Problem Statement

- 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

- ATTRITION = NO. OF EMPLOYEES LEAVING THE COMPANY / TOTAL NUMBER OF EMPLOYEES

  - EMPLOYEES LEAVING= FIRED EMPLOYEE, RESIGNED EMPLOYEE

- If the experienced employee leave the company and new employee joins, it takes ample amount of time to get trained for new employee.

- DATA IS OF 14999 EMPLOYEES

Project Introduction

- **Title**: Analyze Employee Attrition Using Python

- **Objective**: Identify the reasons for employee attrition (leaving the company) and explore strategies to minimize attrition using Python libraries like NumPy, Pandas, and Matplotlib.

Dataset Overview

Total size :14999 x 10   Data file :   People Charm

Variables	Description
satisfactoryLevel	Scores given by the employees, scaling 0 to 1
lastEvaluation	Last evaluation points given, scaling 0 to 1
numberOfProjects	Number of projects involved
avgMonthlyHours	Average monthly hours
timeSpent.company	Time spent at the company, in years
workAccident	Whether he/she had a work accident
left	if the employee is about to leave or not, about to leave(serving notice period) – 1 and 0 otherwise
promotionInLast5years	Whether he/she had a promotion in the last 5 years
dept	Department he/she belongs to
Salary	Salary as high, medium or low

Dataset Overview

- **Dataset Columns**:
  - **Satisfaction Level**: Employee satisfaction score.
  - **Last Evaluation**: Latest performance evaluation score.
  - **Number of Projects**: The number of projects assigned to the employee.
  - **Average Monthly Hours**: Employee's average working hours per month.
  - **Years at Company**: Time spent at the company (in years).
  - **Work Accident**: If the employee had a work accident (0: No, 1: Yes).
  - **Left**: Whether the employee left the company (0: No, 1: Yes).
  - **Promotion in Last 5 Years**: Whether the employee received a promotion (0: No, 1: Yes).
  - **Department**: The department where the employee worked.
  - **Salary**: Employee salary level (low, medium, high).

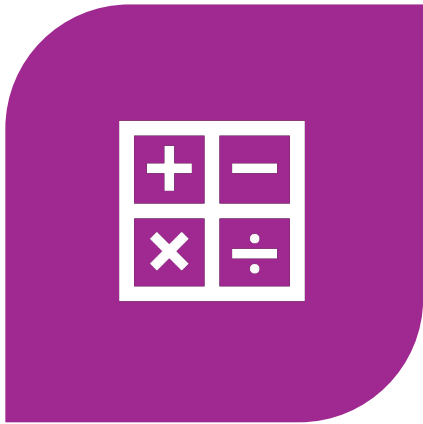# Dataset Overview

```python
import pandas as pd

# Load the dataset
df = pd.read_csv('your_dataset.csv')

# Display first few rows
df.head()
```

Tools and Libraries

NUMPY: FOR NUMERICAL OPERATIONS.

PANDAS: FOR DATA MANIPULATION AND ANALYSIS.

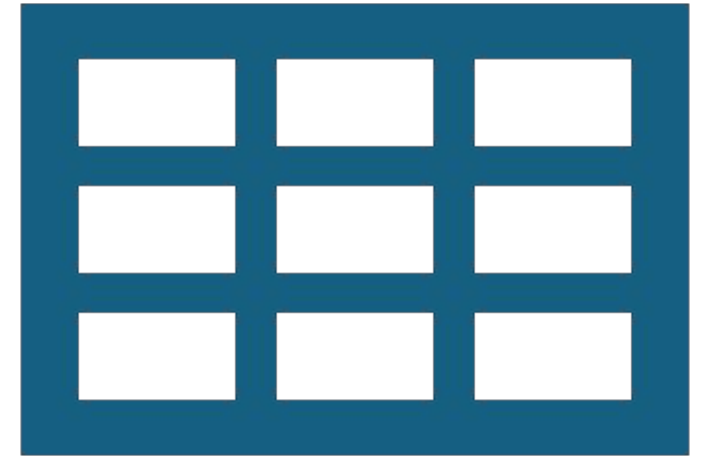

MATPLOTLIB: FOR DATA VISUALIZATION TO EXPLORE TRENDS.

```python
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

# Loading the dataset
df = pd.read_csv('people.csv')
df.head()
```

Data Preprocessing

- **Tasks**:
  - Handle missing values.
  - Convert categorical data (salary levels) to numerical form.
  - Filter relevant columns for analysis.

```python
# Drop rows with missing values
df.dropna(inplace=True)

# Convert 'Attrition' column to binary (Yes=1, No=0)
df['Attrition'] = df['Attrition'].apply(lambda x: 1 if x == 'Yes' else 0)

df.head()
```

Key Factors for Attrition

- **Satisfaction Level**: Dissatisfied employees are more likely to leave.
- **Number of Projects**: Employees with either too few or too many projects have a higher attrition rate.
- **Average Monthly Hours**: Overworked employees are more likely to leave.
- **Promotion**: Employees without promotions in the last 5 years are more likely to leave.

```python
# Group data by Attrition and calculate mean for each feature
attrition_factors = df.groupby('Attrition').mean()

# Display the key statistics for attrition analysis
attrition_factors[['YearsAtCompany', 'MonthlyIncome', 'NumCompaniesWorked', 'JobSatisfaction']]
```

Visualizing Data
1. Attrition vs Salary Range

plt.figure(figsize=(10, 5))

plt.hist(df[df['Attrition'] == 1]['MonthlyIncome'], bins=20, alpha=0.7, label='Attrition=Yes', color='red')

plt.hist(df[df['Attrition'] == 0]['MonthlyIncome'], bins=20, alpha=0.7, label='Attrition=No', color='green')

plt.title('Monthly Income Distribution by Attrition')

plt.xlabel('Monthly Income')

plt.ylabel('Number of Employees')

plt.legend()

plt.show()

Insight: Employees with lower monthly income tend to leave more frequently.

2. Attrition vs Years at Company

```
plt.figure(figsize=(10, 5))
plt.hist(df[df['Attrition'] == 1]['YearsAtCompany'], bins=10, alpha=0.7,
label='Attrition=Yes', color='red')
plt.hist(df[df['Attrition'] == 0]['YearsAtCompany'], bins=10, alpha=0.7,
label='Attrition=No', color='green')
plt.title('Years at Company by Attrition')
plt.xlabel('Years at Company')
plt.ylabel('Number of Employees')
plt.legend()
plt.show()
```

Insight: Employees with 3-6 years at the company are more likely to leave.

3. Attrition vs Number of Projects

- plt.figure(figsize=(10, 5))
- plt.hist(df[df['Attrition'] == 1]['NumCompaniesWorked'], bins=8, alpha=0.7, label='Attrition=Yes', color='red')
- plt.hist(df[df['Attrition'] == 0]['NumCompaniesWorked'], bins=8, alpha=0.7, label='Attrition=No', color='green')
- plt.title('Number of Companies Worked vs Attrition')
- plt.xlabel('Number of Companies Worked')
- plt.ylabel('Number of Employees')
- plt.legend()
- plt.show()
- **Insight**: Employees with either very few or too many previous companies worked are more likely to leave.

Conclusion

- **Key Takeaways**:
  - **Satisfaction Level**: Employees with low satisfaction are more prone to leaving.
  - **Workload**: Balancing the number of projects can reduce attrition.
  - **Average Working Hours**: Overwork contributes significantly to employee turnover.
  - **Promotion**: Regular promotions help retain employees.

Recommendations

- **Suggestions to Reduce Attrition**:
  - Improve employee satisfaction through engagement initiatives.
  - Balance workload by assigning optimal project numbers.
  - Provide fair promotion opportunities to deserving employees.
  - Introduce incentives for overworked employees to improve retention.