In [220…
```
import pandas as pd
import numpy as np
```

In [221…
```
ign_master = pd.read_csv(r"D:\PG-DAI\MachineLearning\Task Afternoon 9 Dec\ign_n.csv")

ign_master.tail(10)
```

Out[221…

	Unnamed: 0	score_phrase	title	url	platform	score	genre	editors_choice	release_year	release_month	release_day
18615	18615	Amazing	Zero Time Dilemma	/games/zero-time-dilemma/vita-20039757	PlayStation Vita	9.2	Adventure	Y	2016	6	29
18616	18616	Good	Batman: The Telltale Series -- Episode 1: Real...	/games/batman-the-telltale-series-episode-1-re...	PC	7.5	Adventure	N	2016	8	2
18617	18617	Great	Abzu	/games/abzu/ps4-20019841	PlayStation 4	8.4	Adventure	N	2016	8	2
18618	18618	Amazing	Starbound	/games/starbound-2016/pc-128879	PC	9.1	Action	Y	2016	7	28
18619	18619	Good	Human Fall Flat	/games/human-fall-flat/pc-20051928	PC	7.9	Puzzle, Action	N	2016	7	28
18620	18620	Good	Tokyo Mirage Sessions #FE	/games/fire-emblem-x-shin-megami-tensei/wii-u-...	Wii U	7.6	RPG	N	2016	6	29
18621	18621	Amazing	LEGO Star Wars: The Force Awakens	/games/lego-star-wars-the-force-awakens/ps4-20...	PlayStation 4	9.0	Action, Adventure	Y	2016	6	29
18622	18622	Mediocre	Star Ocean: Integrity and Faithlessness	/games/star-ocean-5/ps4-20035681	PlayStation 4	5.8	RPG	N	2016	6	28

	Unnamed: 0	score_phrase	title	url	platform	score	genre	editors_choice	release_year	release_month	release_day
18623	18623	Masterpiece	Inside	/games/inside-playdead/xbox-one-121435	Xbox One	10.0	Adventure	Y	2016	6	28
18624	18624	Masterpiece	Inside	/games/inside-playdead/pc-20055740	PC	10.0	Adventure	Y	2016	6	28

In [222...
```
ign_master.describe()
```

Out[222...

	Unnamed: 0	score	release_year	release_month	release_day
count	18625.000000	18625.000000	18625.000000	18625.00000	18625.000000
mean	9312.000000	6.950459	2006.515329	7.13847	15.603866
std	5376.718717	1.711736	4.587529	3.47671	8.690128
min	0.000000	0.500000	1970.000000	1.00000	1.000000
25%	4656.000000	6.000000	2003.000000	4.00000	8.000000
50%	9312.000000	7.300000	2007.000000	8.00000	16.000000
75%	13968.000000	8.200000	2010.000000	10.00000	23.000000
max	18624.000000	10.000000	2016.000000	12.00000	31.000000

In [223...
```
ps3 = ign_master[ign_master['platform'] == 'PlayStation 3']
```

In [224...
```
ign_master.value_counts('score_phrase')['Great']
```

Out[224... 4773

In [225...
```
platforms = ['PlayStation 3','PlayStation 4','Xbox 360','Xbox One','PC']
# ign_master[]

phrase = ['Amazing','Great','Okay']
```

In [226…
```
X_One= pd.DataFrame()
temp = []
final = []
for j in platforms:
    print(j)

    j = ign_master[ign_master['platform'] == j]
    for i in phrase:
        print(i,j.value_counts('score_phrase')[i] )
        temp.append(j.value_counts('score_phrase')[i])

    final.append(temp)
    temp = []
```

```
PlayStation 3
Amazing 181
Great 379
Okay 196
PlayStation 4
Amazing 55
Great 83
Okay 36
Xbox 360
Amazing 196
Great 445
Okay 256
Xbox One
Amazing 35
Great 66
Okay 29
PC
Amazing 351
Great 967
Okay 526
```

In [227…
```
Task1 = pd.DataFrame(final,columns=list(phrase)).T
# Task1 = pd.DataFrame(rows=list(platforms))

Task1
# Task1.index.names = [platforms]
```

Out[227…

	0	1	2	3	4
Amazing	181	55	196	35	351

	0	1	2	3	4
Great	379	83	445	66	967
Okay	196	36	256	29	526

In [228…
```
Task1.set_axis(platforms, axis=1, inplace=False)
```

Out[228…

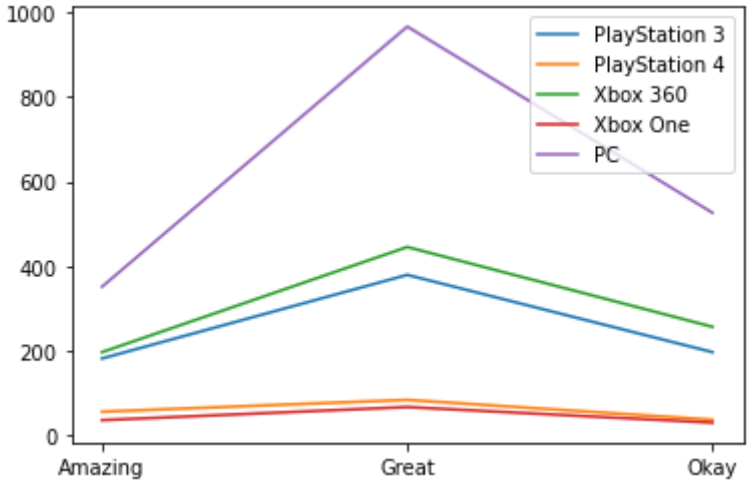
	PlayStation 3	PlayStation 4	Xbox 360	Xbox One	PC
Amazing	181	55	196	35	351
Great	379	83	445	66	967
Okay	196	36	256	29	526

In [229…
```
import matplotlib.pyplot as plt
```

Line Chart

In [230…
```
plt.plot(Task1)
plt.legend(platforms)
```

Out[230… `<matplotlib.legend.Legend at 0x13a7d6442b0>`

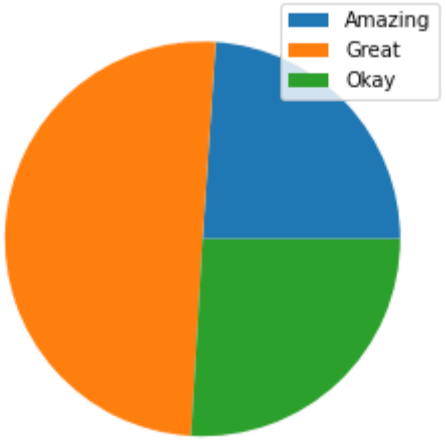


PieChart

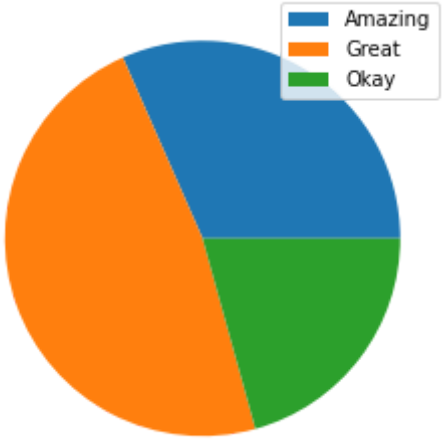
In [231…

```
for i in range(len(platforms)):
    f = plt.figure(figsize=(15,15))
    plt.subplot(2,3,i+1)
    plt.pie(Task1[i])
    print(platforms[i])
    plt.legend(phrase)
    plt.show()
```

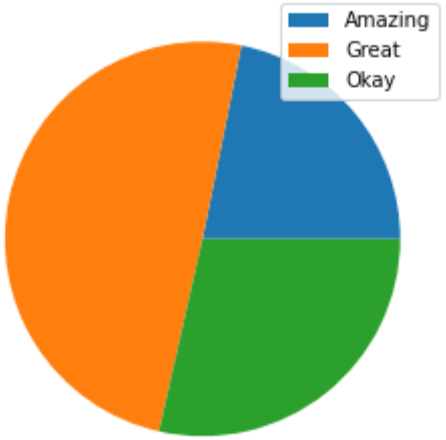
```
PlayStation 3
```



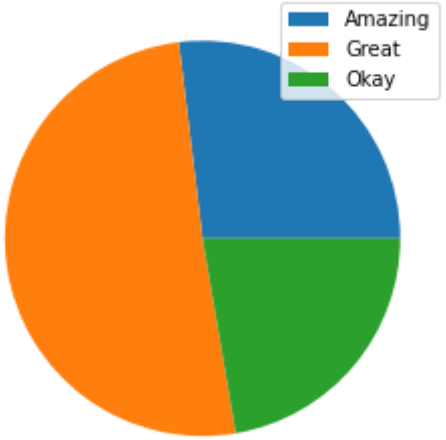

```
PlayStation 4
```



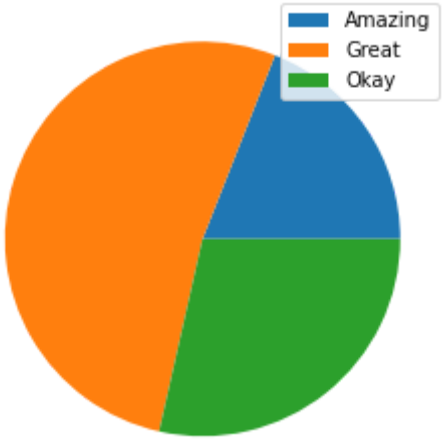

```
Xbox 360
```



Xbox One

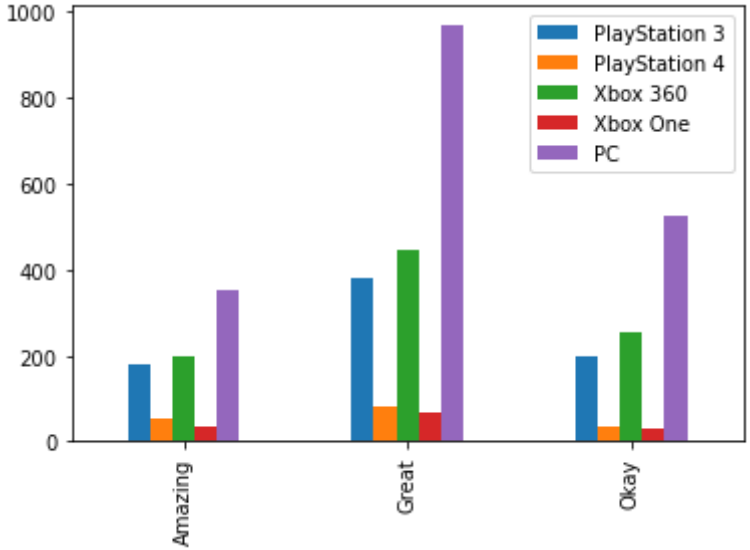


PC



In [232…
```
Task1.plot.bar(legend = platforms).legend(platforms)
```

Out[232…
```
<matplotlib.legend.Legend at 0x13a7df91ee0>
```



In [233…
```
ign_master.head()
```

Out[233…

	Unnamed: 0	score_phrase	title	url	platform	score	genre	editors_choice	release_year	release_month	release_day
0	0	Amazing	LittleBigPlanet PS Vita	/games/littlebigplanet-vita/vita-98907	PlayStation Vita	9.0	Platformer	Y	2012	9	12
1	1	Amazing	LittleBigPlanet PS Vita -- Marvel Super Hero E...	/games/littlebigplanet-ps-vita-marvel-super-he...	PlayStation Vita	9.0	Platformer	Y	2012	9	12
2	2	Great	Splice: Tree of Life	/games/splice/ipad-141070	iPad	8.5	Puzzle	N	2012	9	12
3	3	Great	NHL 13	/games/nhl-13/xbox-360-128182	Xbox 360	8.5	Sports	N	2012	9	11
4	4	Great	NHL 13	/games/nhl-13/ps3-128181	PlayStation 3	8.5	Sports	N	2012	9	11

In [234…
```
# ign_master['Performance'] =
```

In [235…
```
ign_master.loc[ign_master['score'] > ign_master.score.mean(), 'Performance'] = "Above Average"
ign_master.loc[ign_master['score'] <= ign_master.score.mean(), 'Performance'] = "Below Average"
```

In [236…
```
ign_master['Performance']
```

Out[236…
```
0        Above Average
1        Above Average
2        Above Average
3        Above Average
4        Above Average
            ...
18620    Above Average
18621    Above Average
18622    Below Average
18623    Above Average
18624    Above Average
Name: Performance, Length: 18625, dtype: object
```

In [238…
```
list(Performance)
```

Out[238…
```
['Above Average', 'Below Average']
```

In [239…
```
ign_master['Performance'].value_counts()
```

Out[239…
```
Above Average    11373
Below Average     7252
Name: Performance, dtype: int64
```

In [240…
```
ign_master[(ign_master['platform'] == 'PC')].value_counts('Performance')
```

Out[240…
```
Performance
Above Average    2191
Below Average    1179
dtype: int64
```

In [241…
```
Task2_plat = []
Task2_rating = []
for x in platforms:
    print(x,ign_master[(ign_master['platform'] == x)].value_counts('Performance'))
    T, Y = x,ign_master[(ign_master['platform'] == x)].value_counts('Performance')

#     Task2.append(x,ign_master[(ign_master['platform'] == x)].value_counts('Performance'))

    Task2_plat.append(T)
    Task2_rating.append(Y)
```

```
PlayStation 3 Performance
Above Average    889
Below Average    467
dtype: int64
PlayStation 4 Performance
Above Average    208
Below Average     69
dtype: int64
Xbox 360 Performance
Above Average    1021
Below Average     610
dtype: int64
Xbox One Performance
Above Average    155
Below Average     53
dtype: int64
PC Performance
Above Average    2191
Below Average    1179
dtype: int64
```

In [242…
```
Task2_plat
```

Out[242…
```
['PlayStation 3', 'PlayStation 4', 'Xbox 360', 'Xbox One', 'PC']
```

In [243…
```
Task2_DF = pd.DataFrame(Task2_rating)
```

In [244…
```
Task2_DF= Task2_DF.set_axis(platforms, axis=0, inplace=False)
```

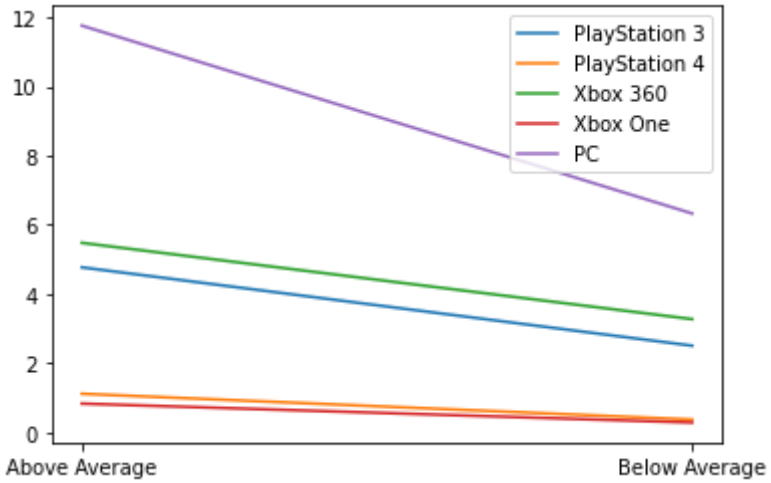
In [245…
```
Task2_DF['Above Average'] = (Task2_DF['Above Average'])*(100/18625)
```

In [246…
```
Task2_DF['Below Average'] = (Task2_DF['Below Average'])*(100/18625)
```

In [252…
```
Task2_DF= Task2_DF.T
```

In [253…
```
plt.plot(Task2_DF)
plt.legend(platforms)
```

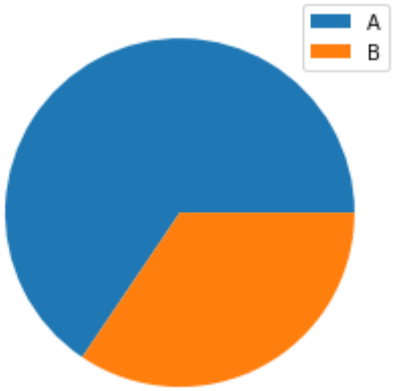
Out[253…
```
<matplotlib.legend.Legend at 0x13a7dc9eb80>
```



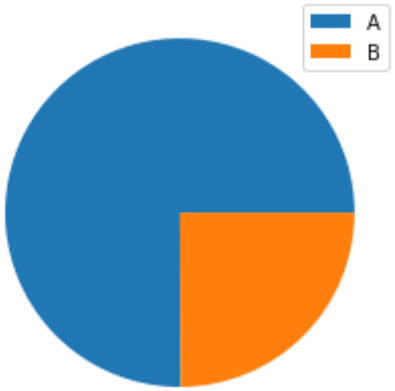
In [284…
```
#Pie Chart
for i in Task2_DF:
    print(i)
#     f = plt.figure(figsize=(15,15))
#     plt.subplot(2,3,i+1)
    plt.pie(Task2_DF[i])
```

```
    plt.legend('AB')
#     print(platforms[i])
#     plt.legend("A","B")
#     plt.legend("Above", "Below")
    plt.show()
```

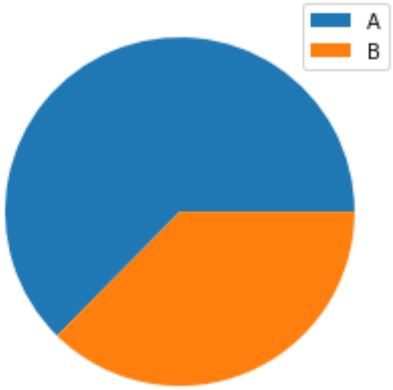
PlayStation 3



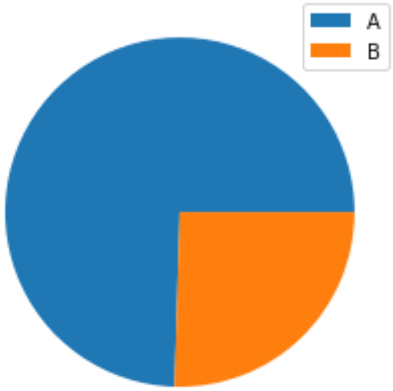
PlayStation 4



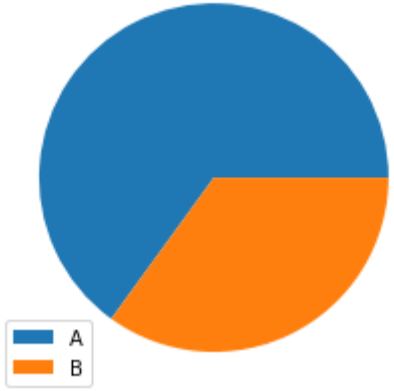
Xbox 360



Xbox One



PC

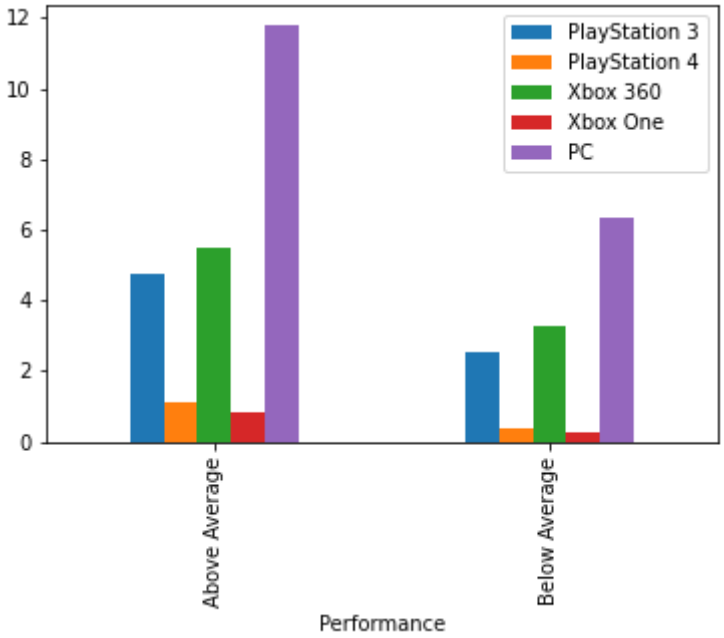


In [279…

```
#BarPlot

Task2_DF.plot.bar(legend = platforms).legend(platforms)
```

Out[279… <matplotlib.legend.Legend at 0x13a7df3e430>



In [ ]:

In [ ]: