Python project

In [76]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [77]:

```
df = pd.read_csv("HR_comma_sep.csv")
```

In [78]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14999 entries, 0 to 14998
Data columns (total 10 columns):
 #   Column                 Non-Null Count  Dtype  
---  ------                 --------------  -----  
 0   satisfaction_level     14999 non-null  float64
 1   last_evaluation        14999 non-null  float64
 2   number_project         14999 non-null  int64  
 3   average_montly_hours   14999 non-null  int64  
 4   time_spend_company     14999 non-null  int64  
 5   Work_accident          14999 non-null  int64  
 6   left                   14999 non-null  int64  
 7   promotion_last_5years  14999 non-null  int64  
 8   Department             14999 non-null  object 
 9   salary                 14999 non-null  object 
dtypes: float64(2), int64(6), object(2)
memory usage: 1.1+ MB
```

In [79]:

```
df.head(5)
```

Out[79]:

	satisfaction_level	last_evaluation	number_project	average_montly_hours	time_spend_company	Work_accident	left	promotion_last_5years	Department	salary
0	0.38	0.53	2	157	3	0	1	0	sale	low

	satisfaction_level	last_evaluation	number_project	average_monthly_hours	time_spend_company	Work_accident	left	promotion_last_5years	Departments	salary
1	0.80	0.86	5	262	6	0	1	0	sales	medium
2	0.11	0.88	7	272	4	0	1	0	sales	medium
3	0.72	0.87	5	223	5	0	1	0	sales	low
4	0.37	0.52	2	159	3	0	1	0	sales	low

In [80]:

```
df.describe()
```

Out[80]:

	satisfaction_level	last_evaluation	number_project	average_monthly_hours	time_spend_company	Work_accident	left	promotion_last_5years
count	14999.000000	14999.000000	14999.000000	14999.000000	14999.000000	14999.000000	14999.000000	14999.000000
mean	0.612834	0.716102	3.803054	201.050337	3.498233	0.144610	0.238083	0.021268
std	0.248631	0.171169	1.232592	49.943099	1.460136	0.351719	0.425924	0.144281
min	0.090000	0.360000	2.000000	96.000000	2.000000	0.000000	0.000000	0.000000
25%	0.440000	0.560000	3.000000	156.000000	3.000000	0.000000	0.000000	0.000000
50%	0.640000	0.720000	4.000000	200.000000	3.000000	0.000000	0.000000	0.000000
75%	0.820000	0.870000	5.000000	245.000000	4.000000	0.000000	0.000000	0.000000
max	1.000000	1.000000	7.000000	310.000000	10.000000	1.000000	1.000000	1.000000

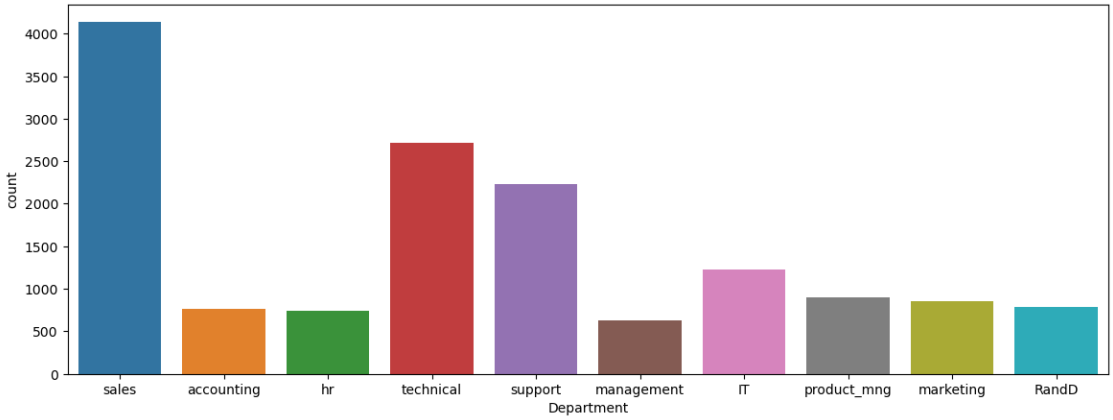
In [ ]:

In [7]:

```
#data visulization
fig, ax = plt.subplots(figsize=(14, 5))
sns.countplot(data=df, x="Department", saturation=0.75)
```

Out[7]:

```
<AxesSubplot:xlabel='Department', ylabel='count'>
```



In [8]:

```
df['Department'].value_counts()
```

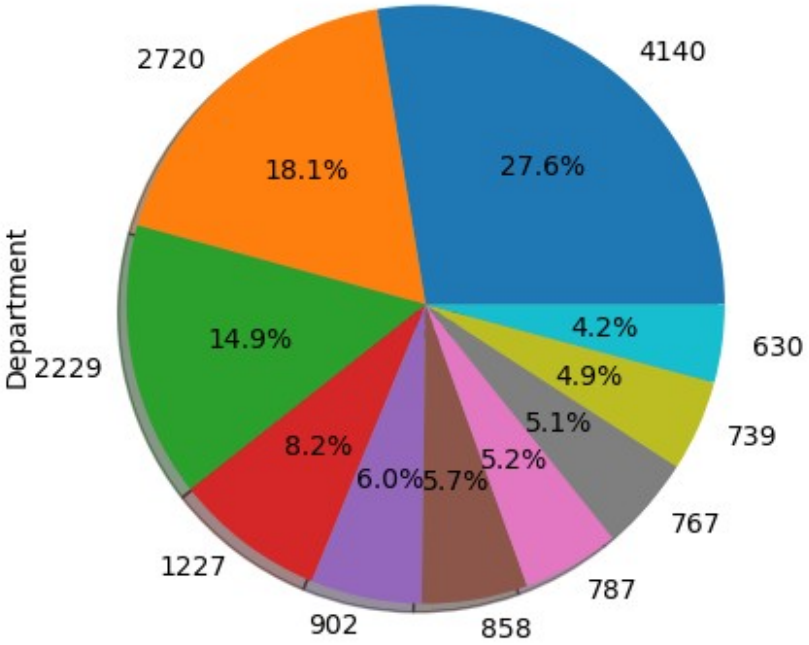
Out[8]:

```
sales          4140
technical      2720
support        2229
IT             1227
product_mng     902
marketing       858
RandD           787
accounting      767
hr              739
management      630
Name: Department, dtype: int64
```

In [9]:

```
#with department values
plt.figure(figsize=(5,5))
df['Department'].value_counts().plot(kind='pie',
labels=df['Department'].value_counts(), autopct='%1.1f%%',
```

```
shadow=True)
plt.show()
```

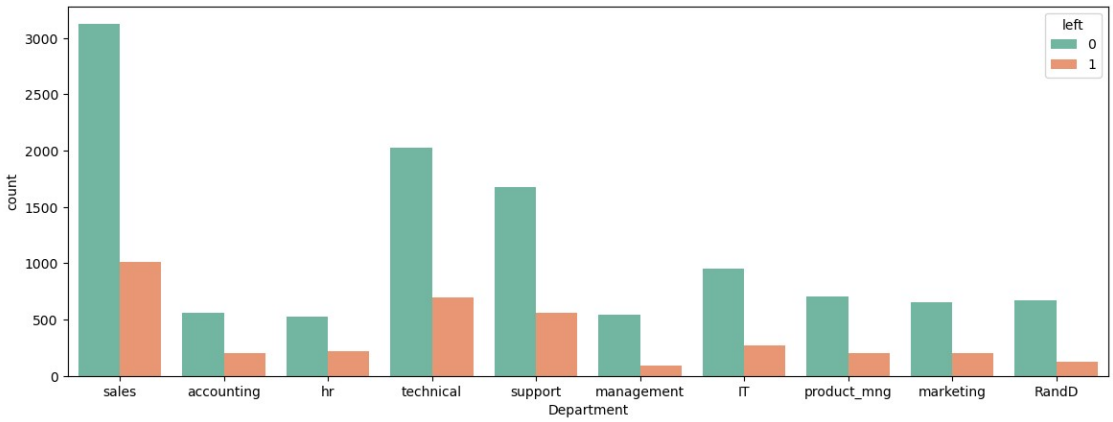


In [10]:

```
#data visualization
fig, ax = plt.subplots(figsize=(14, 5))
sns.countplot(data=df, x='Department', hue='left', palette='Set2')
```

Out[10]:

```
<AxesSubplot:xlabel='Department', ylabel='count'>
```

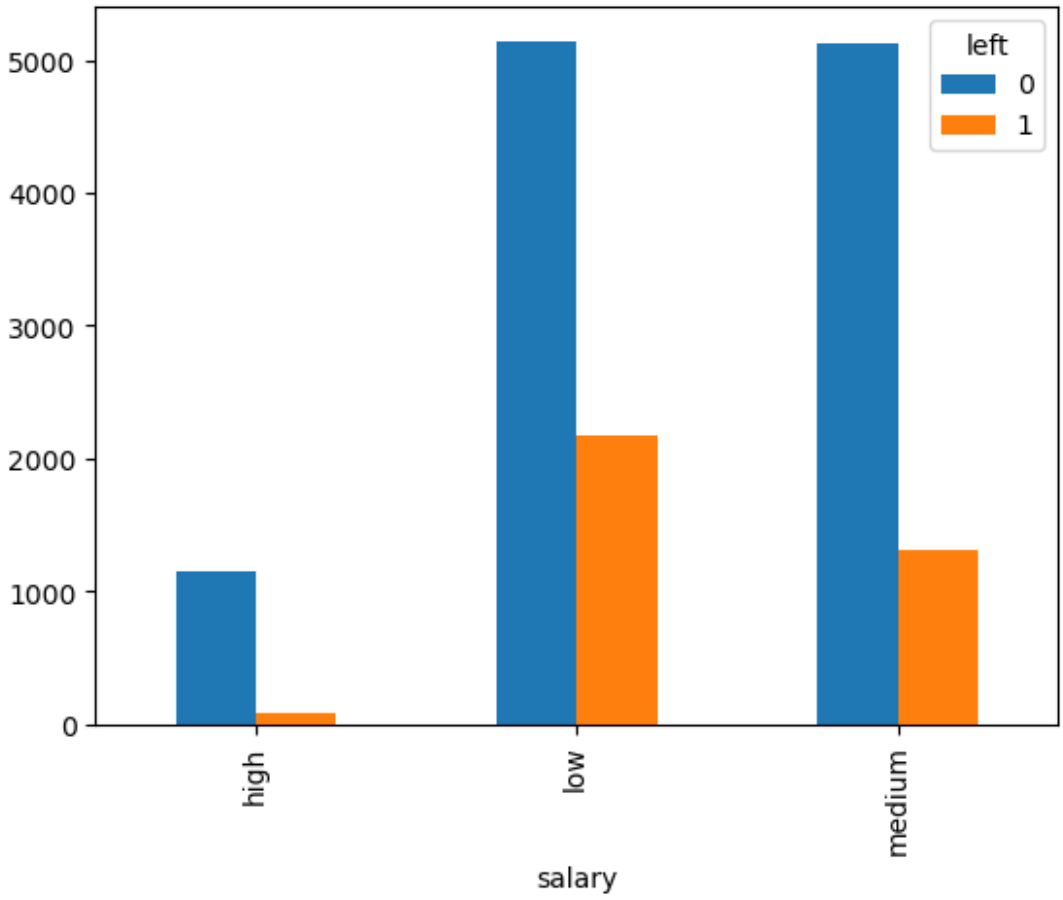


In [39]:

```
#Realation between salary and employees retention
pd.crosstab(df.salary,df.left).plot(kind='bar')
```

Out[39]:

```
<AxesSubplot:xlabel='salary'>
```

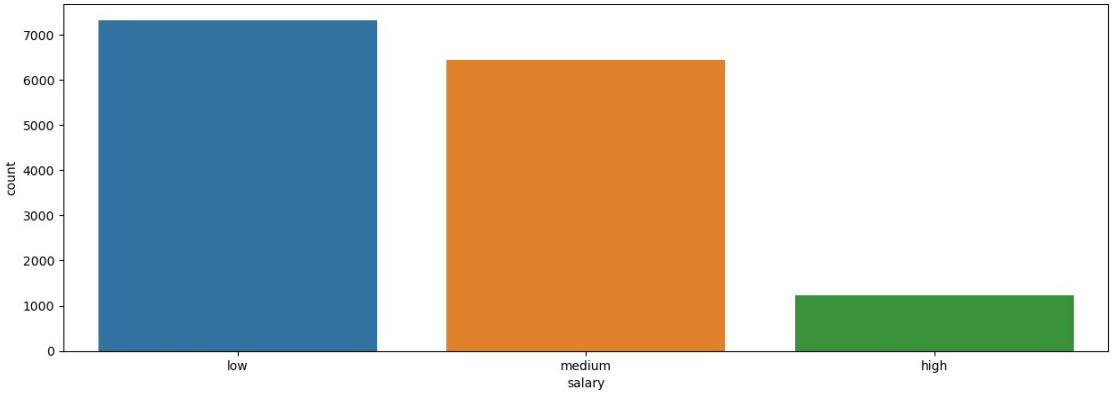


In [12]:

```
#data visualization
fig, ax = plt.subplots(figsize=(15, 5))
sns.countplot(data=df, x="salary")
```

Out[12]:

```
<AxesSubplot:xlabel='salary', ylabel='count'>
```



In [13]:

```
df.groupby(by=['Work_accident'])['number_project'].mean()
```

Out[13]:

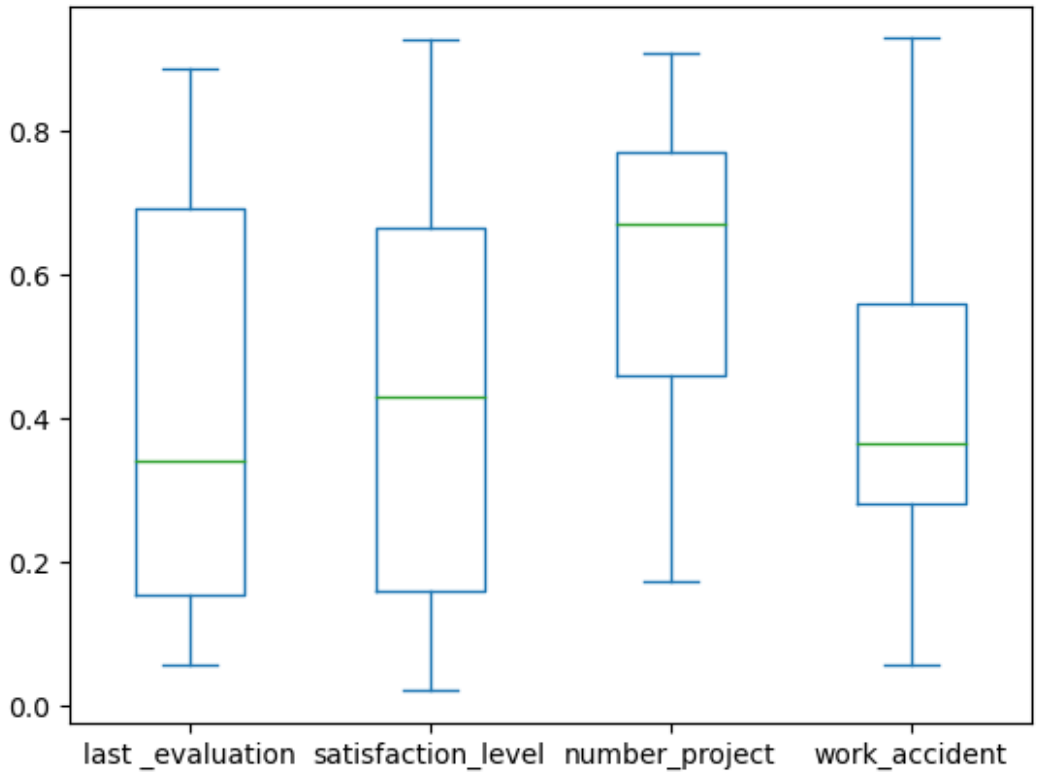
```
Work_accident
0    3.805456
1    3.788843
Name: number_project, dtype: float64
```

In [15]:

```
df = pd.DataFrame(np.random.rand(10, 4), columns=["last
_evaluation","satisfaction_level","number_project","work_accident"])
df.plot.box()
```

Out[15]:

```
<AxesSubplot:>
```

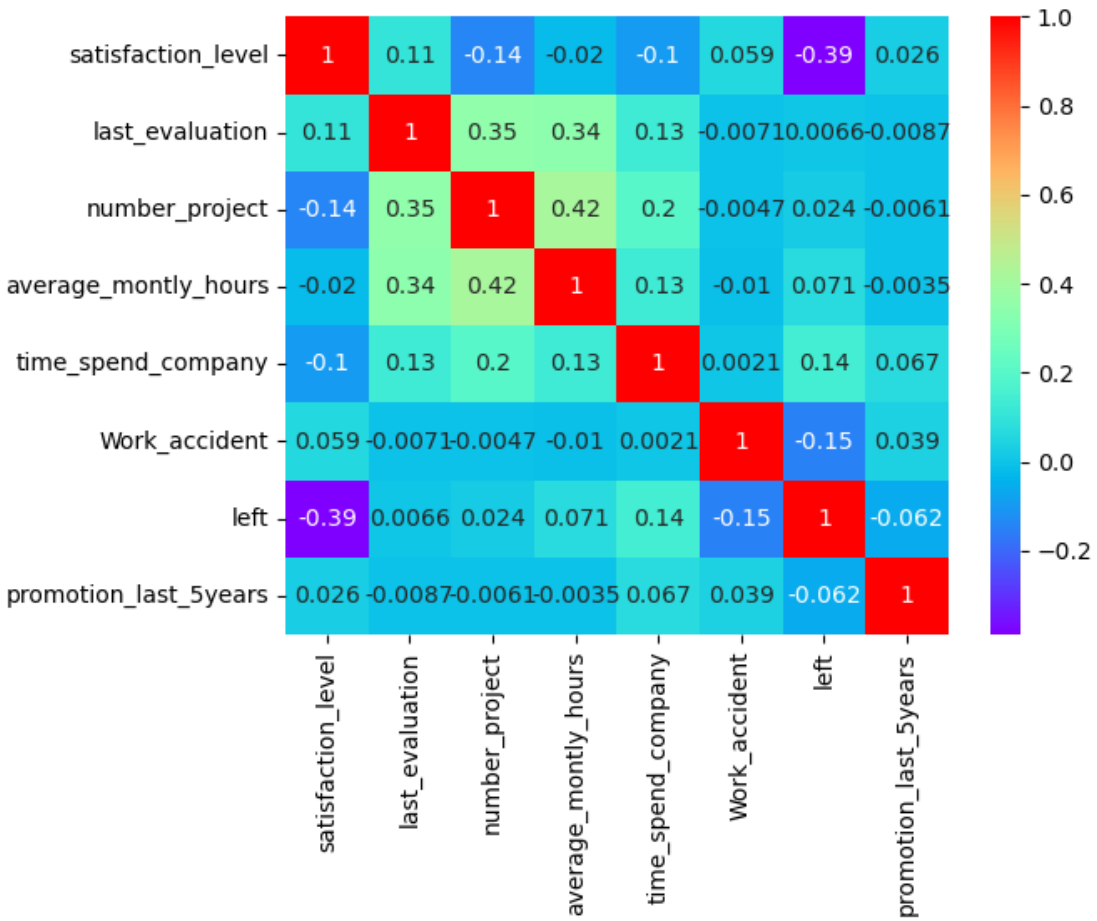


In [49]:

```
#exploratory data analysis
sns.heatmap(df.corr(), annot = True, cmap = 'rainbow')
```

Out[49]:

```
<AxesSubplot:>
```

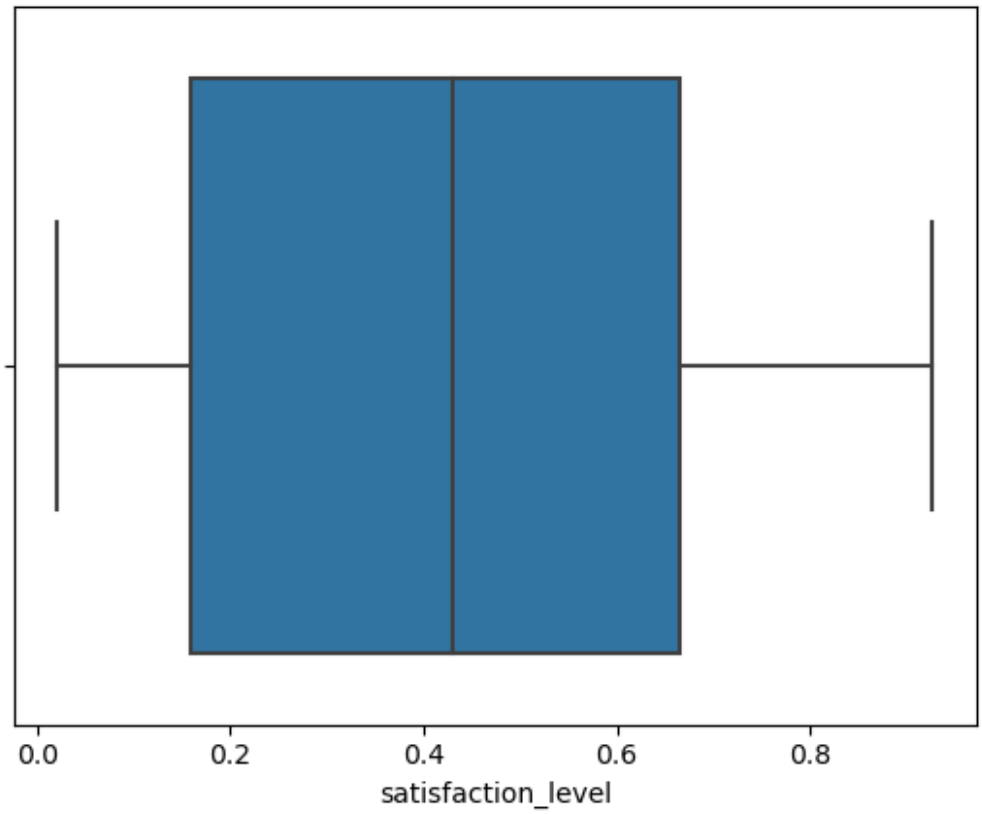


In [17]:

```
# Visualize distribution of 'satisfaction_level'
sns.boxplot(x='satisfaction_level', data=df)
```

Out[17]:

```
<AxesSubplot:xlabel='satisfaction_level'>
```

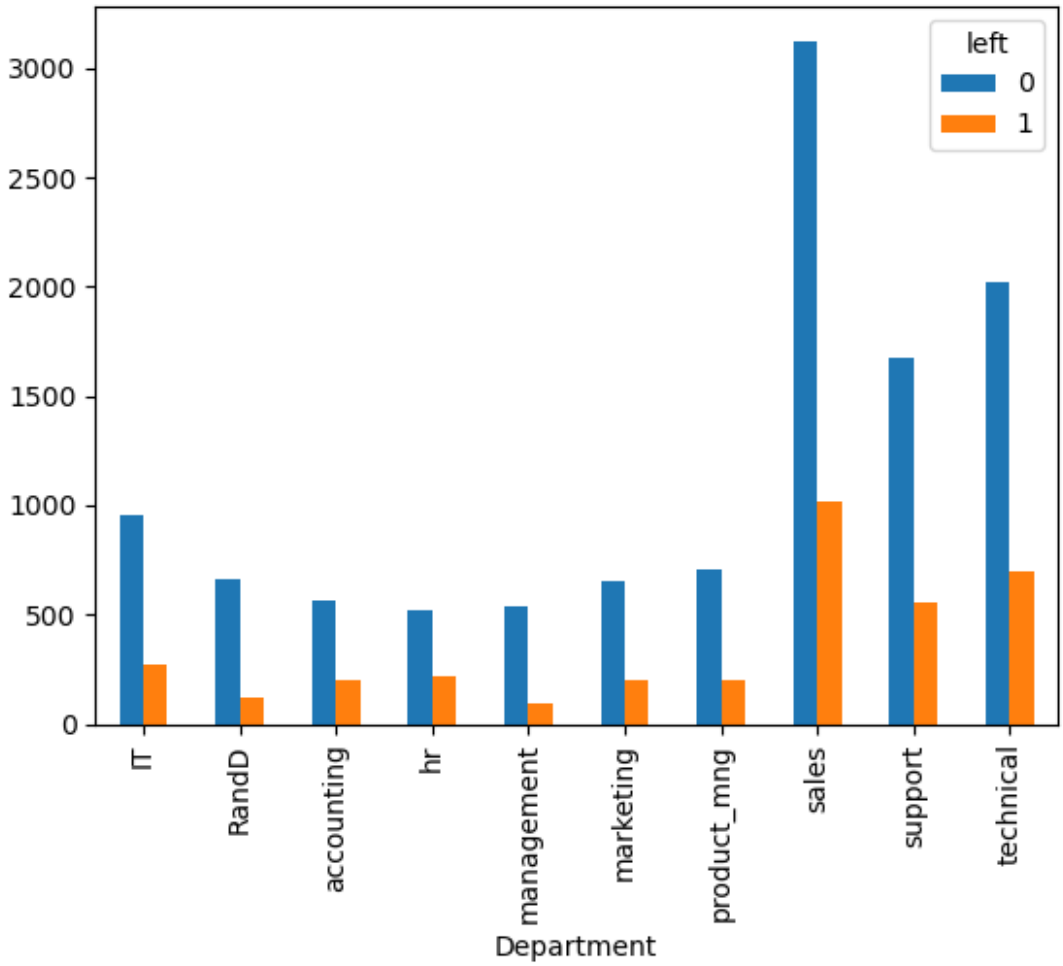


In [31]:

```
#Realation between Department and employees retention
pd.crosstab(df.Department, df.left).plot(kind = 'bar')
```

Out[31]:

```
<AxesSubplot:xlabel='Department'>
```



In [32]:

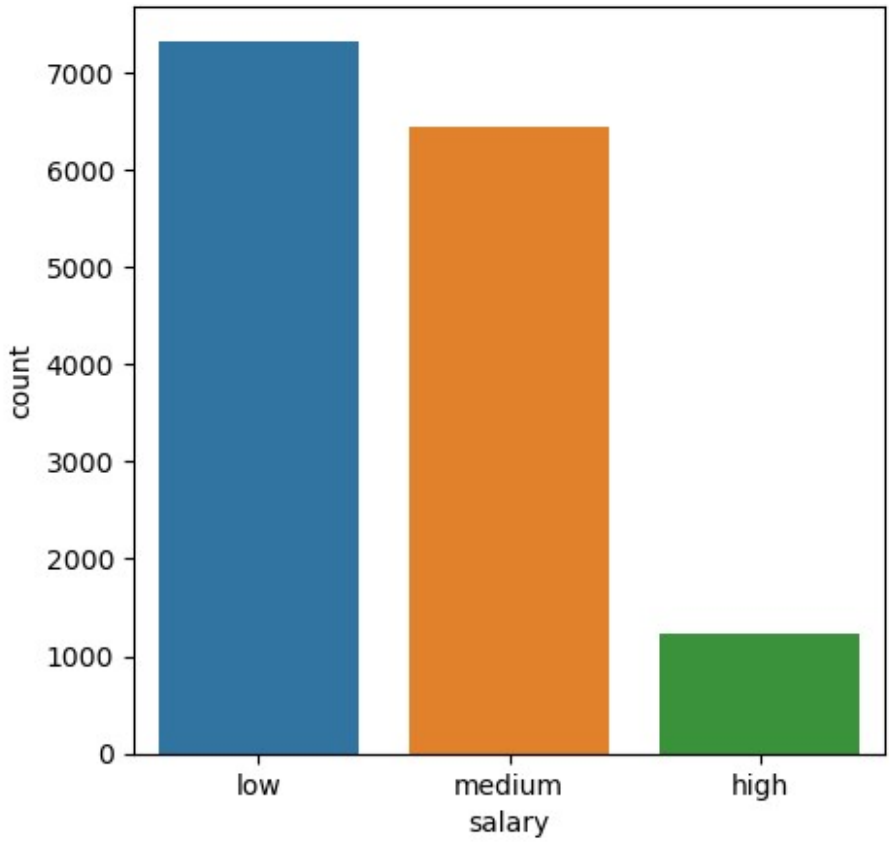
```
import seaborn as sns
import matplotlib.pyplot as plt
%matplotlib inline
```

In [33]:

```
#count of salary
fig, ax = plt.subplots(figsize=(5, 5))
sns.countplot(data=df, x="salary")
```

Out[33]:

```
<AxesSubplot:xlabel='salary', ylabel='count'>
```

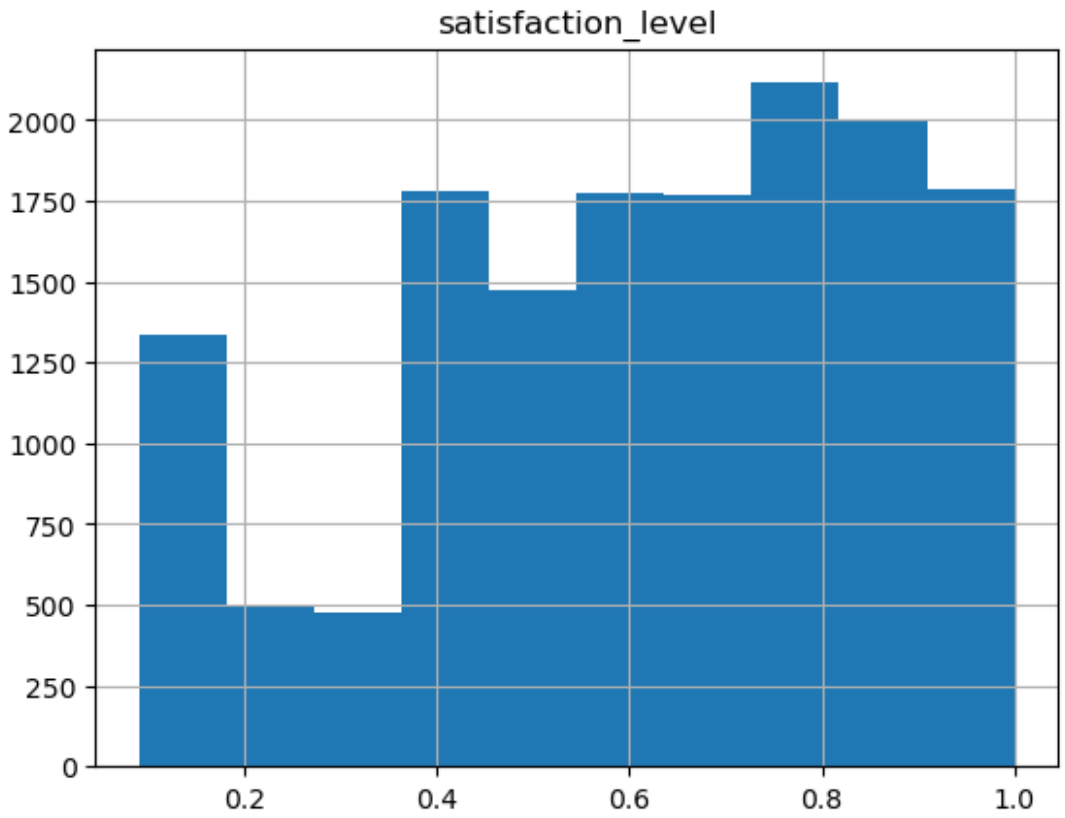


In [40]:

```
# Find the number of employees in each satisfaction level
df.hist(column=['satisfaction_level'], cumulative=False, bins=10)
```

Out[40]:

```
array([[<AxesSubplot:title={'center':'satisfaction_level'}>]],
      dtype=object)
```

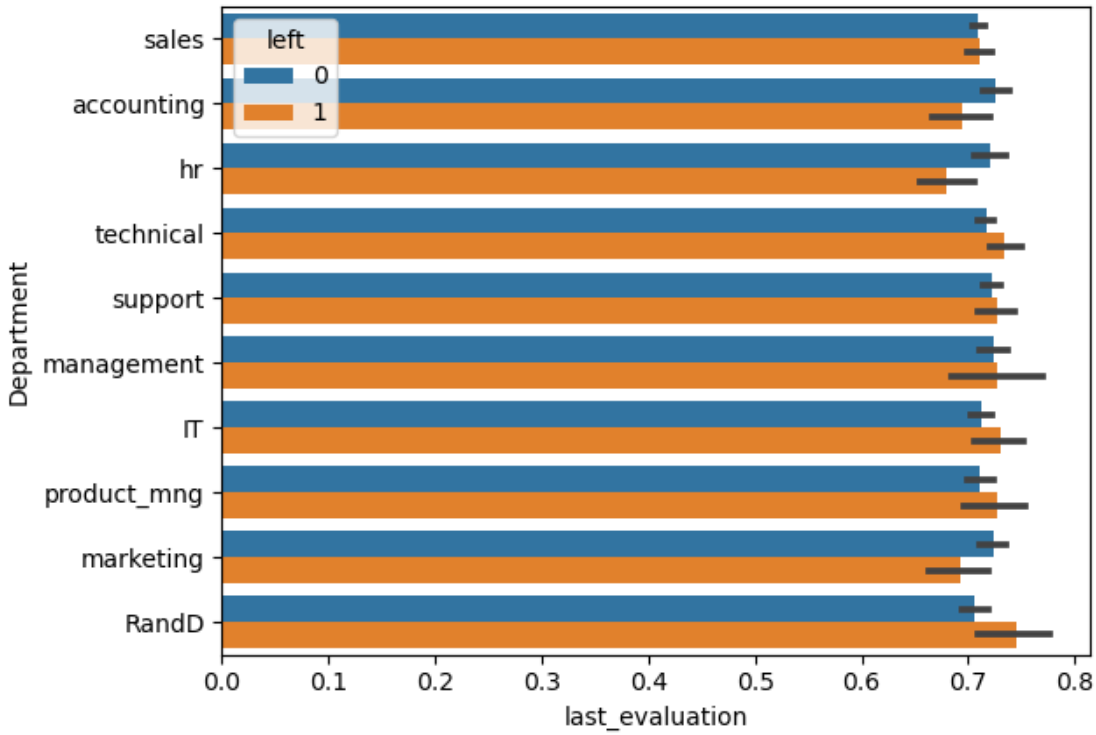


In [50]:

```
# Which department had last evaluation and the employee left the firm?
sns.barplot(x='last_evaluation', y = 'Department',data = df, hue =
'left')
```

Out[50]:

```
<AxesSubplot:xlabel='last_evaluation', ylabel='Department'>
```



In [107]:

```
#pivot tables
```

In [109]:

```
import pandas as pd
import numpy as np
```

In [110]:

```
df = pd.read_csv("HR_comma_sep.csv")
df.head()
```

Out[110]:

	satisfaction_level	last_evaluation	number_project	average_monthly_hours	time_spend_company	Work_accident	left	promotion_last_5years	Department	salary
0	0.38	0.53	2	157	3	0	1	0	sales	low
1	0.80	0.86	5	262	6	0	1	0	sales	medium
2	0.11	0.88	7	272	4	0	1	0	sale	med

	satisfaction_level	last_evaluation	number_project	average_monthly_hours	time_spend_company	Work_accident	left	promotion_last_5years	Departments	salary
3	0.72	0.87	5	223	5	0	1	0	sales	low
4	0.37	0.52	2	159	3	0	1	0	sales	low

In [111]:

```
pd.pivot_table(df,index='time_spend_company')
```

Out[111]:

time_spend_company	Work_accident	average_monthly_hours	last_evaluation	left	number_project	promotion_last_5years	satisfaction_level
2	0.172010	200.133169	0.717596	0.016338	3.687423	0.016646	0.697078
3	0.138910	186.632935	0.668721	0.246159	3.327798	0.020798	0.626314
4	0.124364	223.455221	0.767927	0.348064	4.627689	0.013688	0.467517
5	0.116090	222.978955	0.813666	0.565513	4.519348	0.011541	0.610305
6	0.149025	212.051532	0.754875	0.291086	4.213092	0.023677	0.603440
7	0.138298	200.744681	0.682766	0.000000	3.851064	0.191489	0.635957
8	0.271605	193.802469	0.711975	0.000000	3.777778	0.061728	0.665062
10	0.233645	199.224299	0.731495	0.000000	3.682243	0.074766	0.655327

In [112]:

```
pd.pivot_table(df,index='Department',values='number_project')
```

Out[112]:

	number_project
Department	
IT	3.816626
RandD	3.853875
accounting	3.825293
hr	3.654939
management	3.860317
marketing	3.687646
product_mng	3.807095
sales	3.776329
support	3.803948
technical	3.877941

In [113]:

```
pd.pivot_table(df,index='Department',values='number_project',columns='
left')
```

Out[113]:

left	0	1
Department		
IT	3.756813	4.025641
RandD	3.822823	4.024793
accounting	3.808171	3.872549
hr	3.702290	3.539535
management	3.812616	4.142857
marketing	3.720611	3.581281
product_mng	3.795455	3.848485
sales	3.789187	3.736686
support	3.783751	3.864865
technical	3.814632	4.061693

In [114]:

```
pd.pivot_table(df,index='Department',values='number_project',columns='
left',aggfunc='median')
```

Out[114]:

left	0	1
Department		
IT	4	4
RandD	4	4
accounting	4	4
hr	4	2
management	4	4
marketing	4	2
product_mng	4	4
sales	4	4
support	4	4
technical	4	4

In [115]:

```
pd.pivot_table(df,index='Department',values='number_project',columns='
left',
               aggfunc='median',margins=True)
```

Out[115]:

left	0	1	All
Department			
IT	4	4	4.0
RandD	4	4	4.0
accounting	4	4	4.0
hr	4	2	4.0
management	4	4	4.0
marketing	4	2	4.0
product_mng	4	4	4.0
sales	4	4	4.0
support	4	4	4.0
technical	4	4	4.0
All	4	4	4.0

In [121]:

```
pd.pivot_table(df,index=['Department','salary'],values=['number_projec
t'],
               columns=['left'],aggfunc=np.mean,margins=True)
```

Out[121]:

		number_project		
	left	0	1	All
Department	salary			
IT	high	3.886076	3.500000	3.867470
	low	3.743707	3.924419	3.794745
	medium	3.746575	4.226804	3.833645
RandD	high	3.851064	2.750000	3.764706
	low	3.792880	3.872727	3.804945
	medium	3.848387	4.241935	3.913978
accounting	high	3.985507	2.800000	3.905405
	low	3.795367	3.818182	3.801676
	medium	3.770213	3.980000	3.832836
hr	high	3.871795	4.000000	3.888889
	low	3.740741	3.565217	3.692537
	medium	3.636364	3.495726	3.590529
management	high	3.767857	6.000000	3.777778
	low	3.801653	3.728814	3.777778
	medium	3.871134	4.870968	4.008889
marketing	high	3.605634	2.000000	3.425000
	low	3.775362	3.698413	3.751244
	medium	3.698052	3.573529	3.675532
product_mng	high	3.741935	3.333333	3.705882
	low	3.745665	4.085714	3.824834
	medium	3.864865	3.597701	3.804178
sales	high	3.807843	4.785714	3.858736
	low	3.781027	3.711621	3.757980
	medium	3.793737	3.745875	3.785553
support	high	3.804511	3.625000	3.794326
	low	3.782034	3.796915	3.787086
	medium	3.781888	4.044304	3.825902
technical	high	3.715909	3.200000	3.651741
	low	3.828974	4.124339	3.910350
	medium	3.818288	4.054422	3.878814
All		3.786664	3.855503	3.803054

In [ ]: