w4asnev8j

June 1, 2025

[ ]:

People Charm', a growing company is facing a high attrition rate among their employees which in turn affects their business due to lack of expertise and experience. Their HR department is assigned the task to reduce the attrition rate by retaining employees who are about to churn out. They need to recommend special plans or strategies which will help them to retain their employees which in turn will help them to grow bigger as a company

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

[ ]:
```
df=pd.read_csv('/content/employe[1].csv')
df
```

[ ]:

	satisfactoryLevel	lastEvaluation	numberOfProjects	avgMonthlyHours	\
0	0.38	0.53	2	157	
1	0.80	0.86	5	262	
2	0.11	0.88	7	272	
3	0.37	0.52	2	159	
4	0.41	0.50	2	153	
…	…	…	…	…	
14994	0.11	0.85	7	275	
14995	0.99	0.83	4	274	
14996	0.72	0.72	4	175	
14997	0.24	0.91	5	177	
14998	0.77	0.83	6	271	

	timeSpent.company	workAccident	left	promotionInLast5years	\
0	3	0	1	0	
1	6	0	1	0	
2	4	0	1	0	
3	3	0	1	0	
4	3	0	1	0	
…	…	…	…	…	
14994	4	0	1	0	
14995	2	0	0	0	

```
14996                  4             0     0                       0
14997                  5             0     0                       0
14998                  3             0     0                       0

            dept  salary
0          sales     low
1          sales  medium
2          sales  medium
3          sales     low
4          sales     low
…          …     …
14994    support  medium
14995      sales     low
14996  technical     low
14997      sales     low
14998    support     low

[14999 rows x 10 columns]
```

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

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 14999 entries, 0 to 14998
Data columns (total 10 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   satisfactoryLevel      14999 non-null  float64
 1   lastEvaluation         14999 non-null  float64
 2   numberOfProjects       14999 non-null  int64
 3   avgMonthlyHours        14999 non-null  int64
 4   timeSpent.company      14999 non-null  int64
 5   workAccident           14999 non-null  int64
 6   left                   14999 non-null  int64
 7   promotionInLast5years  14999 non-null  int64
 8   dept                   14999 non-null  object
 9   salary                 14999 non-null  object
dtypes: float64(2), int64(6), object(2)
memory usage: 1.1+ MB
```

[ ]:
```
df.isnull().sum()
```

[ ]:
```
satisfactoryLevel        0
lastEvaluation           0
numberOfProjects         0
avgMonthlyHours          0
timeSpent.company        0
workAccident             0
```

```
left                    0
promotionInLast5years   0
dept                    0
salary                  0
dtype: int64
```

[ ]:
```
df.duplicated().sum()
```

[ ]:
```
np.int64(3008)
```

[ ]:
```
df[df.duplicated()]
```

[ ]:
```
      satisfactoryLevel  lastEvaluation  numberOfProjects  avgMonthlyHours  \
263                0.46            0.57                 2              139
877                0.37            0.51                 2              127
974                0.42            0.53                 2              142
1017               0.40            0.50                 2              127
1241               0.10            0.85                 6              266
…                   …               …                   …                …
14985              0.95            0.84                 3              270
14987              0.37            0.45                 2              126
14988              0.43            0.57                 2              157
14993              0.61            0.89                 3              242
14994              0.11            0.85                 7              275

      timeSpent.company  workAccident  left  promotionInLast5years     dept  \
263                   3             0     1                      0    sales
877                   3             0     1                      0    sales
974                   3             0     1                      0    sales
1017                  3             0     1                      0       IT
1241                  4             0     1                      0    sales
…                   …             …     …                      …       …
14985                 3             1     0                      1    sales
14987                 3             0     1                      0  support
14988                 3             0     1                      0    sales
14993                10             0     0                      0    sales
14994                 4             0     1                      0  support

      salary
263       low
877    medium
974       low
1017      low
1241      low
…          …
14985  medium
14987  medium
```

```
14988     low
14993    high
14994  medium

[3008 rows x 10 columns]
```

[ ]:
```
df.drop_duplicates(inplace=True)
```

[ ]:
```
df.duplicated().sum()
```

[ ]:
```
np.int64(0)
```

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

```
<class 'pandas.core.frame.DataFrame'>
Index: 11991 entries, 0 to 14998
Data columns (total 10 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   satisfactoryLevel      11991 non-null  float64
 1   lastEvaluation         11991 non-null  float64
 2   numberOfProjects       11991 non-null  int64
 3   avgMonthlyHours        11991 non-null  int64
 4   timeSpent.company      11991 non-null  int64
 5   workAccident           11991 non-null  int64
 6   left                   11991 non-null  int64
 7   promotionInLast5years  11991 non-null  int64
 8   dept                   11991 non-null  object
 9   salary                 11991 non-null  object
dtypes: float64(2), int64(6), object(2)
memory usage: 1.0+ MB
```

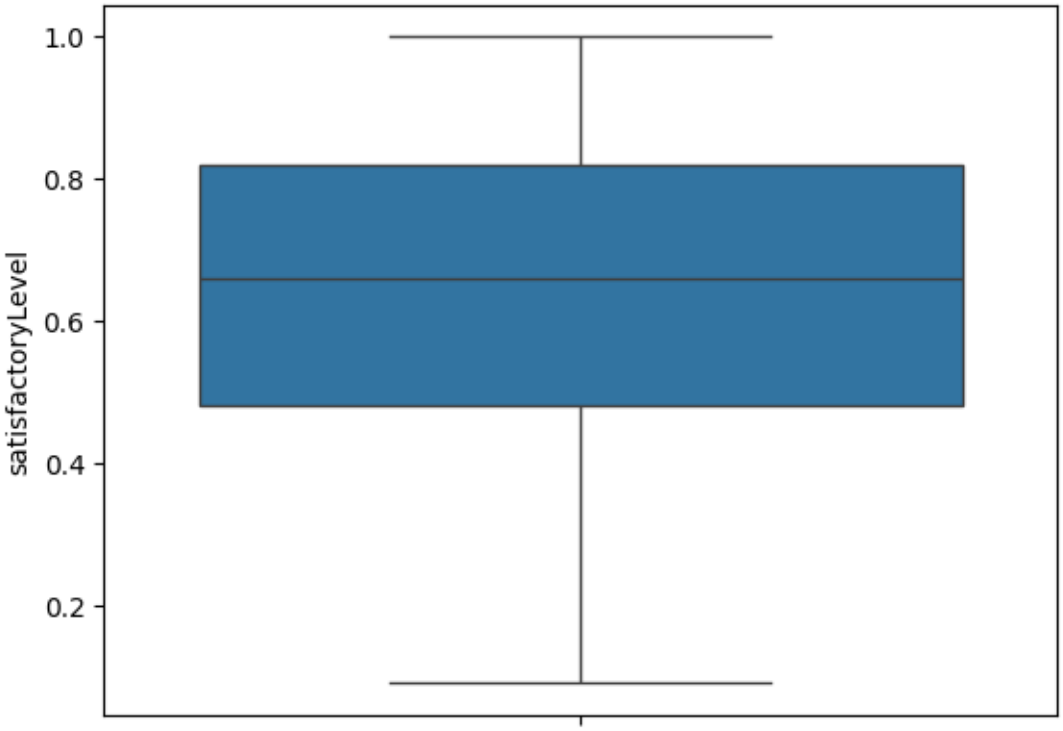
[ ]:
```
df.columns
```

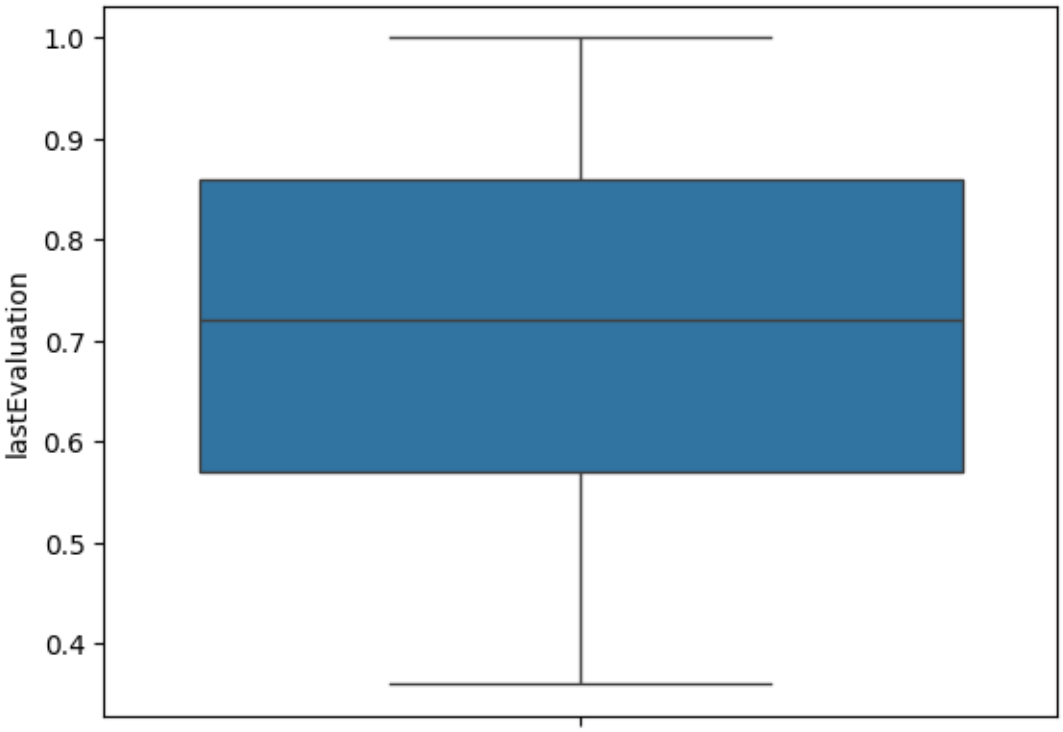
[ ]:
```
Index(['satisfactoryLevel', 'lastEvaluation', 'numberOfProjects',
       'avgMonthlyHours', 'timeSpent.company', 'workAccident', 'left',
       'promotionInLast5years', 'dept', 'salary'],
      dtype='object')
```

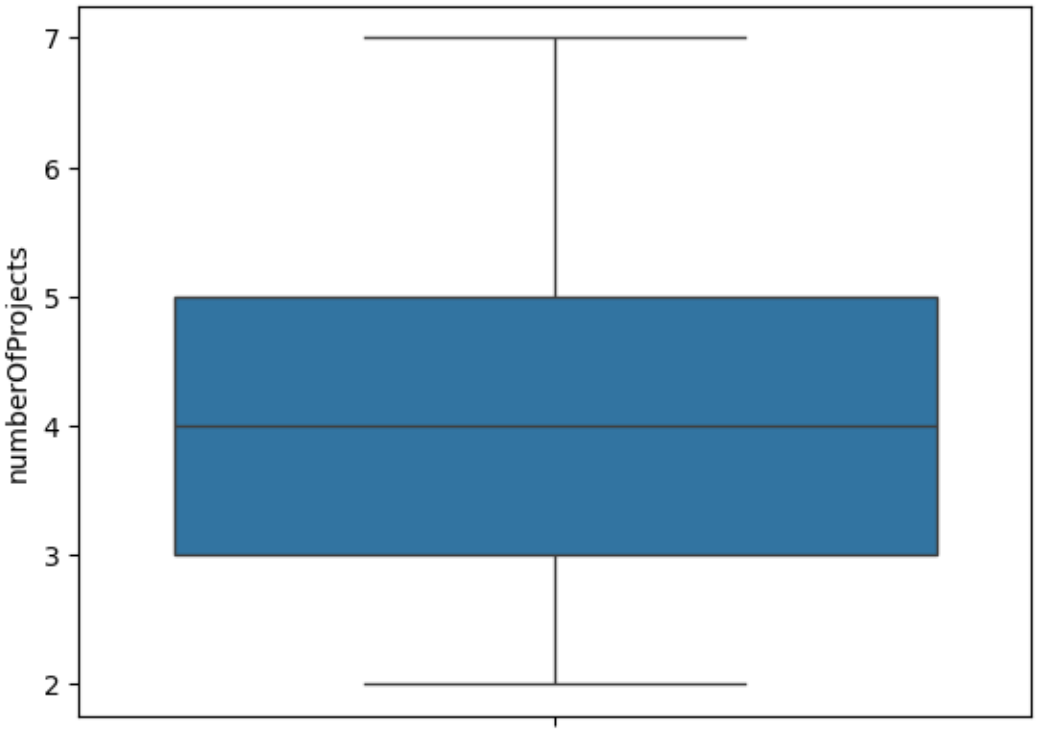
[ ]:
```
# prompt: box plot for finding outlier

import matplotlib.pyplot as plt
# Assuming 'df' is your DataFrame and you want to find outliers in a numerical␣
 ↪column,
# replace 'NumericalColumnName' with the actual column name
plt.figure(figsize=(8, 6))
sns.boxplot(x=df['Age'])
```

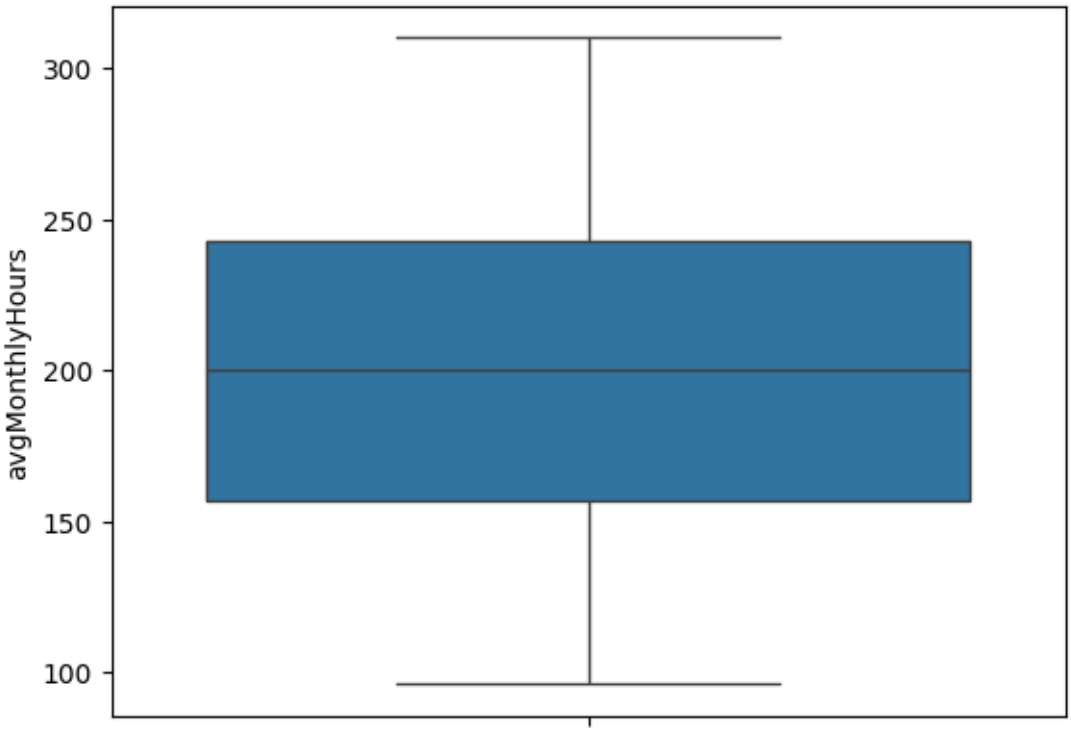
```
plt.title('Box Plot of Age to Identify Outliers')
plt.xlabel('Age')
plt.show()
```

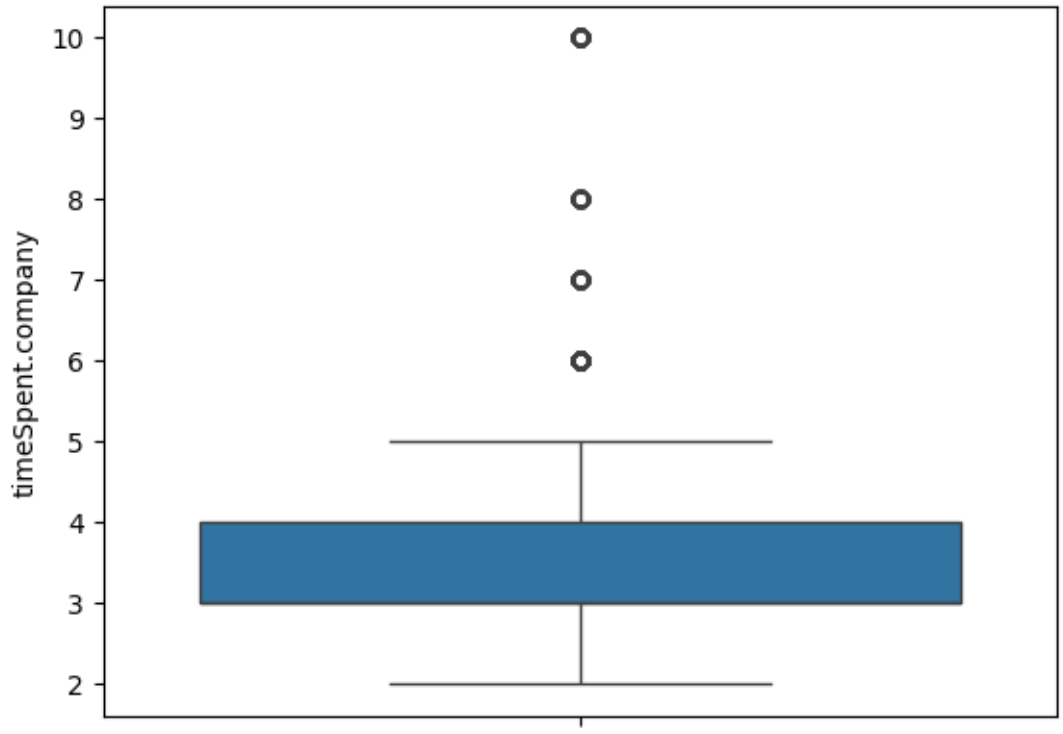
[ ]:
```
for col in df.columns:
  if df[col].dtype !='object':
    sns.boxplot(df[col])
    plt.show()
```

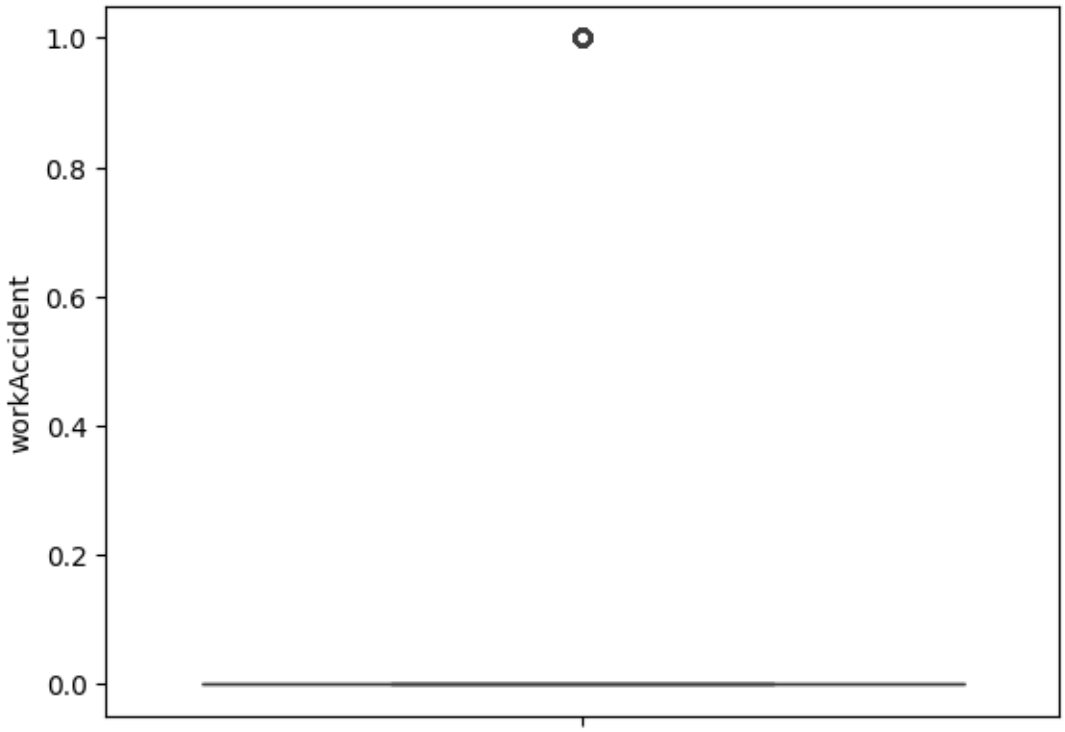


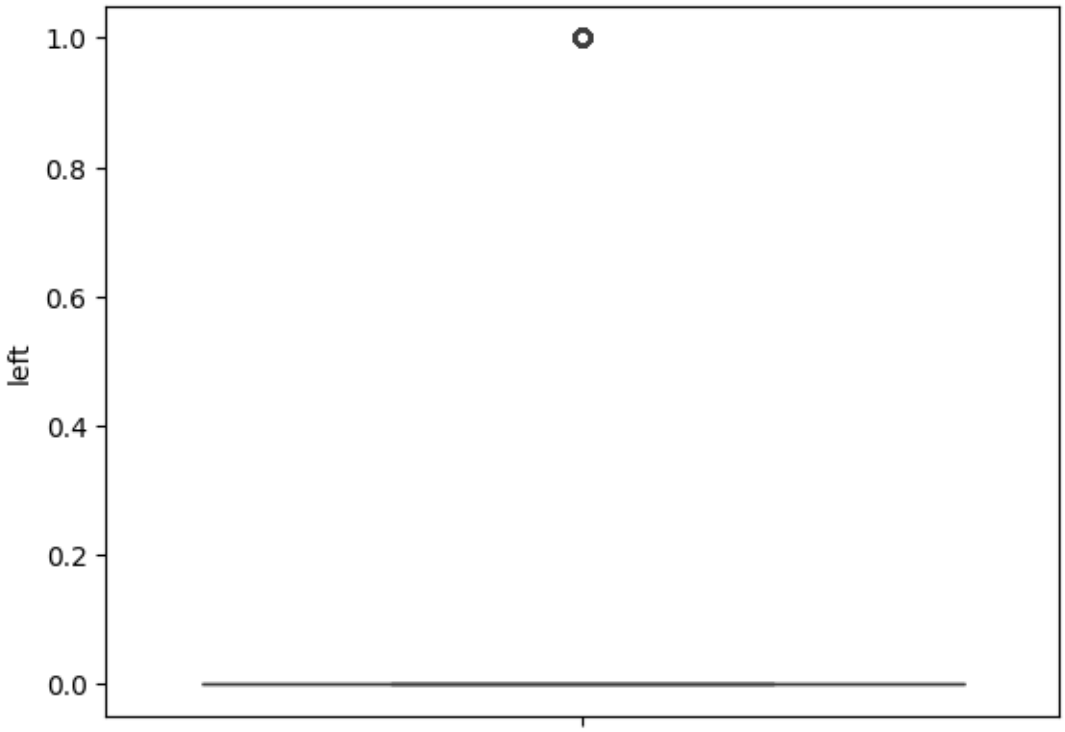

lastEvaluation
1.0
0.9
0.8
0.7
0.6
0.5
0.4

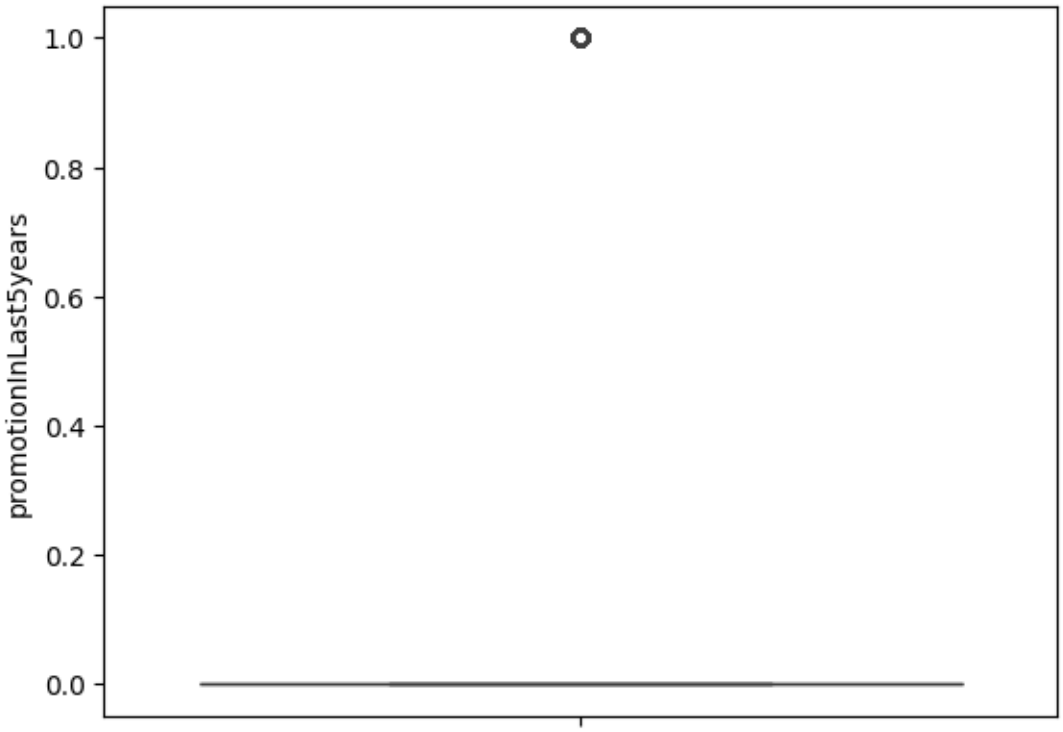

numberOfProjects
7
6
5
4
3
2


avgMonthlyHours
300
250
200
150
100


timeSpent.company
10
9
8
7
6
5
4
3
2


1.0
0.8
0.6
0.4
0.2
0.0
workAccident

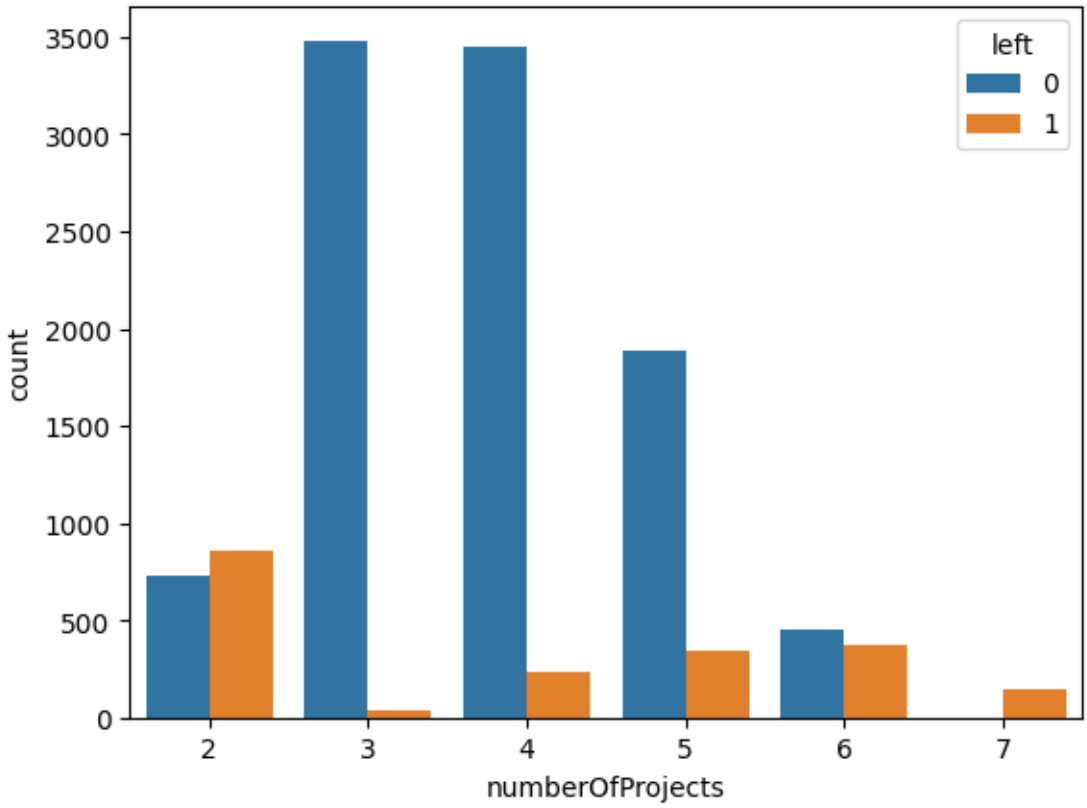

1.0
0.8
0.6
0.4
0.2
0.0
left



[ ]:

is find why people are about the leave the company

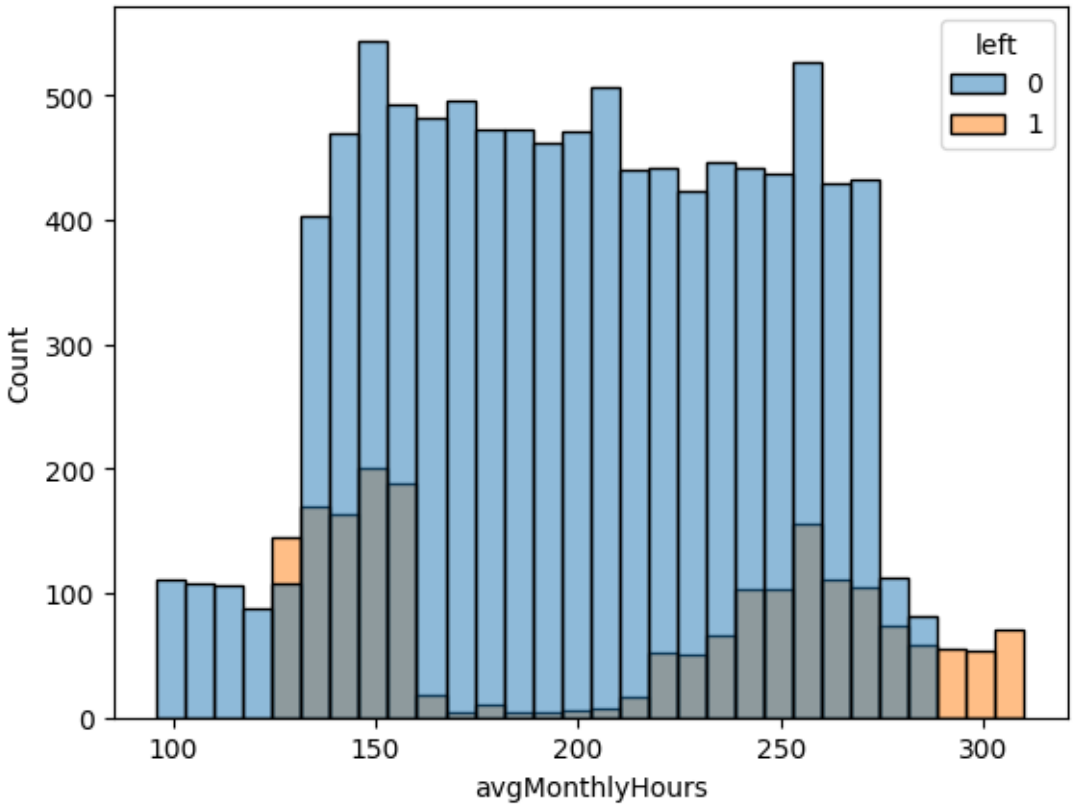
[ ]:
```
sns.countplot(data=df, x='numberOfProjects', hue='left')
plt.show()
```



[ ]:

Its better to give 3-5 projects per head and avoid burden for the employees

[ ]:
```
sns.histplot(data=df,x='avgMonthlyHours', hue='left',bins=30)
plt.show()
```

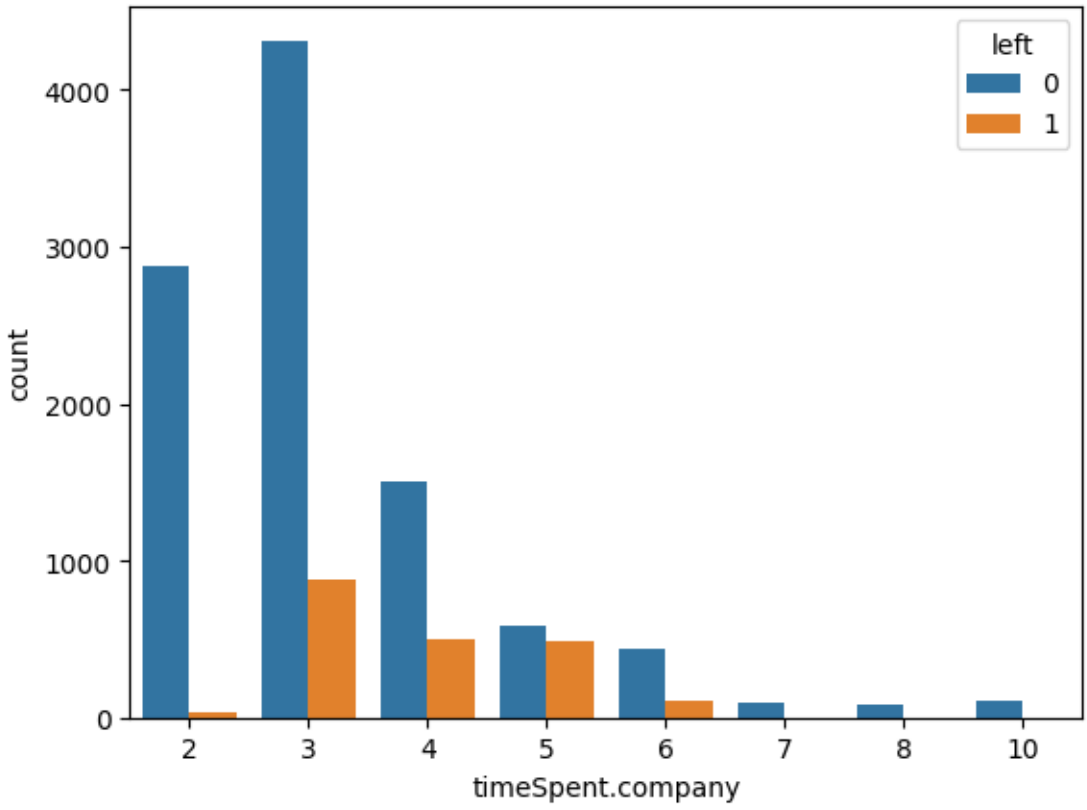


the most preferrable number of monthly working hours on an avg is 100 and 200 #totally avoid 300 because people are leaving that slot

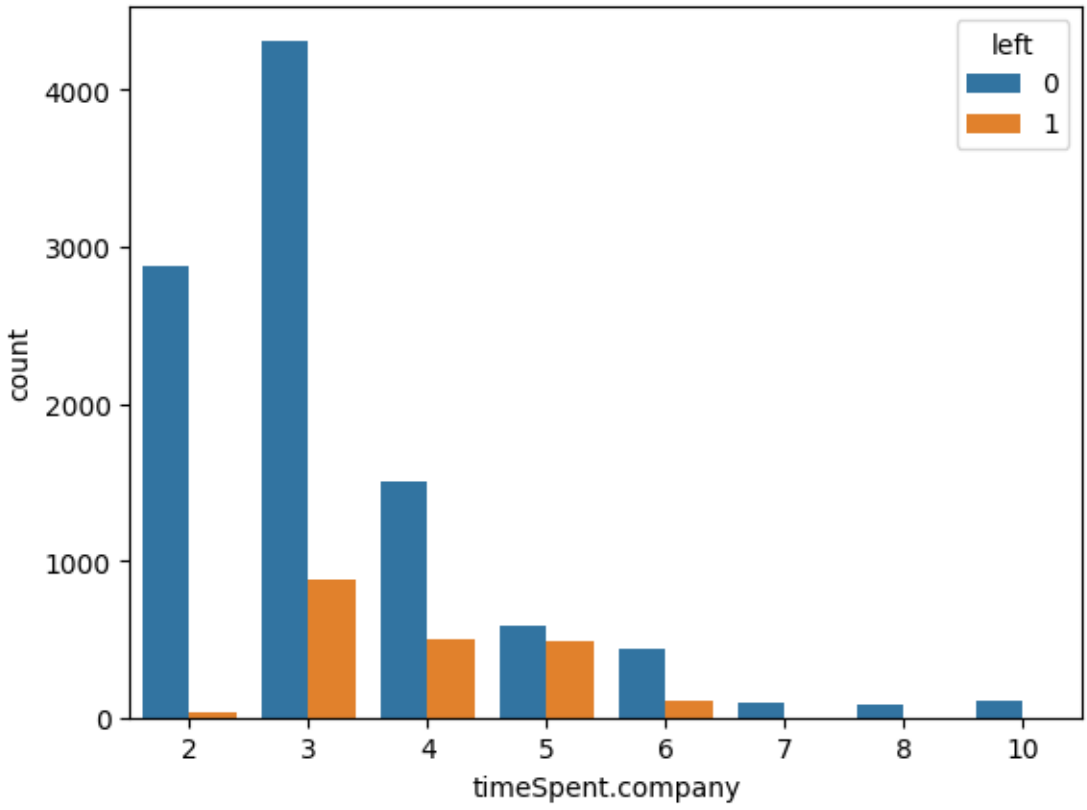
[ ]:
```
df.columns
```

[ ]:
```
Index(['satisfactoryLevel', 'lastEvaluation', 'numberOfProjects',
       'avgMonthlyHours', 'timeSpent.company', 'workAccident', 'left',
       'promotionInLast5years', 'dept', 'salary'],
      dtype='object')
```

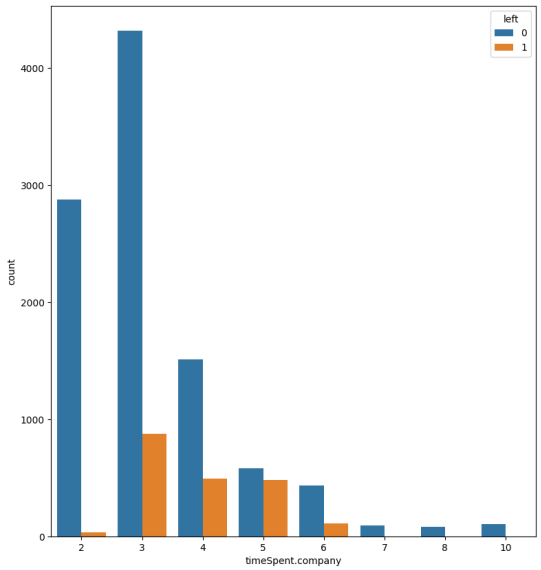
[ ]:
```
sns.countplot(data=df, x='timeSpent.company', hue='left')
plt.show()
```

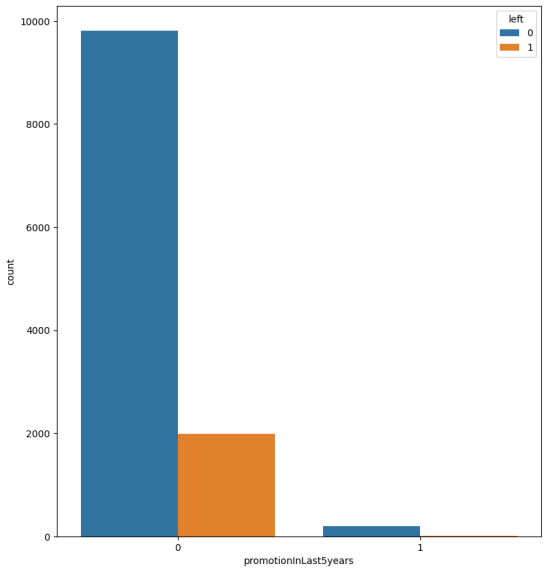


[ ]:
```
sns.countplot(data=df, x='timeSpent.company', hue='left')
plt.show()
```



[ ]:
```
fig=plt.subplots(1,2,figsize=(20,10))
plt.subplot(1,2,1)
sns.countplot(data=df, x='timeSpent.company', hue='left')
plt.subplot(1,2,2)
sns.countplot(data=df, x='promotionInLast5years', hue='left')
plt.show()
```



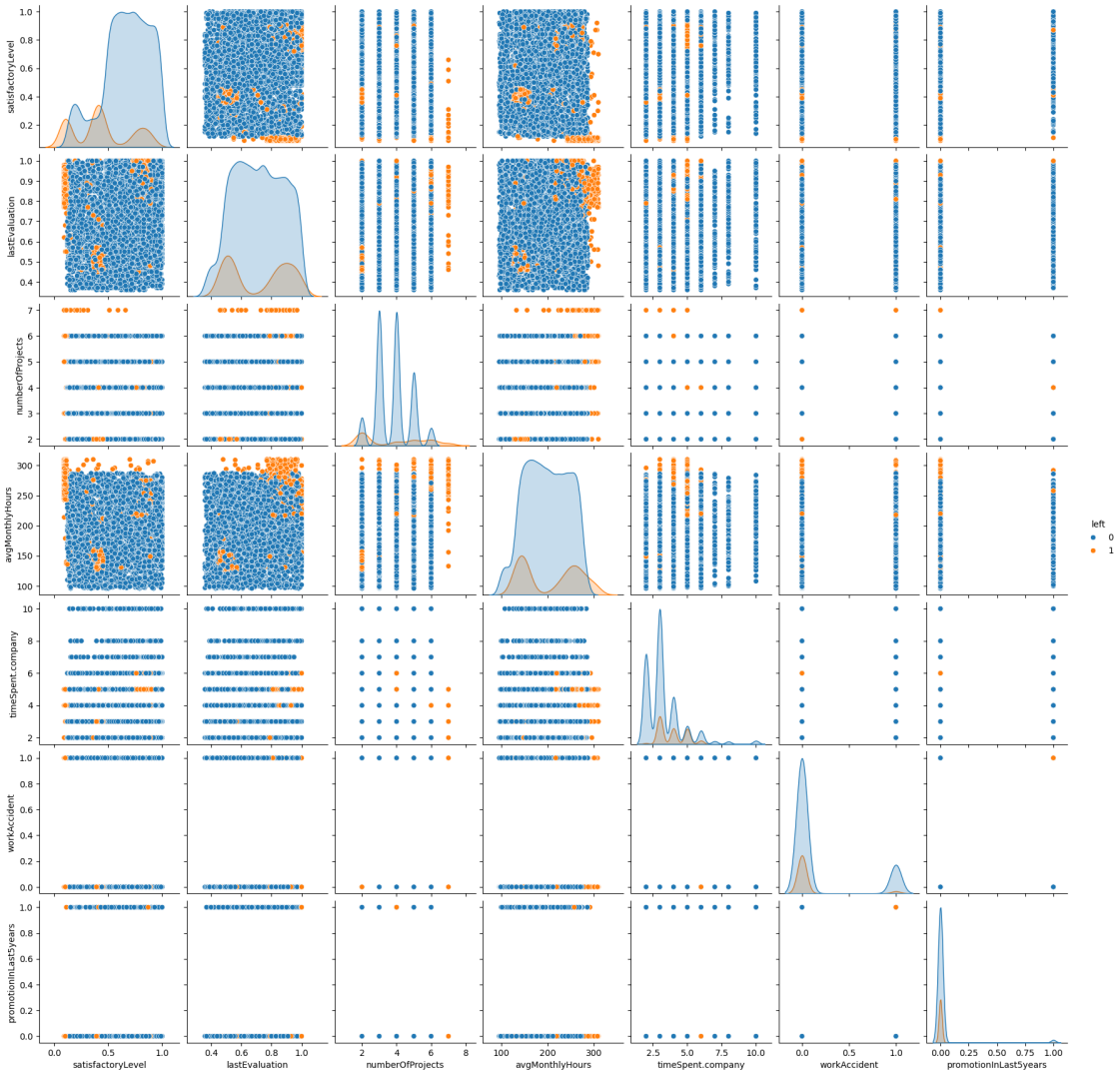


[ ]:

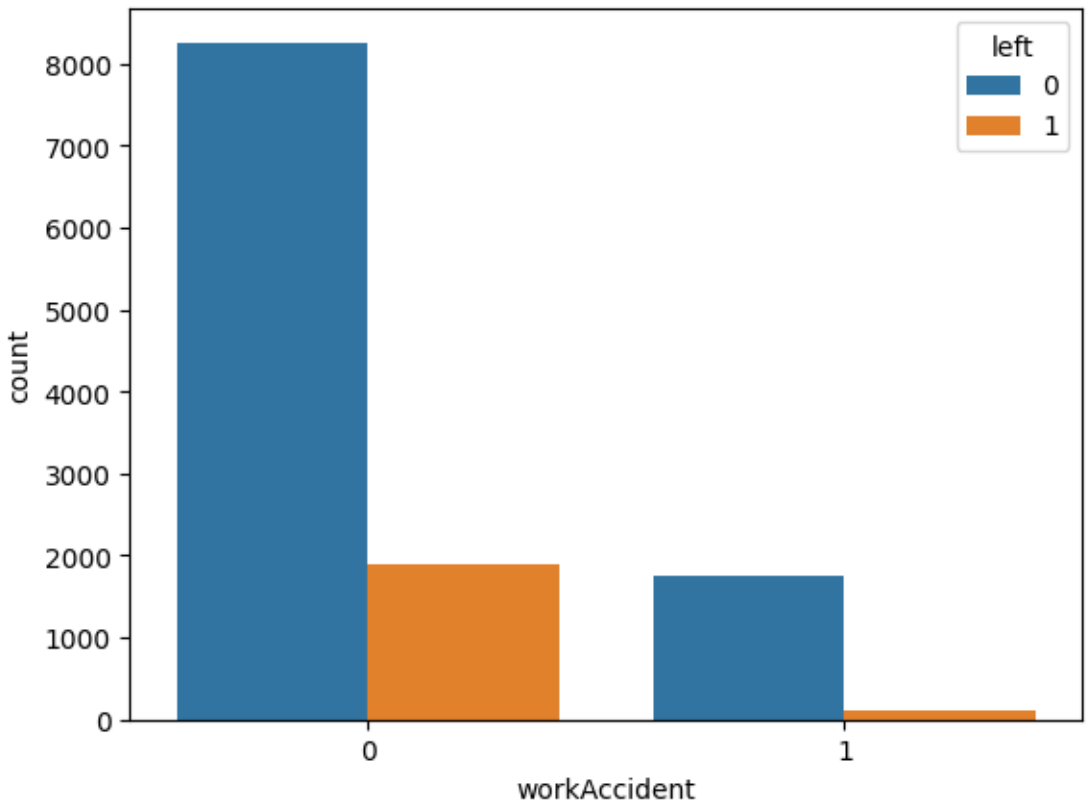
due to lack of proper promotions, employees are tending to the leave the␣ company #Try to give promotions and hikes and better treatment to retain employees␣ after a long period

[ ]:
```
sns.pairplot(df,hue='left')
```

[ ]: <seaborn.axisgrid.PairGrid at 0x7b37671223d0>



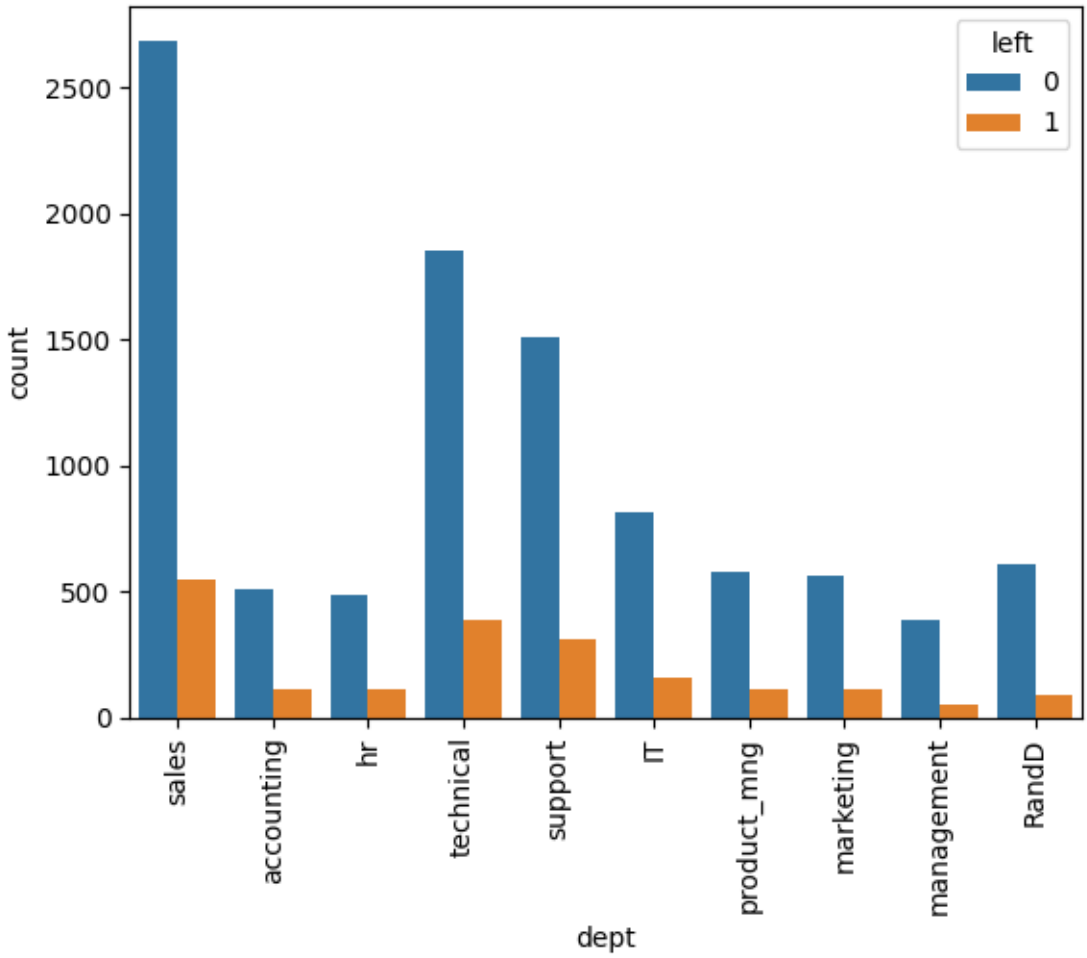
[ ]:
```
sns.countplot(data=df, x='workAccident', hue='left')
plt.show()
```



[ ]:

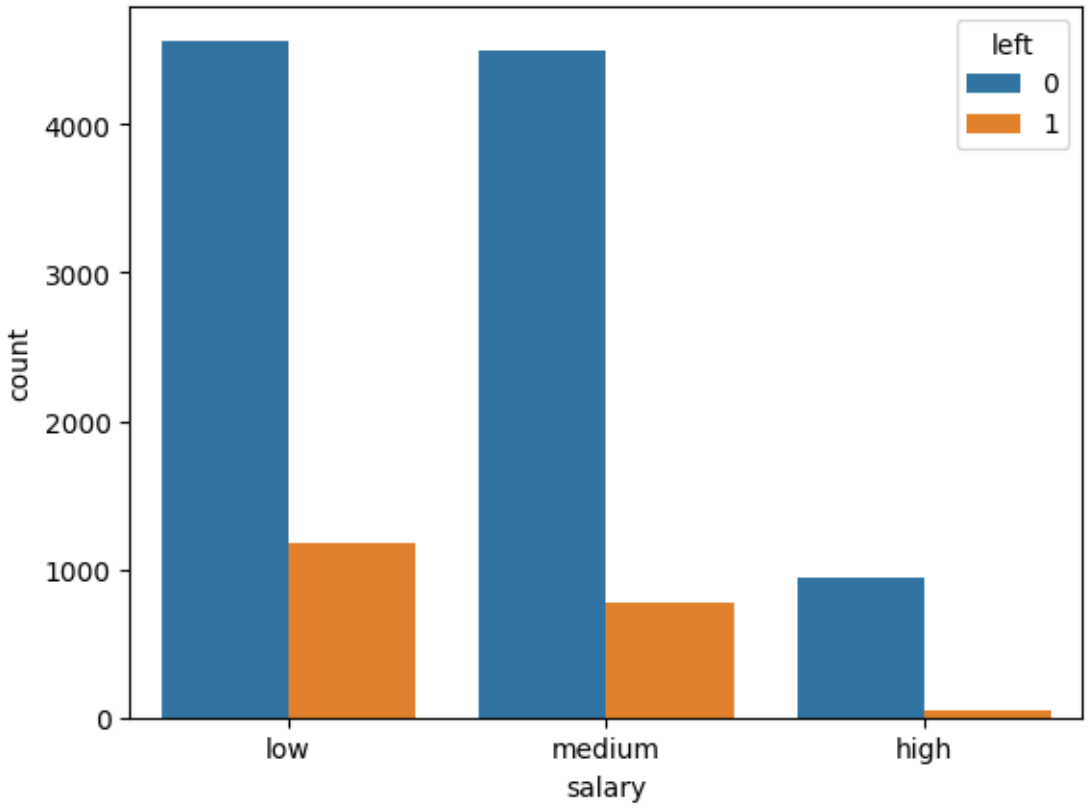
improve the work environment and safety measures

[ ]:
```
sns.countplot(data=df, x='dept', hue='left')
plt.xticks(rotation=90)
plt.show()
```



more employees are present in the department : Sales, Technical and Support so focus on those more

[ ]:
```
sns.countplot(data=df, x='salary', hue='left')
plt.show()
```



Better give appropriate salary for the deserving candidates and people leaving due to low salary

[ ]: