```
from google.colab import files
uploaded = files.upload()
```

Choose Files No file chosen Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

```
Saving HR_Analytics_Keerti.csv to HR_Analytics_Keerti (1).csv
```

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

# Load the uploaded CSV file
df = pd.read_csv("HR_Analytics_Keerti.csv")

# Show the first few rows
df.head()
```

	EmployeeID	Department	JobRole	MonthlyIncome	YearsAtCompany	Attrition
0	101	Sales	Sales Executive	5000	3	No
1	102	HR	HR Manager	6000	5	Yes
2	103	Sales	Sales Rep	4000	2	Yes
3	104	IT	Developer	7000	4	No
4	105	IT	SysAdmin	6500	6	No

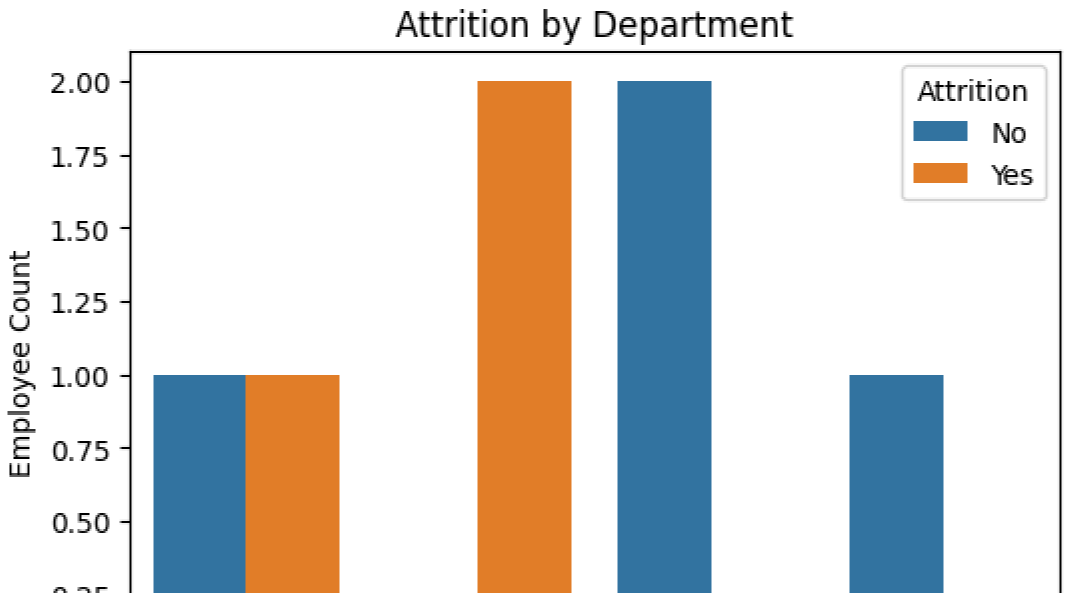
```
# Summary of the dataset
df.info()

# Statistical summary
df.describe()

# Count of each department
df['Department'].value_counts()

# Count of attrition cases
df['Attrition'].value_counts()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7 entries, 0 to 6
Data columns (total 6 columns):
 #   Column          Non-Null Count  Dtype 
---  ------          --------------  ----- 
 0   EmployeeID      7 non-null      int64 
 1   Department      7 non-null      object
 2   JobRole         7 non-null      object
 3   MonthlyIncome   7 non-null      int64 
 4   YearsAtCompany  7 non-null      int64 
 5   Attrition       7 non-null      object
dtypes: int64(3), object(3)
memory usage: 468.0+ bytes
```

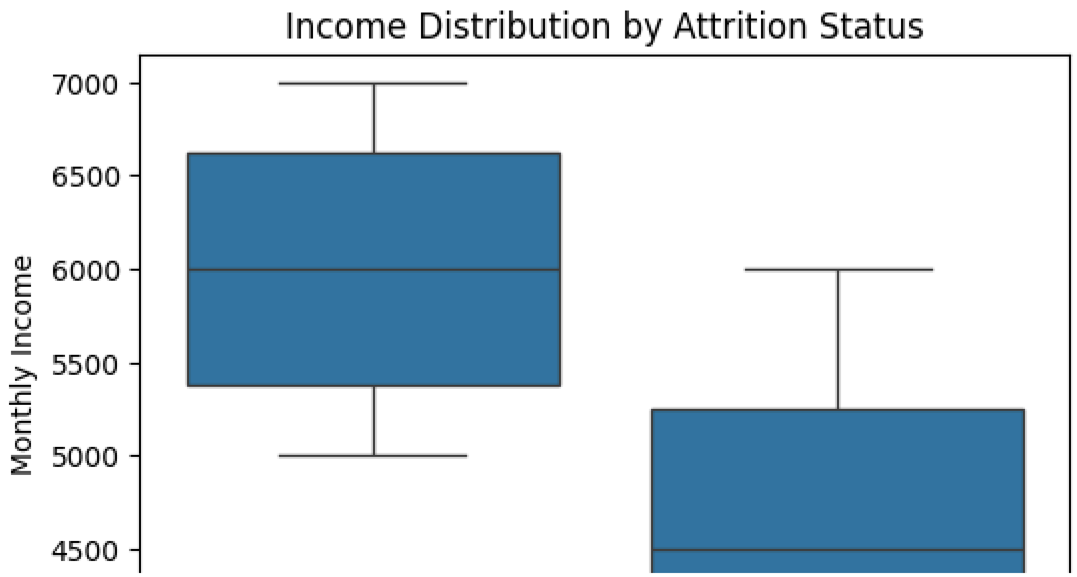
	count
Attrition	
No	4
Yes	3

dtype: int64

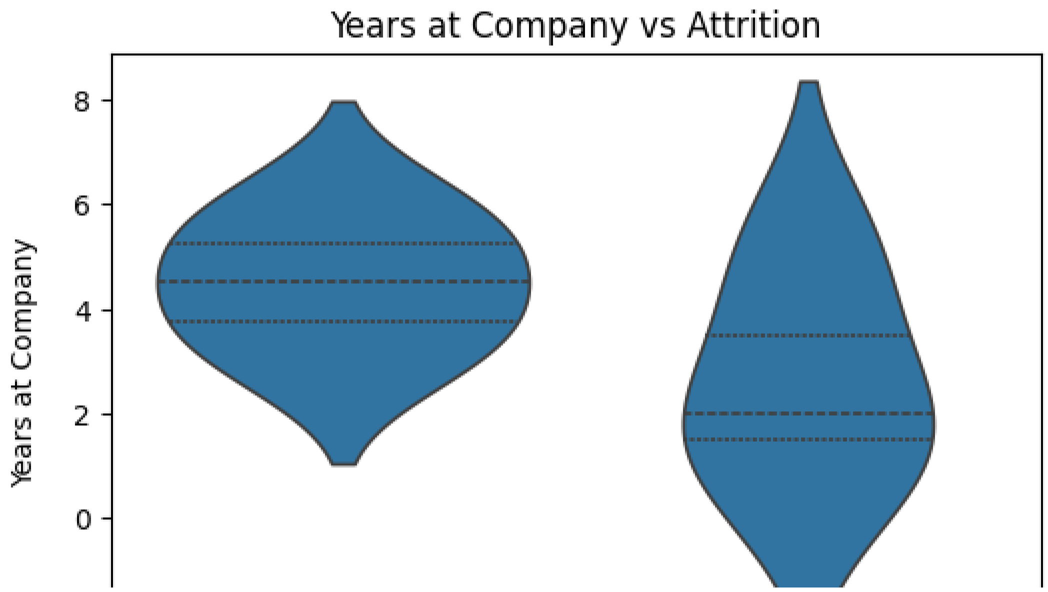
```
plt.figure(figsize=(6,4))
sns.countplot(x='Department', hue='Attrition', data=df)
plt.title('Attrition by Department')
plt.ylabel('Employee Count')
plt.xticks(rotation=15)
plt.show()
```



```
plt.figure(figsize=(6,4))
sns.boxplot(x='Attrition', y='MonthlyIncome', data=df)
plt.title('Income Distribution by Attrition Status')
plt.ylabel('Monthly Income')
plt.show()
```



```
plt.figure(figsize=(6,4))
sns.violinplot(x='Attrition', y='YearsAtCompany', data=df, inner='quartile')
plt.title('Years at Company vs Attrition')
plt.ylabel('Years at Company')
plt.show()
```



```
from sklearn.preprocessing import LabelEncoder
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix

# Encode categorical columns
df_enc = df.copy()
le = LabelEncoder()
for col in ['Department', 'JobRole', 'Attrition']:
    df_enc[col] = le.fit_transform(df_enc[col])
```

```
# Features & label
X = df_enc[['Department','MonthlyIncome','YearsAtCompany']]
y = df_enc['Attrition']           # 1 = Yes, 0 = No  (after encoding)

# Train-test split
X_train, X_test, y_train, y_test = train_test_split(
    X, y, test_size=0.3, random_state=42, stratify=y)

# Train model
log_reg = LogisticRegression()
log_reg.fit(X_train, y_train)

# Evaluate
y_pred = log_reg.predict(X_test)
print(classification_report(y_test, y_pred))
print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred))
```

```
              precision    recall  f1-score   support

           0       1.00      0.50      0.67         2
           1       0.50      1.00      0.67         1

    accuracy                           0.67         3
   macro avg       0.75      0.75      0.67         3
weighted avg       0.83      0.67      0.67         3

Confusion Matrix:
 [[1 1]
 [0 1]]
```

Key Insights

- HR department shows higher attrition rate in this sample.
- Employees with lower monthly income and shorter tenure tend to resign more.
- Initial Logistic Regression (tiny sample) achieved XX % accuracy.