In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [7]:
```
mart=pd.read_csv(r'C:\Users\lenovo\Downloads\train.csv').sample(50)
mart.head()
```

Out[7]:

	Item_ID	Item_W	Item_Type	Item_MRP	Outlet_ID	Outlet_Year	Outlet_Size	Outlet_Locatio
34876	FDR31	13.161458	Baking Goods	68.118613	OUT013	1999	Small	
24773	DRF23	20.857822	Canned	143.331651	OUT046	2004	Small	
71832	FDD05	11.158053	Frozen Foods	263.219504	OUT013	2004	Small	
67527	FDZ52	16.507745	Snack Foods	132.326101	OUT018	1997	Small	
24981	FDY47	20.444305	Soft Drinks	131.451701	OUT046	2009	Medium	

In [8]:
```
mart.shape
```

Out[8]:
```
(50, 9)
```

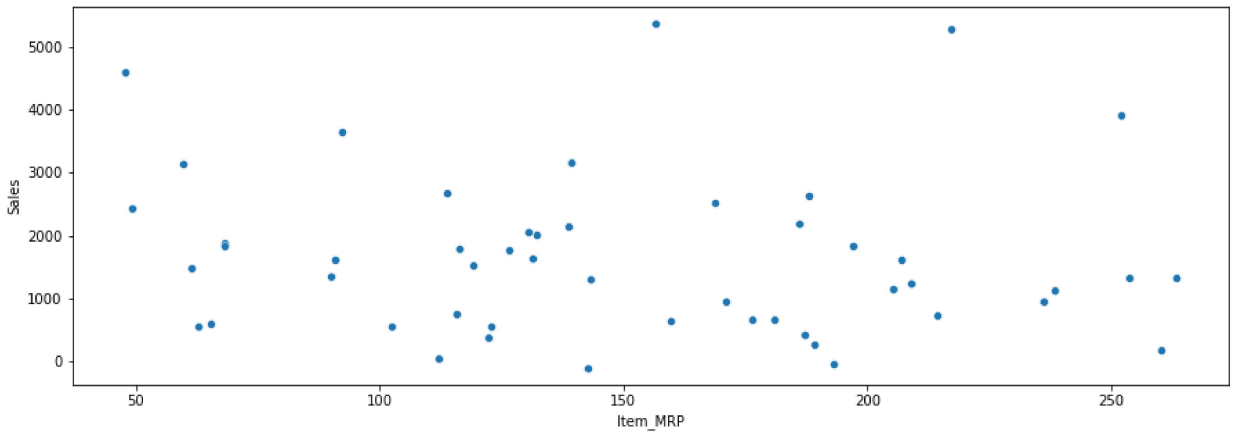
SCATTER PLOT

A diagram which shows the relationship between two variables by plotting the points/dots

1. Create a basic scatter plot

In [25]:
```
plt.rcParams['figure.figsize']=(15,5)
sns.scatterplot(data=mart,x='Item_MRP',y='Sales')
```

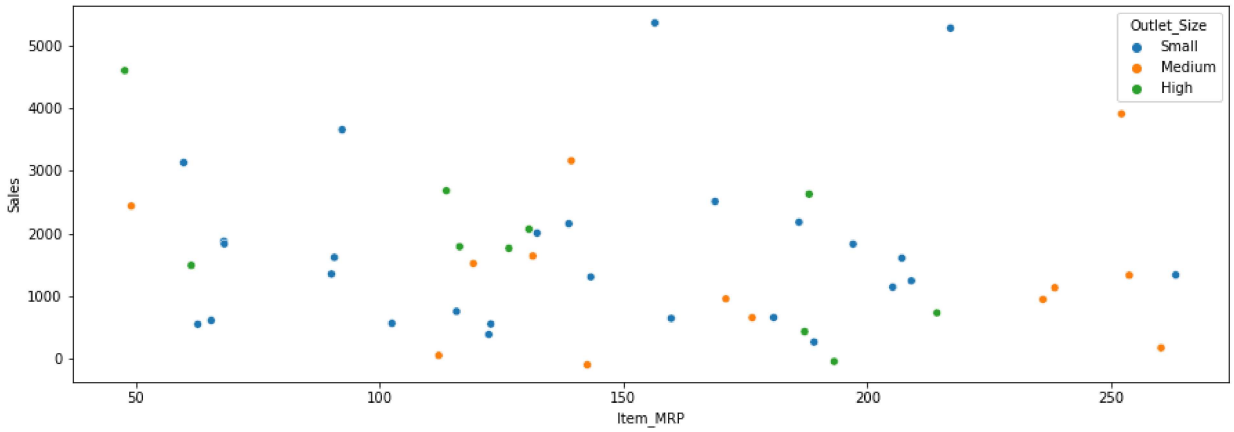
Out[25]:
```
<AxesSubplot:xlabel='Item_MRP', ylabel='Sales'>
```



2. Grouping basis on a categorical variable using hue

In [26]:
```
sns.scatterplot(data=mart,x='Item_MRP',y='Sales',hue='Outlet_Size')
```

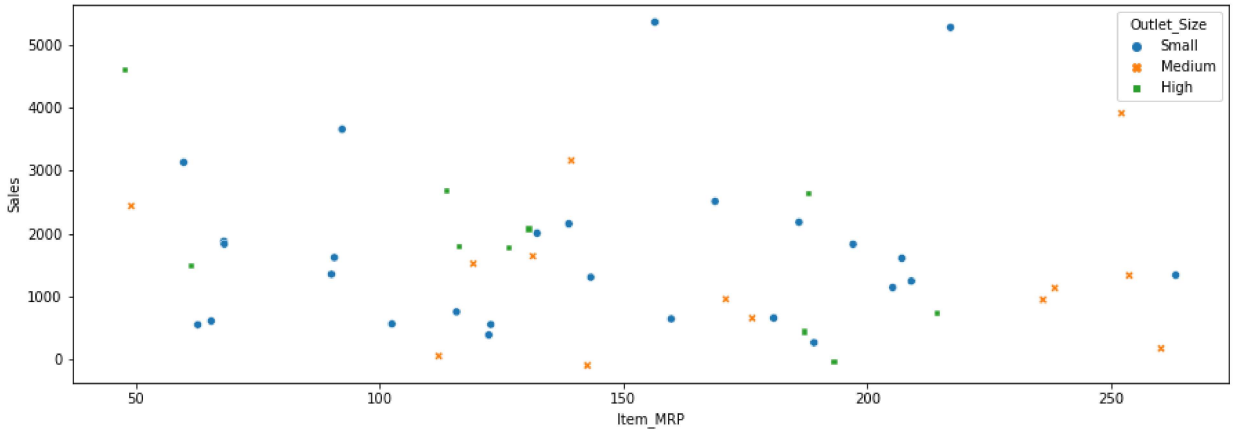
Out[26]:
```
<AxesSubplot:xlabel='Item_MRP', ylabel='Sales'>
```



3. Grouping basis on a categorical variable using STYLE and Styling markers as well

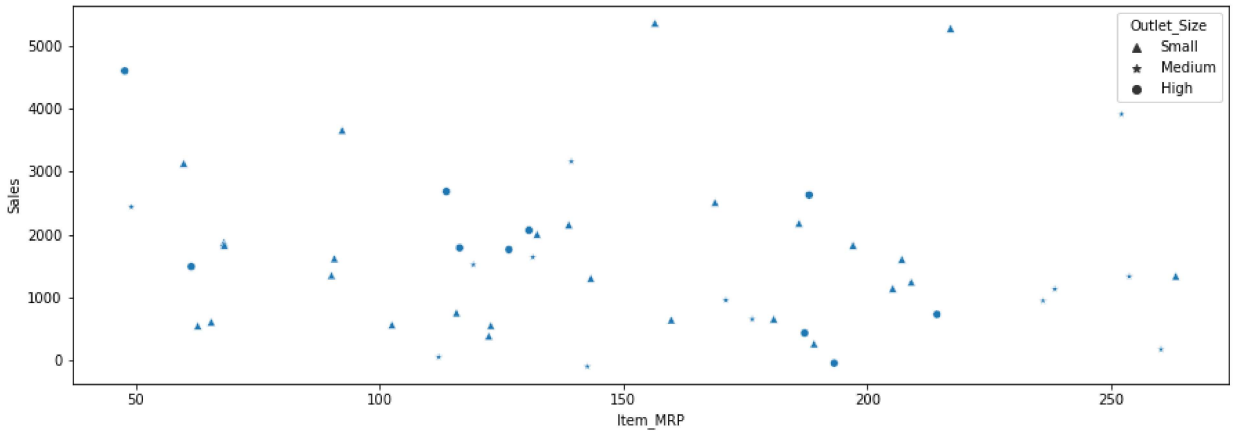
In [23]:
```
sns.scatterplot(data=mart,x='Item_MRP',y='Sales',hue='Outlet_Size',style='Outlet_Size'
```

Out[23]:
```
<AxesSubplot:xlabel='Item_MRP', ylabel='Sales'>
```



In [24]:
```
sns.scatterplot(data=mart,x='Item_MRP',y='Sales',style='Outlet_Size',markers={"Small"
```

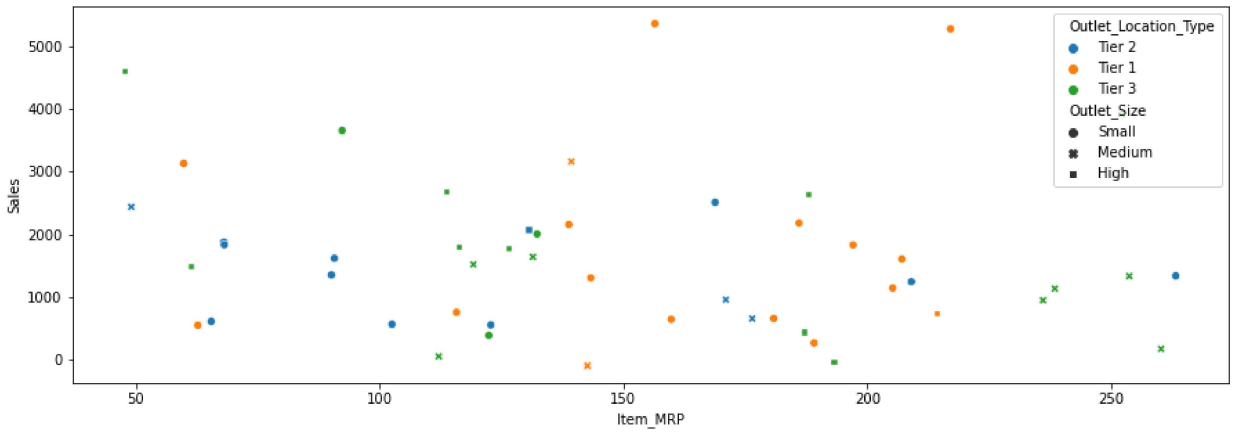
Out[24]:
```
<AxesSubplot:xlabel='Item_MRP', ylabel='Sales'>
```



4. Grouping basis on a categorical variable using HUE & STYLE together

In [22]:
```
sns.scatterplot(data=mart,x='Item_MRP',y='Sales',hue='Outlet_Location_Type',style='Out
```

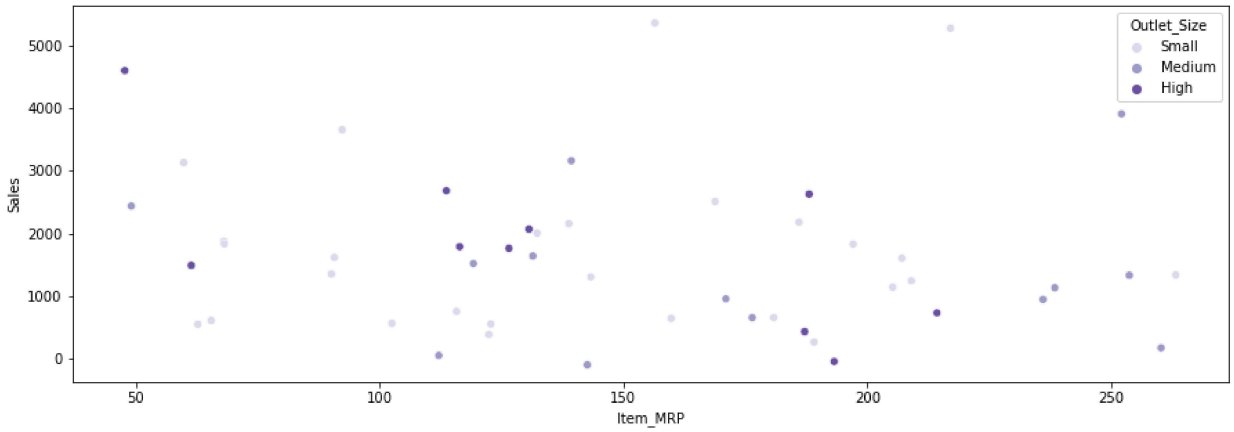
Out[22]:
```
<AxesSubplot:xlabel='Item_MRP', ylabel='Sales'>
```



5. Grouping basis on numeric variable using HUE and use PALETTE

In [27]:
```
sns.scatterplot(data=mart,x='Item_MRP',y='Sales',hue='Outlet_Size',palette='Purples')
```

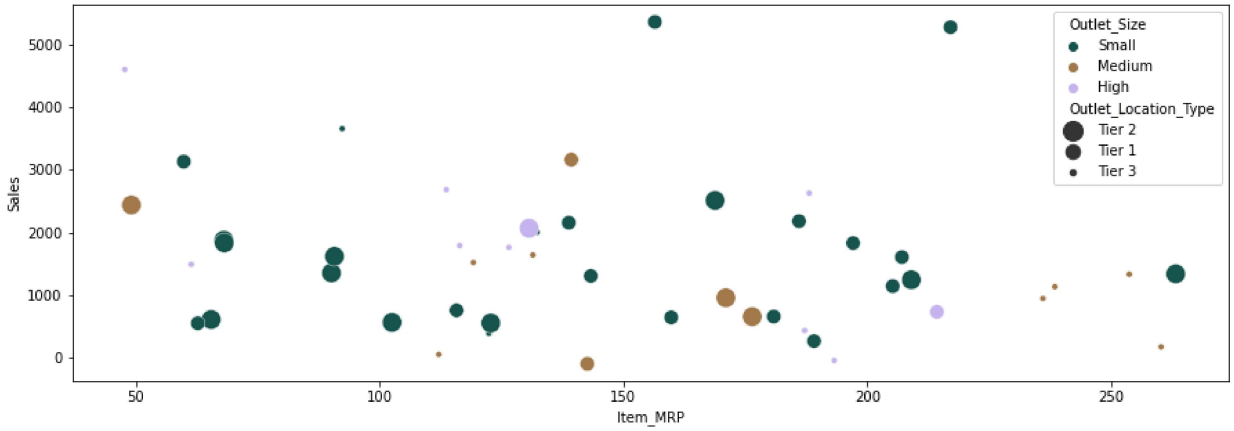
Out[27]:
```
<AxesSubplot:xlabel='Item_MRP', ylabel='Sales'>
```



6. Grouping basis on a numeric variable using HUE & SIZE

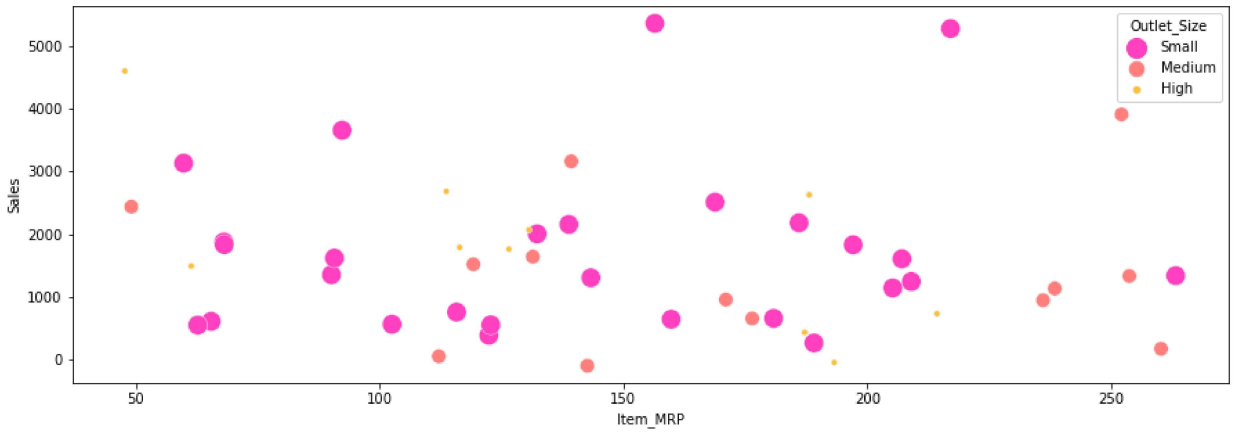
In [33]:
```
sns.scatterplot(data=mart,x='Item_MRP',y='Sales'
                ,hue='Outlet_Size'
                ,size='Outlet_Location_Type'
                ,sizes=(20,200)
                ,legend='full'
                ,palette='cubehelix'
                 )
```

Out[33]:
```
<AxesSubplot:xlabel='Item_MRP', ylabel='Sales'>
```



In [37]:
```
# it's better to use hue and size with same feature
sns.scatterplot(data=mart,x='Item_MRP',y='Sales'
                ,hue='Outlet_Size'
                ,size='Outlet_Size'
                ,sizes=(20,200)
                ,legend='full'
                ,palette='spring'
                 )
```

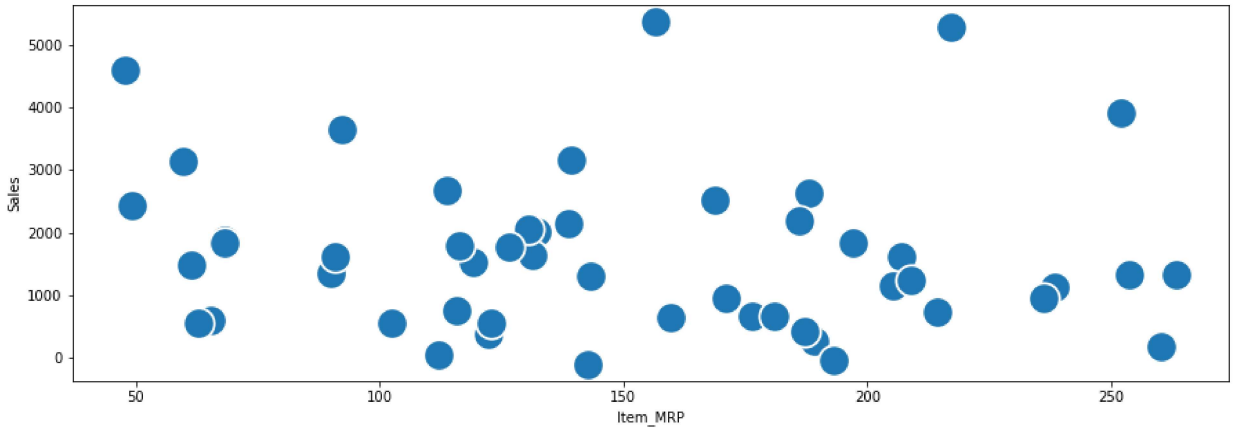
Out[37]:
```
<AxesSubplot:xlabel='Item_MRP', ylabel='Sales'>
```



7. Change the Marker Size using S

In [43]:
```
sns.scatterplot(data=mart,x='Item_MRP',y='Sales',s=500)
```

Out[43]:
```
<AxesSubplot:xlabel='Item_MRP', ylabel='Sales'>
```



LINE PLOT

Shoes the relation between two variables || Majorly used in time series analysis

Showcases how the value of variable changes over the time

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

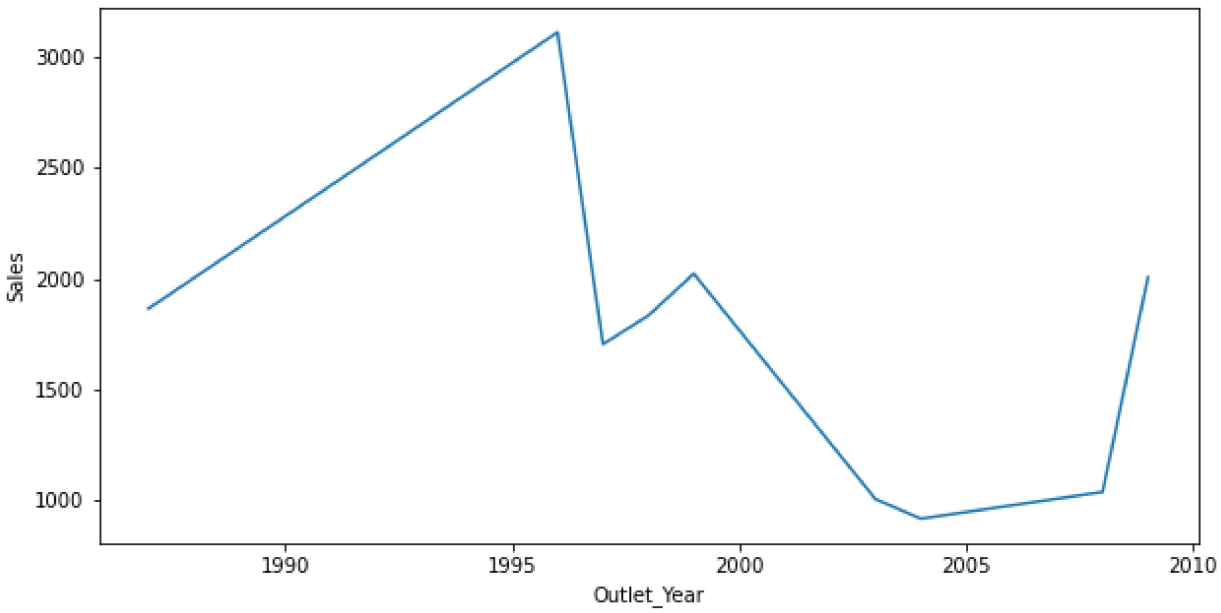
Out[49]:

	Item_ID	Item_W	Item_Type	Item_MRP	Outlet_ID	Outlet_Year	Outlet_Size	Outlet_Locatio
34876	FDR31	13.161458	Baking Goods	68.118613	OUT013	1999	Small	
24773	DRF23	20.857822	Canned	143.331651	OUT046	2004	Small	
71832	FDD05	11.158053	Frozen Foods	263.219504	OUT013	2004	Small	
67527	FDZ52	16.507745	Snack Foods	132.326101	OUT018	1997	Small	
24981	FDY47	20.444305	Soft Drinks	131.451701	OUT046	2009	Medium	

1. Create BASIC Line PLot

In [55]:
```
plt.figure(figsize=(10,5))
sns.lineplot(data=mart,x='Outlet_Year',y='Sales',ci=None)
```

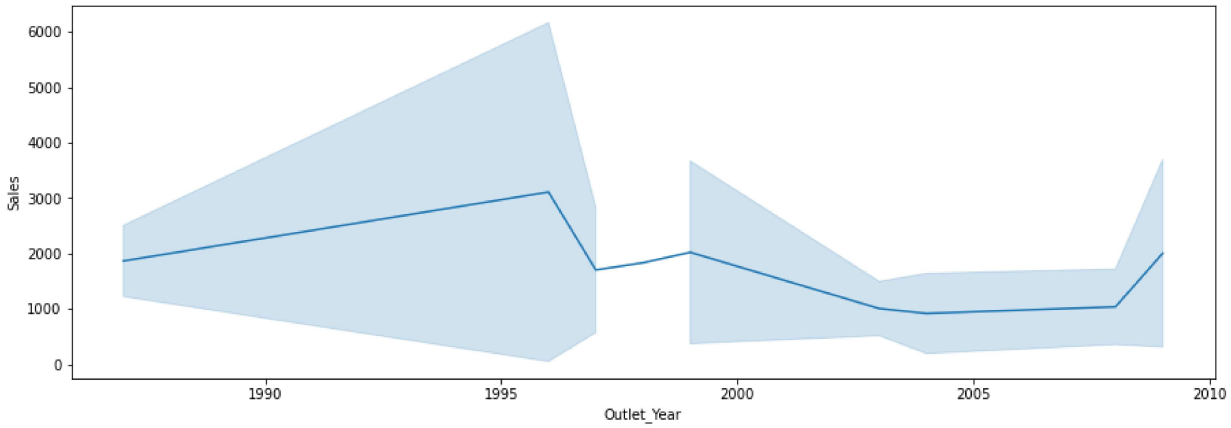
Out[55]:
```
<AxesSubplot:xlabel='Outlet_Year', ylabel='Sales'>
```



2. Test with different confidence interval and with different number of Bootstrapping

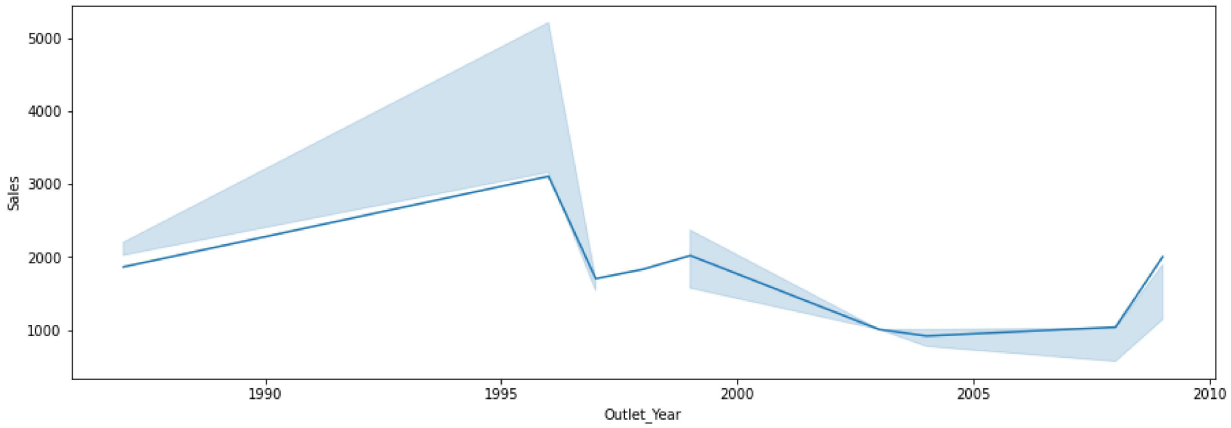
In [61]:
```
sns.lineplot(data=mart,x='Outlet_Year',y='Sales',ci="sd") #sd --> standard deviation
```

Out[61]: <AxesSubplot:xlabel='Outlet_Year', ylabel='Sales'>



In [66]:
```
sns.lineplot(data=mart,x='Outlet_Year',y='Sales',n_boot=2) #n_boot default is 1000
```

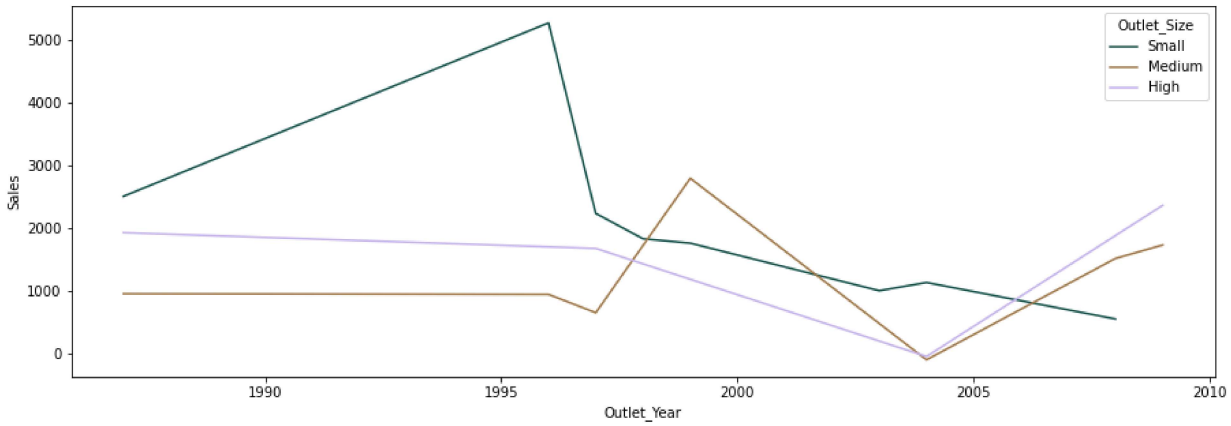
Out[66]: <AxesSubplot:xlabel='Outlet_Year', ylabel='Sales'>



3. Grouping using HUE and use different Palettes

In [72]:
```
sns.lineplot(data=mart,x='Outlet_Year',y='Sales',ci=None,hue='Outlet_Size',palette='cu
```

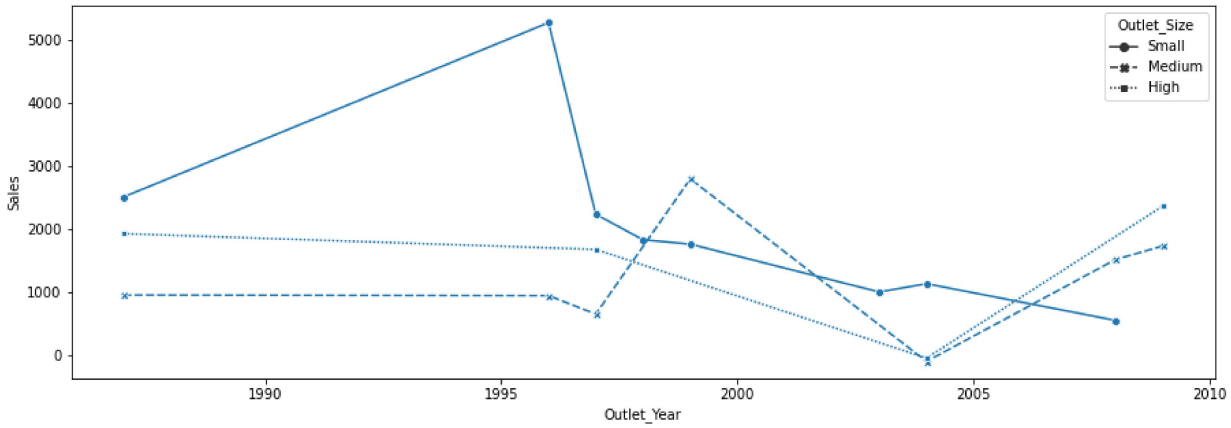
Out[72]: <AxesSubplot:xlabel='Outlet_Year', ylabel='Sales'>



4. Grouping using style,use different Markers and Dashes

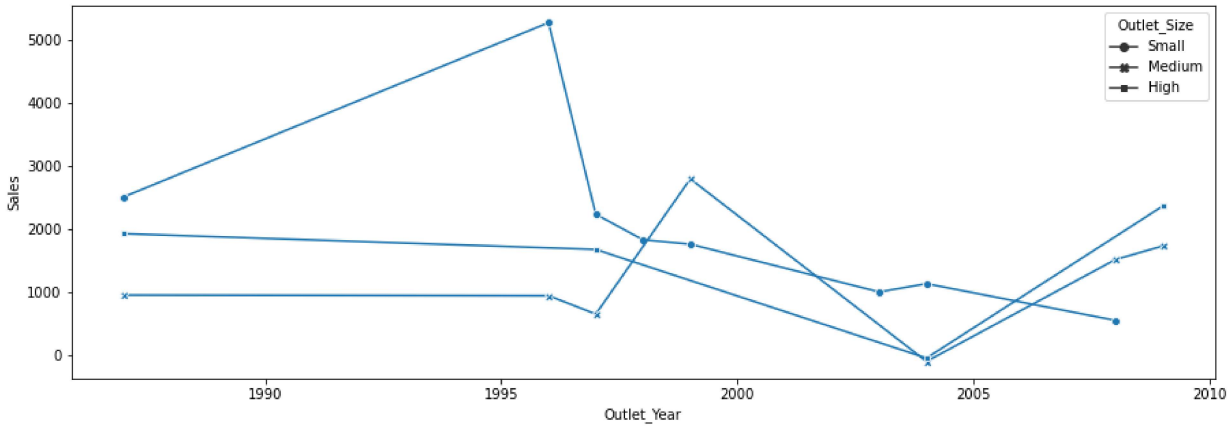
In [76]:
```
sns.lineplot(data=mart,x='Outlet_Year',y='Sales',ci=None,style='Outlet_Size',markers=T
```

Out[76]: <AxesSubplot:xlabel='Outlet_Year', ylabel='Sales'>



In [77]:
```
sns.lineplot(data=mart,x='Outlet_Year',y='Sales',ci=None,style='Outlet_Size',markers=T
```

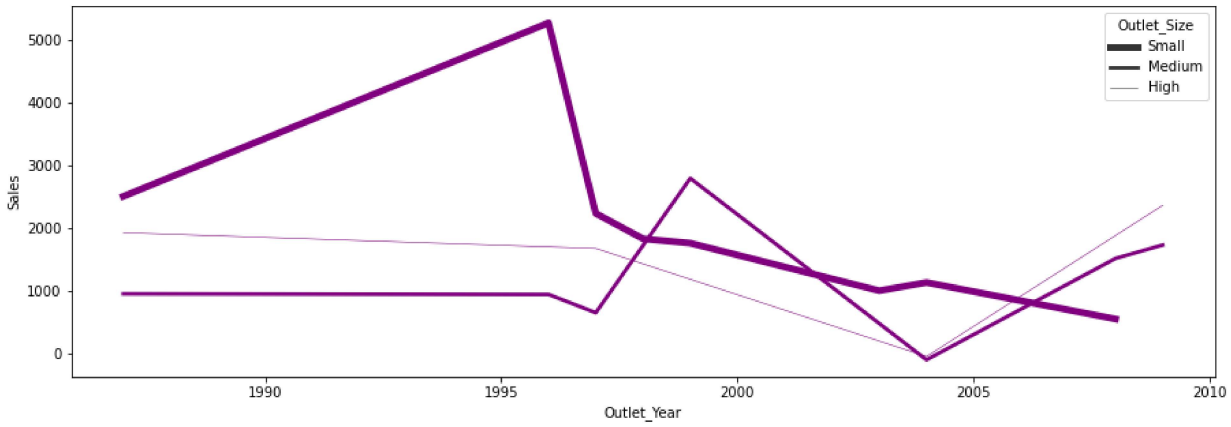
Out[77]: <AxesSubplot:xlabel='Outlet_Year', ylabel='Sales'>



5. Grouping using size

In [82]:
```
sns.lineplot(data=mart,x='Outlet_Year',y='Sales',ci=None,size='Outlet_Size',color='Pur
```

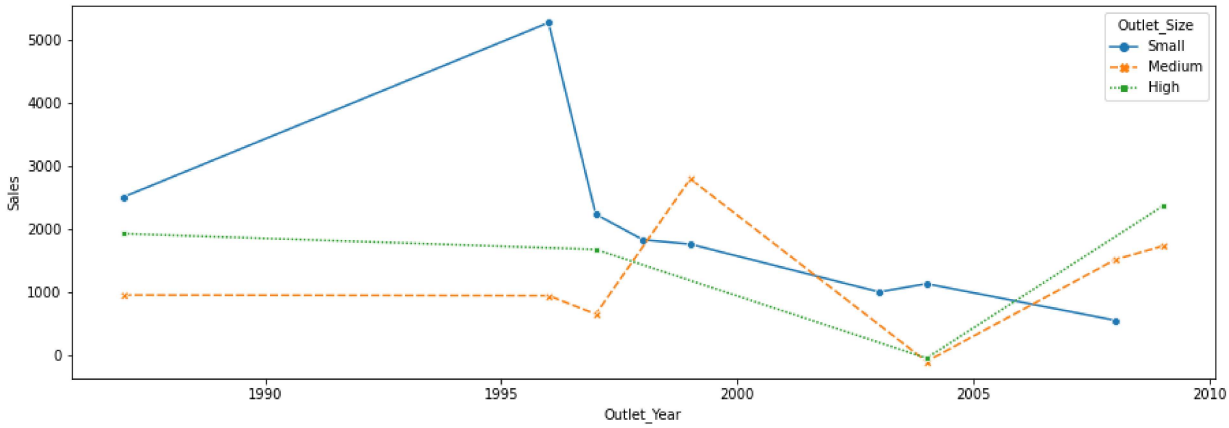
Out[82]: <AxesSubplot:xlabel='Outlet_Year', ylabel='Sales'>



6. Use different semantic styling parameters on same different variables

In [85]:
```
sns.lineplot(data=mart,x='Outlet_Year',y='Sales',ci=None,hue='Outlet_Size',style='Outl
```

Out[85]: <AxesSubplot:xlabel='Outlet_Year', ylabel='Sales'>



REL PLOT

In [86]:
```
mart.head()
```

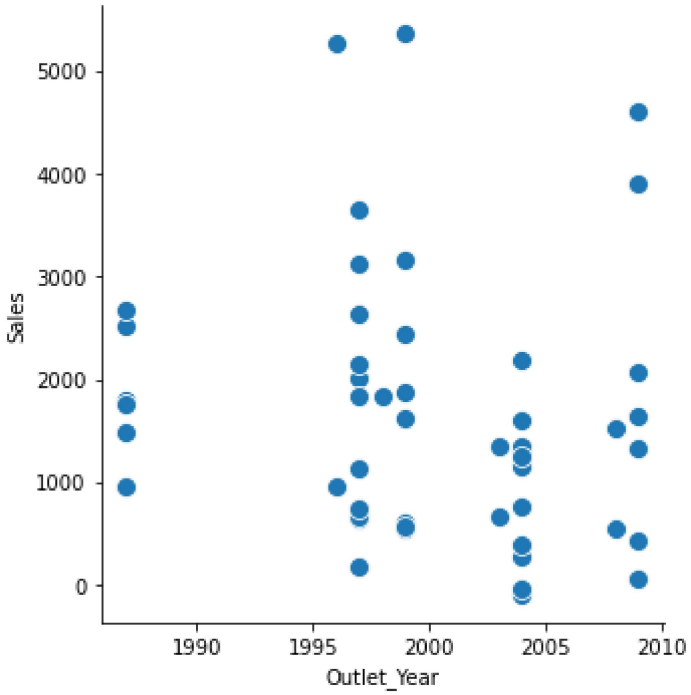
Out[86]:

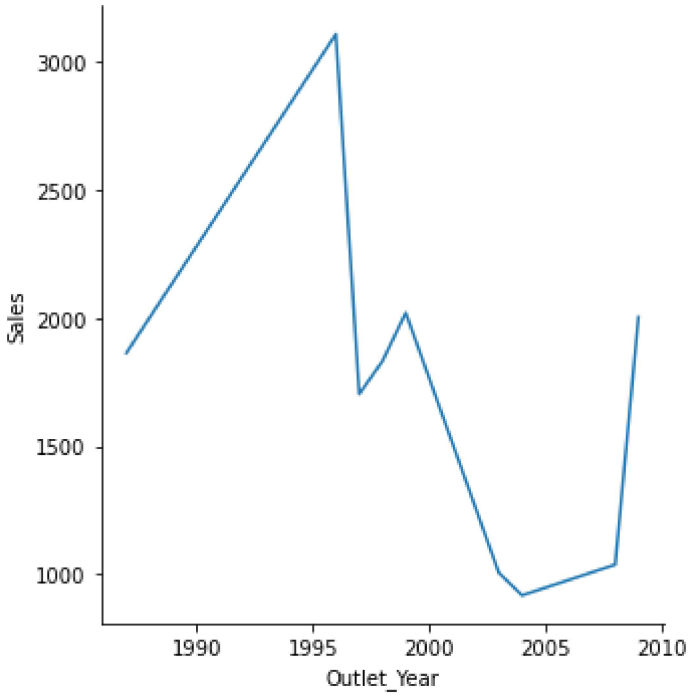
	Item_ID	Item_W	Item_Type	Item_MRP	Outlet_ID	Outlet_Year	Outlet_Size	Outlet_Locatio
34876	FDR31	13.161458	Baking Goods	68.118613	OUT013	1999	Small	
24773	DRF23	20.857822	Canned	143.331651	OUT046	2004	Small	
71832	FDD05	11.158053	Frozen Foods	263.219504	OUT013	2004	Small	
67527	FDZ52	16.507745	Snack Foods	132.326101	OUT018	1997	Small	
24981	FDY47	20.444305	Soft Drinks	131.451701	OUT046	2009	Medium	

1. First Rel Plot

In [90]:
```
sns.relplot(x='Outlet_Year',y='Sales',data=mart,s=100)
sns.relplot(x='Outlet_Year',y='Sales',data=mart,kind='line',ci=None)
```

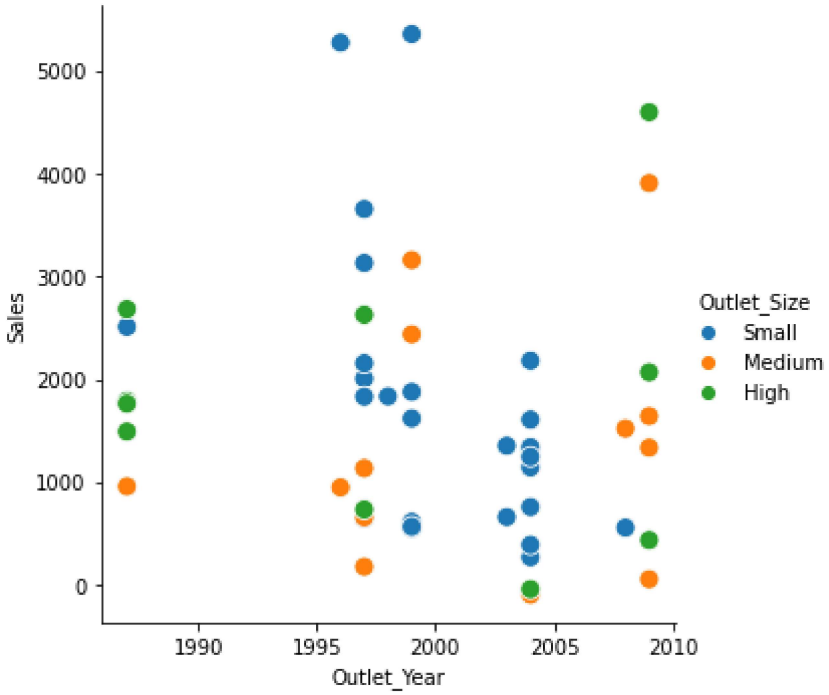
Out[90]: <seaborn.axisgrid.FacetGrid at 0x2766196eb80>





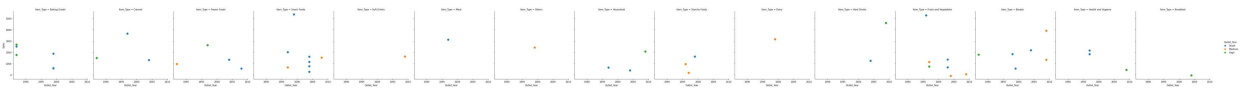
In [92]:
```
sns.relplot(x='Outlet_Year',y='Sales',data=mart,s=100,hue='Outlet_Size')
```

Out[92]:
```
<seaborn.axisgrid.FacetGrid at 0x2766199d970>
```



In [93]:
```
sns.relplot(x='Outlet_Year',y='Sales',data=mart,s=100,hue='Outlet_Size',col='Item_Type
```

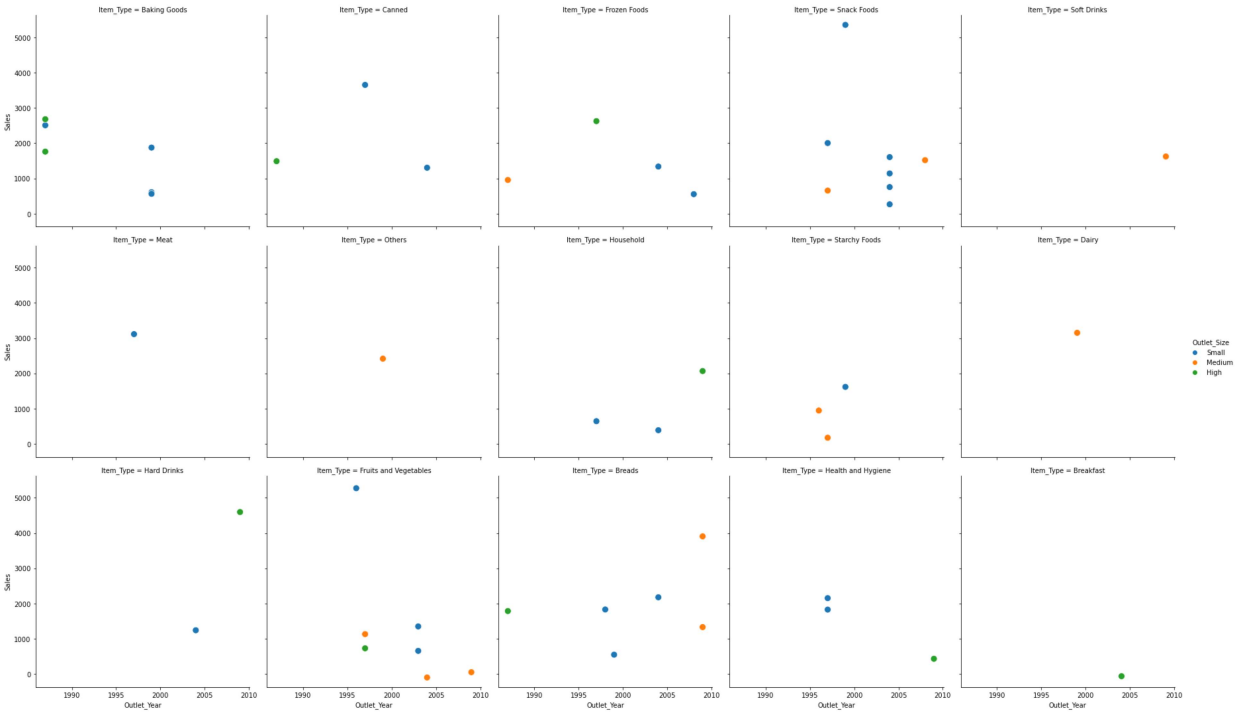
Out[93]:
```
<seaborn.axisgrid.FacetGrid at 0x27661b0b8b0>
```



since there are 15 unique values it showing in one row. So we use col_wrap to make it more organised

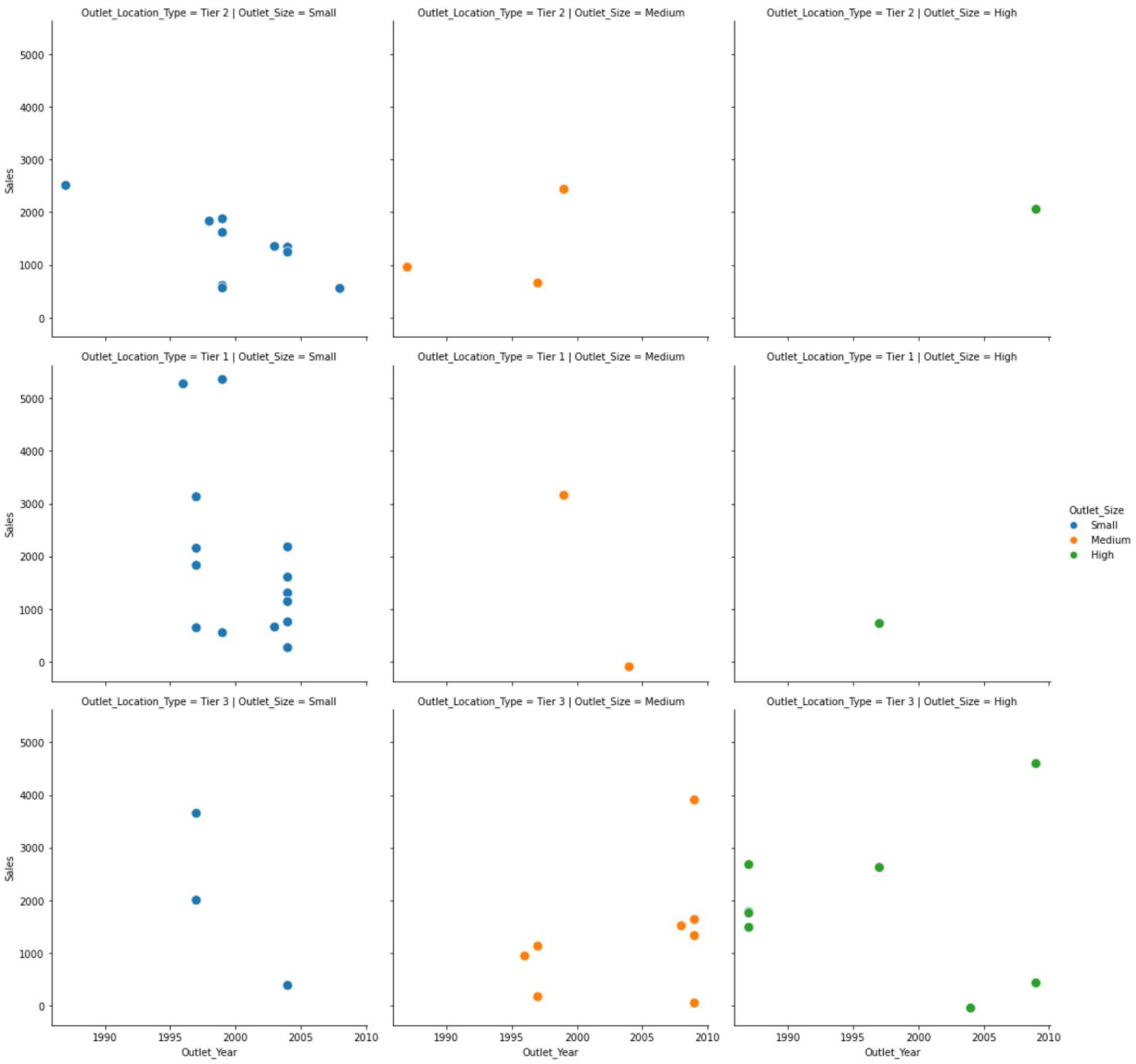
In [97]:
```
sns.relplot(x='Outlet_Year',y='Sales',data=mart,s=100,hue='Outlet_Size',col='Item_Type
```

Out[97]:
```
<seaborn.axisgrid.FacetGrid at 0x27665127850>
```



In [98]:
```
sns.relplot(x='Outlet_Year',y='Sales',data=mart,s=100,hue='Outlet_Size',col='Outlet_Si
```

Out[98]:
```
<seaborn.axisgrid.FacetGrid at 0x27665261850>
```



In [ ]:
```
```