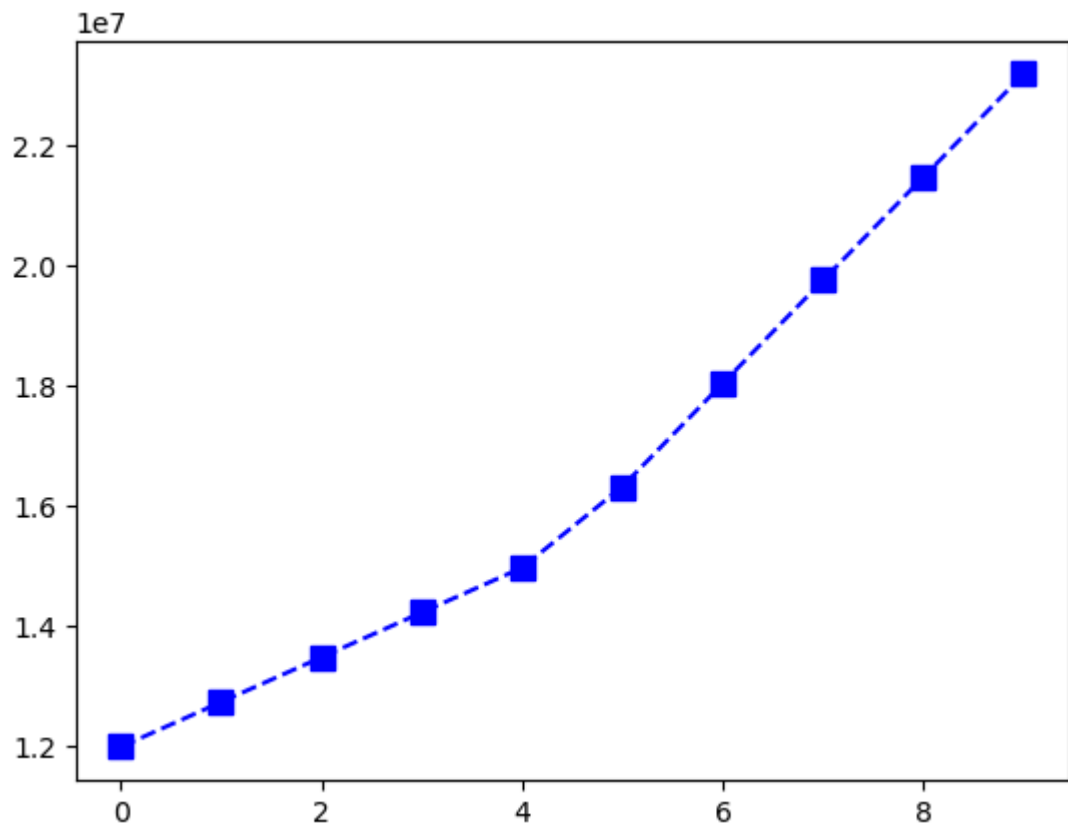
```
In [45]: Salary[1]
```

```
Out[45]: array([12000000, 12744189, 13488377, 14232567, 14976754, 16324500,  
              18038573, 19752645, 21466718, 23180790])
```

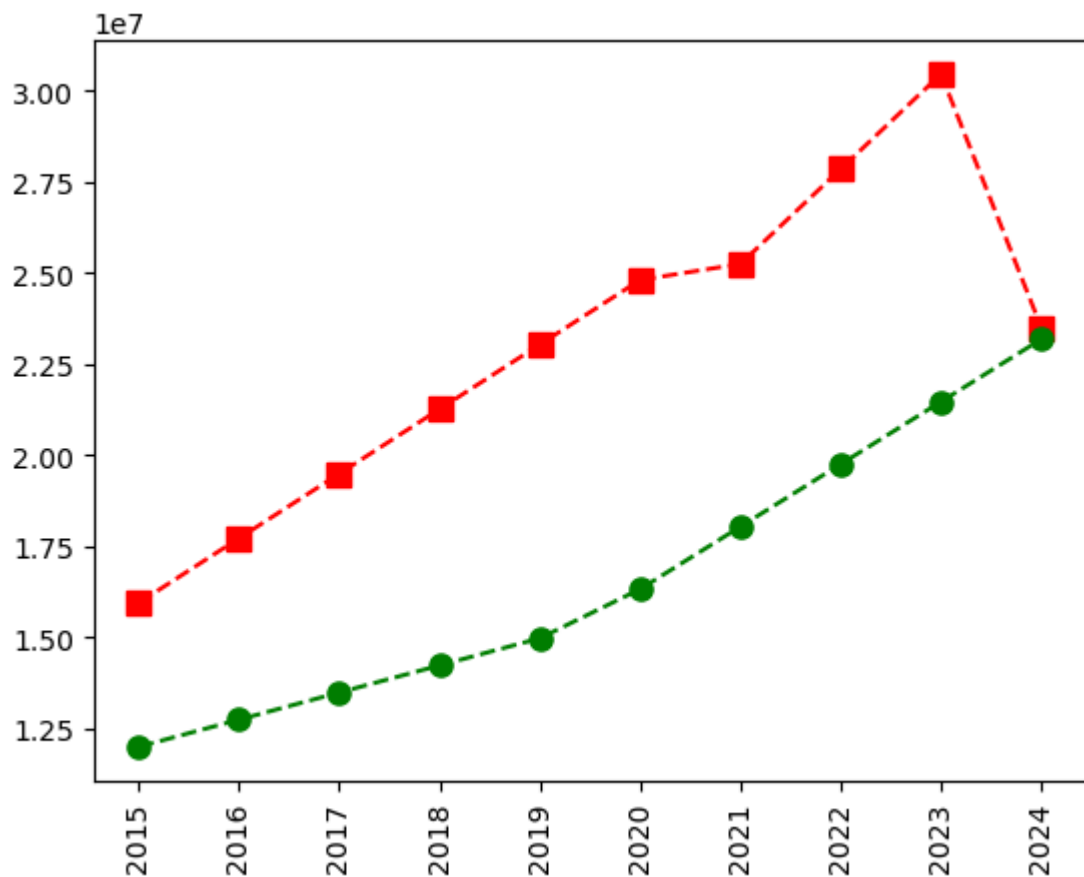
```
In [46]: plt.plot(Salary[1], c = 'blue', ls='--', marker= 's', ms=8)  
plt.show()
```



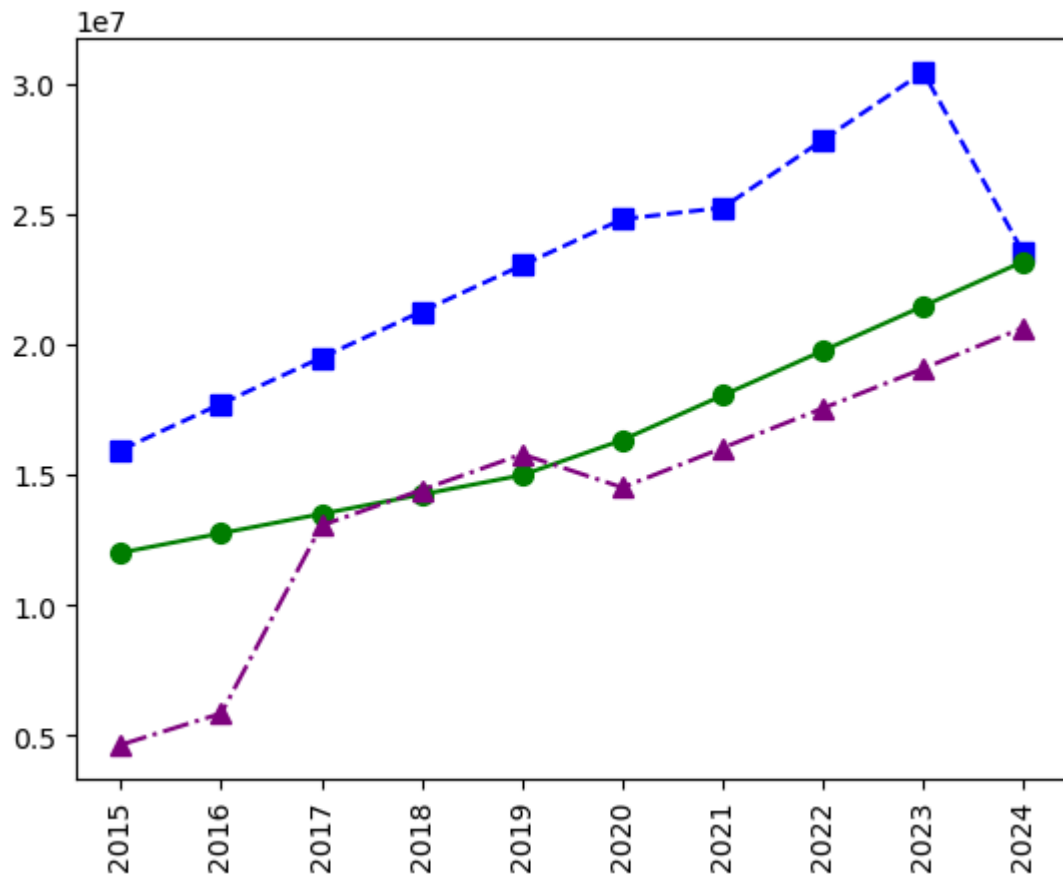
```
In [47]: plt.plot(Salary[1], c = 'blue', ls='--', marker= 's', ms=8, label=Players[1])  
plt.show()
```



```
In [53]: plt.plot(Salary[0], c = 'red', ls='--', marker= 's', ms=8, label=Players[0])
plt.plot(Salary[1], c = 'green', ls='--', marker= 'o', ms=8, label=Players[1])
plt.xticks(list(range(0,10)), Seasons, rotation='vertical') # rotation='vertical'
plt.show()
```

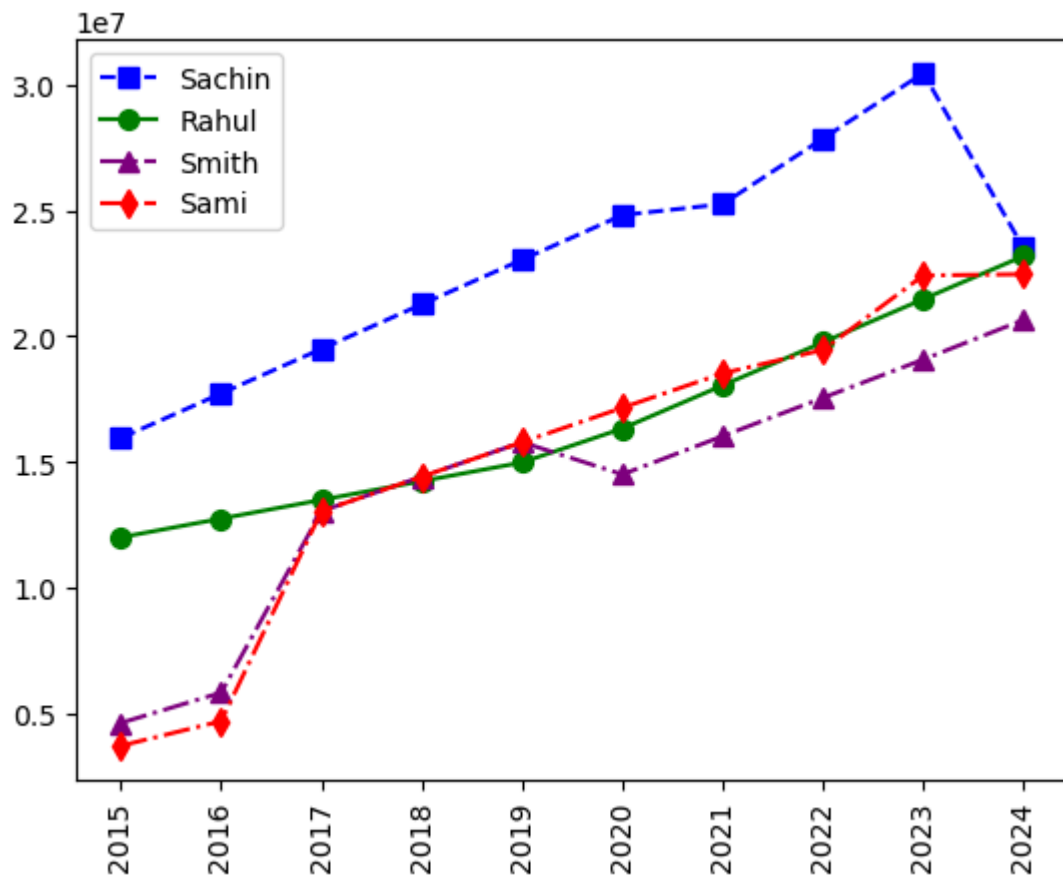


```
In [56]: plt.plot(Salary[0], c = 'blue', ls='--', marker= 's', ms=7, label=Players[0])
plt.plot(Salary[1], c = 'green', ls='-', marker= 'o', ms=7, label=Players[1])
plt.plot(Salary[2], c = 'purple', ls='-.', marker= '^', ms=7, label=Players[2])
plt.xticks(list(range(0,10)), Seasons, rotation='vertical') # rotation='vertical'
plt.show()
```



```
In [58]: plt.plot(Salary[0], c = 'blue', ls='--', marker= 's', ms=7, label=Players[0])
plt.plot(Salary[1], c = 'green', ls='-', marker= 'o', ms=7, label=Players[1])
plt.plot(Salary[2], c = 'purple', ls='-.', marker= '^', ms=7, label=Players[2])
plt.plot(Salary[3], c = 'red', ls='-.', marker= 'd', ms=7, label=Players[3])

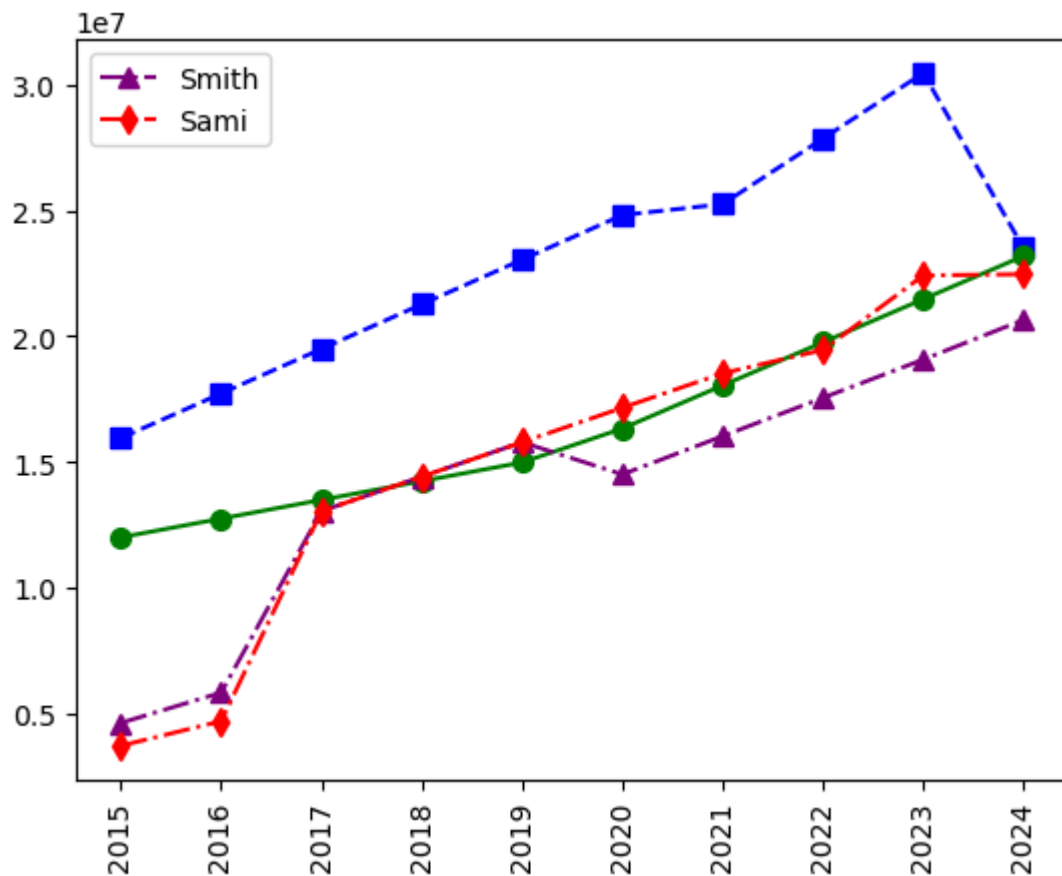
plt.legend()
plt.xticks(list(range(0,10)), Seasons, rotation='vertical') # rotation='vertical'
plt.show()
```



```
In [60]: plt.plot(Salary[0], c = 'blue', ls='--', marker= 's', ms=7)
plt.plot(Salary[1], c = 'green', ls='-', marker= 'o', ms=7)
plt.plot(Salary[2], c = 'purple', ls='-.', marker= '^', ms=7, label=Players[2])
plt.plot(Salary[3], c = 'red', ls='-.', marker= 'd', ms=7, label=Players[3])

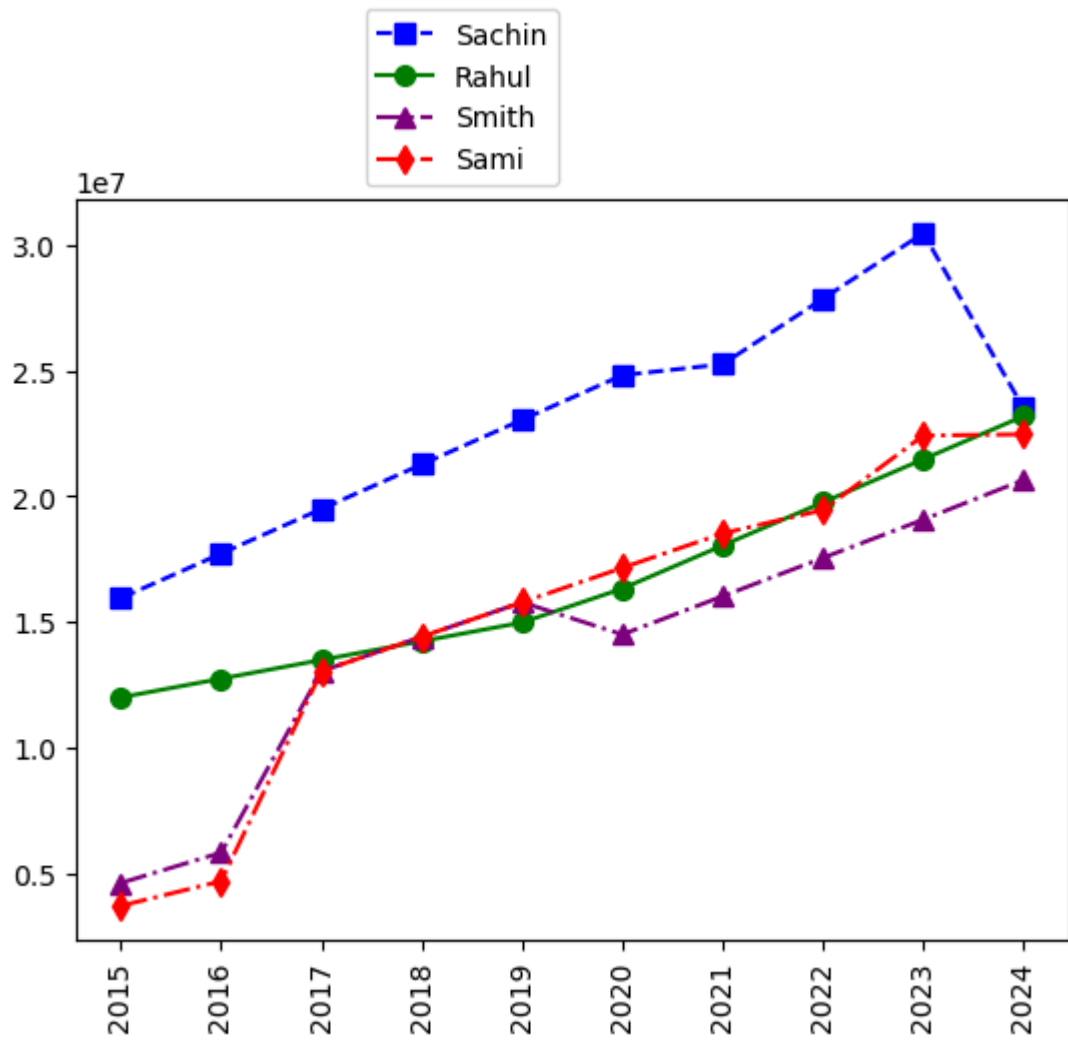
plt.legend()
plt.xticks(list(range(0,10)), Seasons, rotation='vertical') # rotation='vertical'
plt.show()
```





```
In [61]: plt.plot(Salary[0], c = 'blue', ls='--', marker= 's', ms=7, label=Players[0])
plt.plot(Salary[1], c = 'green', ls='-', marker= 'o', ms=7, label=Players[1])
plt.plot(Salary[2], c = 'purple', ls='-.', marker= '^', ms=7, label=Players[2])
plt.plot(Salary[3], c = 'red', ls='-.', marker= 'd', ms=7, label=Players[3])

plt.legend(loc='lower right' , bbox_to_anchor=(0.5,1)) # boundry box use shift
plt.xticks(list(range(0,10)), Seasons, rotation='vertical') # rotation='vertical'
plt.show()
```



In [ ]: