Data Visualization with Matplotlib - Exercises

จงทำตามคำสั่งต่อไปนี้ด้วย data ที่กำหนดให้ต่อไปนี้

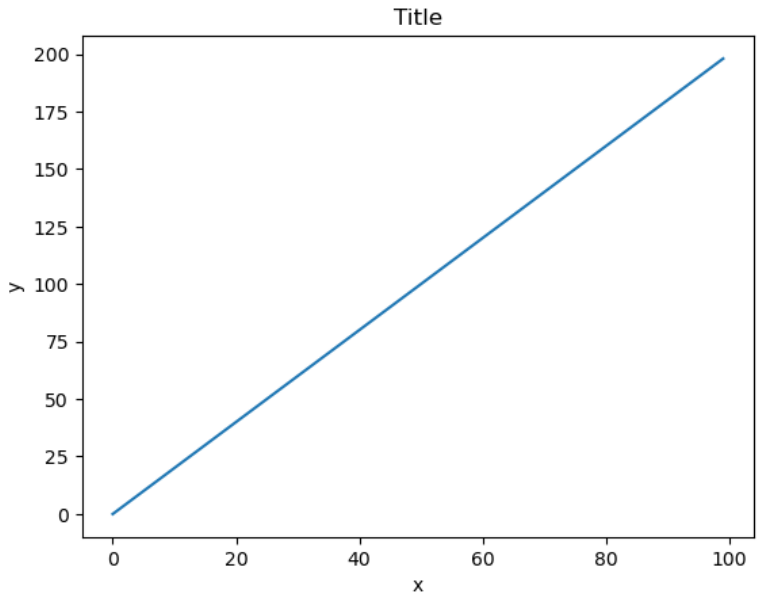
Data

In [1]:
```
1 import matplotlib.pyplot as plt
2 import numpy as np
3 import pandas as pd
4 x = np.arange(0,100)
5 y = x*2
6 z = x**2
7 df = pd.read_csv('Superstore.csv',encoding = 'iso-8859-1')
```

Exercise 1

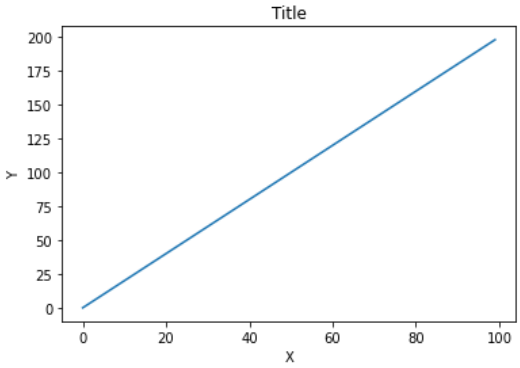
In [10]:
```
1 plt.plot(x, y)
2 plt.title("Title")
3 plt.xlabel("x")
4 plt.ylabel("y")
```

Out[10]: Text(0, 0.5, 'y')



In [2]:
```
1
```

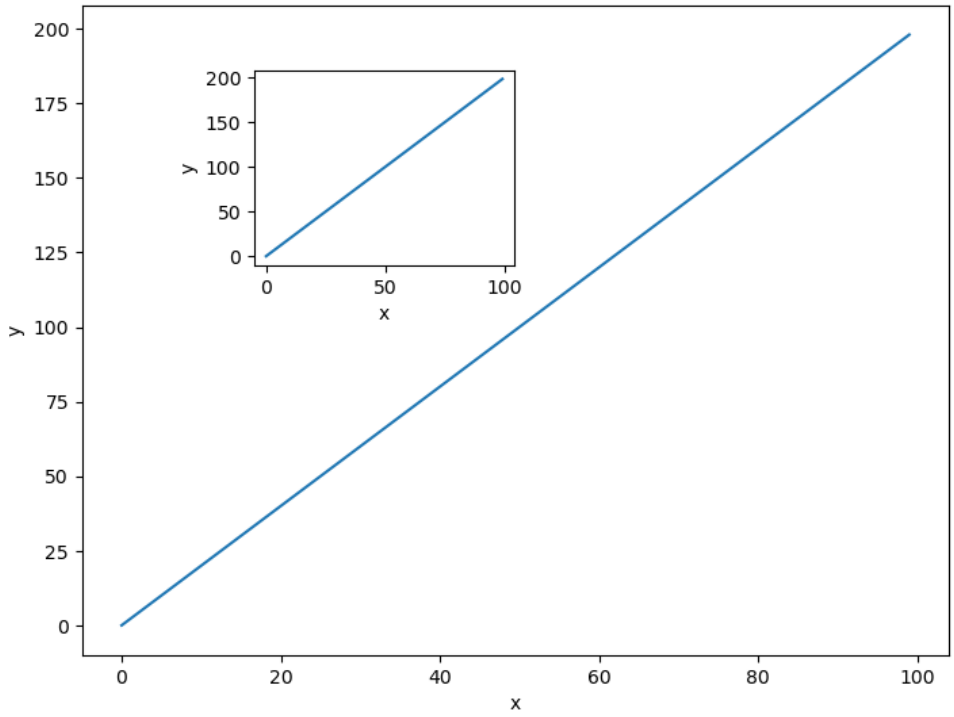
Out[2]: Text(0, 0.5, 'Y')



Exercise 2

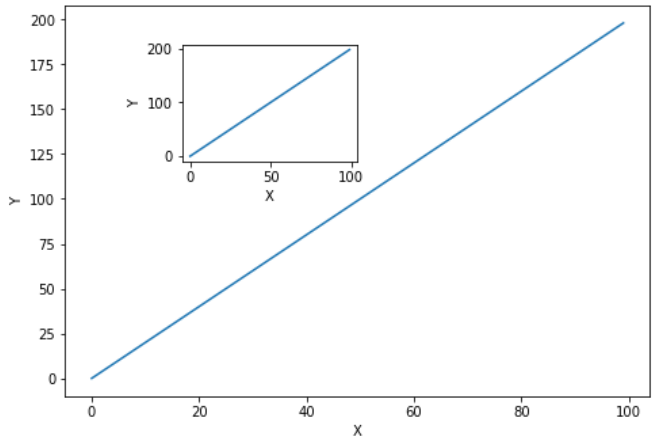
In [47]:
```
fig = plt.figure()
axes1 = fig.add_axes([0, 0, 1, 1])
axes1.set_xlabel("x")
axes1.set_ylabel("y")
axes2 = fig.add_axes([0.1986, 0.6, .3, .3])
axes2.set_xlabel("x")
axes2.set_ylabel("y")
axes1.plot(x, y)
axes2.plot(x, y)
```

Out[47]:
```
[<matplotlib.lines.Line2D at 0x2c220fa1a90>]
```



In [5]:
```
1
```

Out[5]:
```
Text(0, 0.5, 'Y')
```

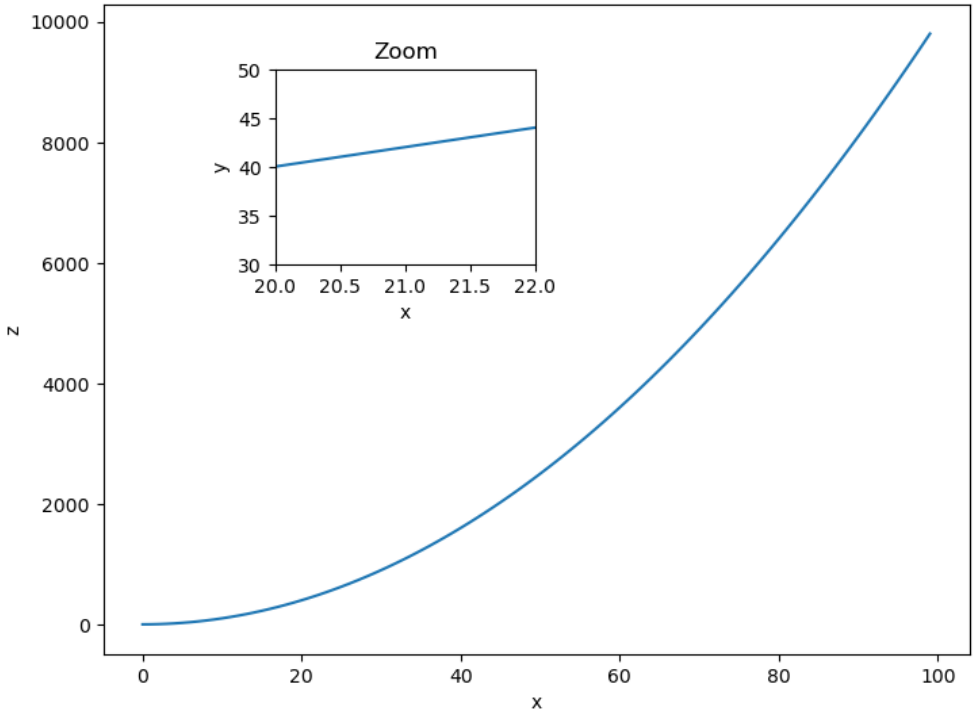


Exercise 3

ใช้ arrays x, y และ z เพื่อทำการ plot บนแกนที่สร้างจากข้อที่แล้ว (Notice อย่าลืมกำหนด x - limits และ y - limits)

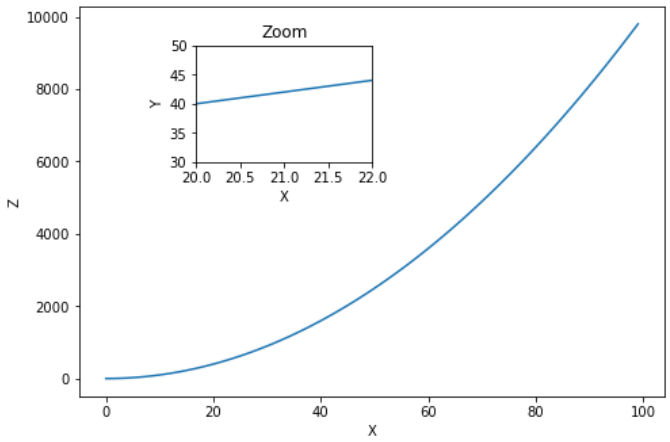
In [48]:
```
fig = plt.figure()
axes1 = fig.add_axes([0, 0, 1, 1])
axes1.set_xlabel("x")
axes1.set_ylabel("z")

axes2 = fig.add_axes([0.1986, 0.6, .3, .3])
axes2.set_title("Zoom")
axes2.set_xlabel("x")
axes2.set_ylabel("y")
axes1.plot(x, z)
axes2.plot(x, y)
axes2.set_ylim(30, 50)
axes2.set_xlim(20.0, 22.0)
```

Out[48]: (20.0, 22.0)



In [10]:
```
1
```

Out[10]: (30.0, 50.0)

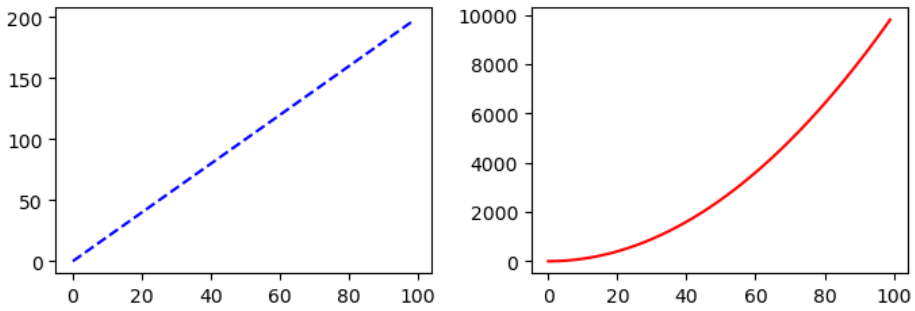


Exercise 4

จงใช้คำสั่ง plt.subplots(nrows=1, ncols=2)

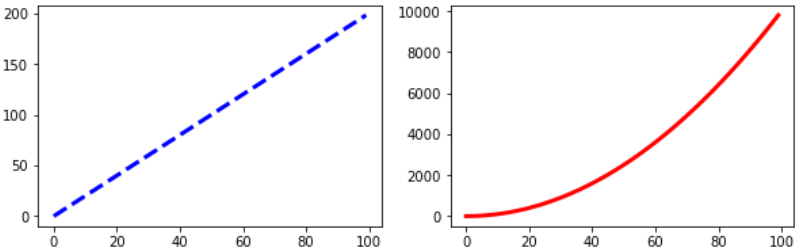
จากนั้นให้ทำการ plot (x,y) และ plot (x,z) บนแกน axes และให้ใช้งานคำสั่ง linewidth and style เพื่อตกแต่งเส้นของกราฟ

In [87]:
```
fig, axes = plt.subplots(nrows=1, ncols=2, figsize=(7, 2.5))
axes[0].plot(x, y, "--b")
axes[1].plot(x, z, "r")
fig.tight_layout()
```



In [13]:
```
1
```

Out[13]:
```
[<matplotlib.lines.Line2D at 0x1ea20aaa8e0>]
```



Exercise 5

In [49]:
```
df.head()
```

Out[49]:

	Order ID	Customer Name	Segment	Day	Month	Year	Ship Mode	City	State	Category	Sub-Category	Product Name	Sales	Quantity	Discount	Profi
0	CA-2016-152156	Claire Gute	Consumer	8	11	2016	Second Class	Henderson	Kentucky	Furniture	Bookcases	Bush Somerset Collection Bookcase	261.9600	2	0.00	41.913
1	CA-2016-152156	Claire Gute	Consumer	8	11	2016	Second Class	Henderson	Kentucky	Furniture	Chairs	Hon Deluxe Fabric Upholstered Stacking Chairs,...	731.9400	3	0.00	219.582
2	CA-2016-138688	Darrin Van Huff	Corporate	12	6	2016	Second Class	Los Angeles	California	Office Supplies	Labels	Self-Adhesive Address Labels for Typewriters b...	14.6200	2	0.00	6.871
3	US-2015-108966	Sean O'Donnell	Consumer	11	10	2015	Standard Class	Fort Lauderdale	Florida	Furniture	Tables	Bretford CR4500 Series Slim Rectangular Table	957.5775	5	0.45	-383.031
4	US-2015-108966	Sean O'Donnell	Consumer	11	10	2015	Standard Class	Fort Lauderdale	Florida	Office Supplies	Storage	Eldon Fold 'N Roll Cart System	22.3680	2	0.20	2.516

In [7]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9994 entries, 0 to 9993
Data columns (total 16 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   Order ID       9994 non-null   object
 1   Customer Name  9994 non-null   object
 2   Segment        9994 non-null   object
 3   Day            9994 non-null   int64
 4   Month          9994 non-null   int64
 5   Year           9994 non-null   int64
 6   Ship Mode      9994 non-null   object
 7   City           9994 non-null   object
 8   State          9994 non-null   object
 9   Category       9994 non-null   object
 10  Sub-Category   9994 non-null   object
 11  Product Name   9994 non-null   object
 12  Sales          9994 non-null   float64
 13  Quantity       9994 non-null   int64
 14  Discount       9994 non-null   float64
 15  Profit         9994 non-null   float64
dtypes: float64(3), int64(4), object(9)
memory usage: 1.2+ MB
```

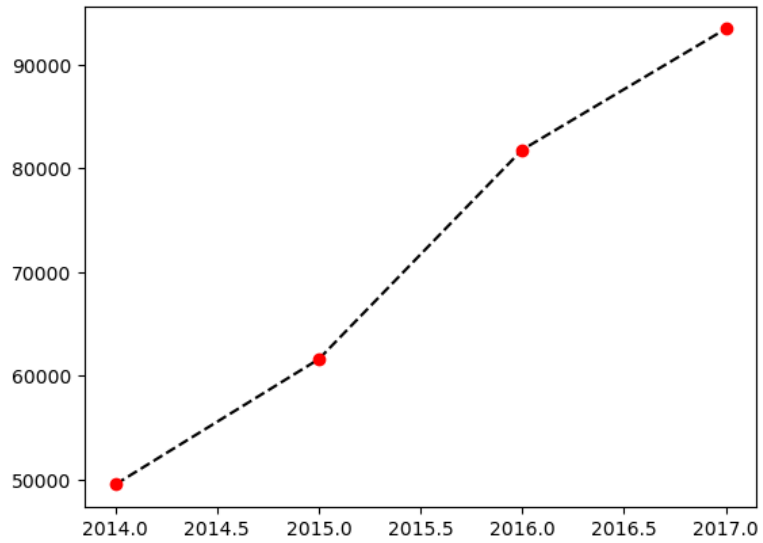
จงแสดงกราฟรายได้ของทุกปี

In [93]:
```
df1 = df.groupby('Year')['Profit'].sum()
a = df1.index
b = df1
```

```
Index([2014, 2015, 2016, 2017], dtype='int64', name='Year')
```

In [89]:
```
plt.plot(a, b, "o--k", mec = "r", mfc = "r" )
```

Out[89]:
```
[<matplotlib.lines.Line2D at 0x2c22642e1d0>]
```



โค้ดต่อไปนี้ใช้ในสองข้อสุดท้าย

In [90]:
```
df['Category'].unique()
```

Out[90]:
```
array(['Furniture', 'Office Supplies', 'Technology'], dtype=object)
```

In [91]:
```
df[ df['Category'] == 'Furniture' ].groupby('Year').sum()['Profit']
```

Out[91]:
```
Year
2014    5457.7255
2015    3015.2029
2016    6959.9531
2017    3018.3913
Name: Profit, dtype: float64
```

Dictionary of Category

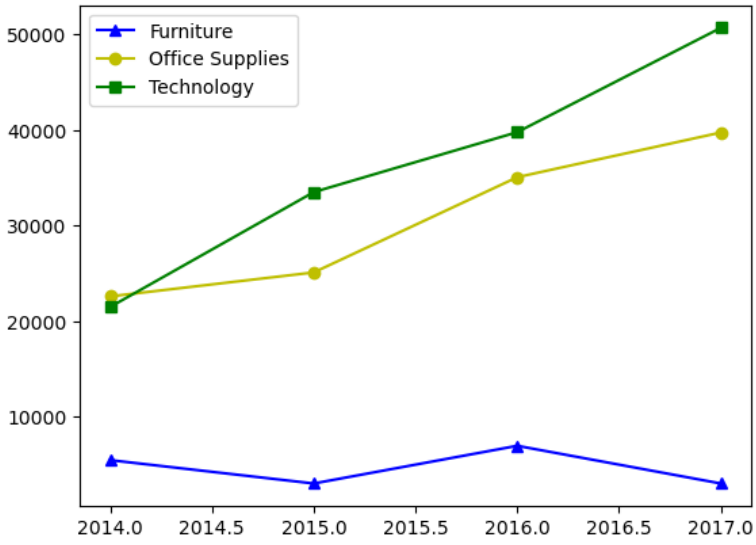
In [92]:
```
arr_df = {}
for i in range(0,df['Category'].nunique()) :
    arr_df[df['Category'].unique()[i]] = df[ df['Category'] == df['Category'].unique()[i] ].groupby('Year').sum()['Profit'
```

จงแสดงกราฟรายได้ของแต่ละ Category ในแต่ละปีในกราฟเดียว

In [103]:
```
a = arr_df["Furniture"].index
b = arr_df["Furniture"]
c = arr_df["Office Supplies"].index
d = arr_df["Office Supplies"]
e = arr_df["Technology"].index
f = arr_df["Technology"]
```

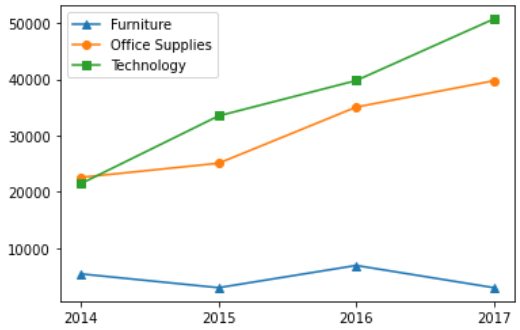
In [112]:
```
plt.plot(a, b, "^-b", label = "Furniture")
plt.plot(c, d, "o-y", label = "Office Supplies")
plt.plot(e, f, "s-g", label = "Technology")
plt.legend(loc = "upper left")
```

Out[112]: <matplotlib.legend.Legend at 0x2c229b4ea50>



In [90]:
```

```

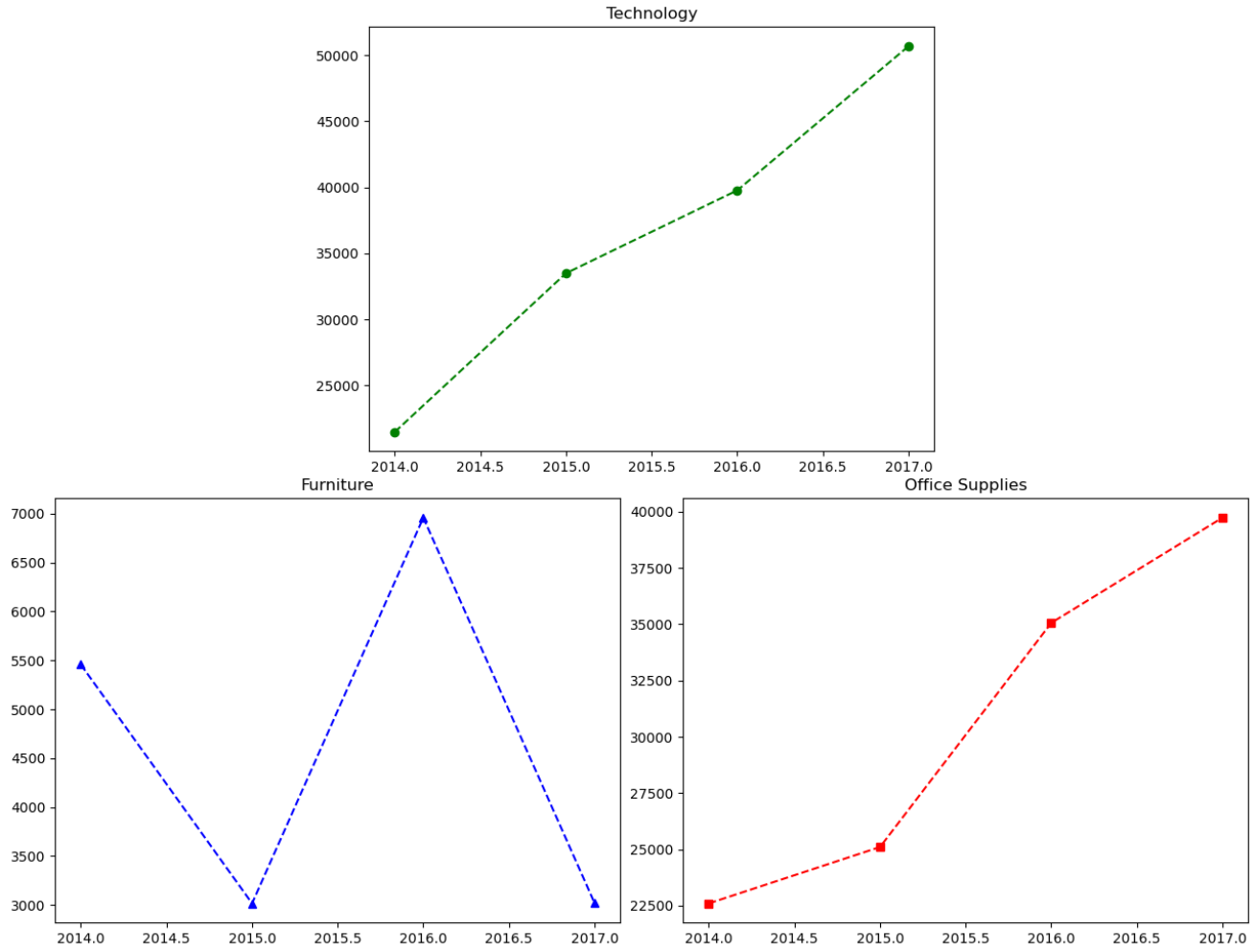
Out[90]: <matplotlib.legend.Legend at 0x1ea23a16d30>



จงแสดงกราฟรายได้ของแต่ละ Category ในแต่ละปี แบบแยกกราฟ

In [123]:
```
fig = plt.figure()
axes1 = fig.add_axes([0, 0, .9, .9])
axes1.set_title("Furniture")
axes1.plot(a, b, "^--b")

axes2 = fig.add_axes([1, 0, .9, .9])
axes2.set_title("Office Supplies")
axes2.plot(c, d, "s--r")

axes3 = fig.add_axes([.5, 1, .9, .9])
axes3.set_title("Technology")
axes3.plot(e, f, "o--g")
```

Out[123]: [<matplotlib.lines.Line2D at 0x2c22cb5bd50>]



In [ ]:
```
1
```