

HR Attrition Analysis & Dashboard

(Interactive Analysis using Microsoft Excel, Python & Power BI)

PROJECT OVERVIEW

Employee attrition—the departure of employees from an organization—poses significant challenges for businesses, impacting productivity, morale, and financial performance. Understanding the underlying factors contributing to employee turnover is essential for developing effective retention strategies. The HR Attrition Data Analysis Project at Unified Mentor Pvt Ltd was initiated to delve into these factors, utilizing data-driven methodologies to uncover insights and inform decision-making.

The primary objective of this project was to analyze historical employee data to identify patterns and predictors of attrition within the organization. By leveraging statistical analysis and machine learning techniques, the project aimed to provide actionable insights that could aid in reducing turnover rates and enhancing employee satisfaction.

Project Purpose

The primary purpose of the project was to unlock actionable intelligence from unstructured data. By systematically converting raw data into structured insights, the project aimed to enhance the decision-making process within the organization. The initiative was designed to address key business challenges by revealing hidden patterns and operational inefficiencies.

- **Data Collection and Integration:** Gather and consolidate employee data from various sources, including demographics, job roles, performance metrics, and exit interviews, to create a comprehensive dataset for analysis.
- **Exploratory Data Analysis (EDA):** Conduct thorough examinations of the dataset to identify patterns, trends, and correlations related to employee attrition.
- **Predictive Modeling:** Develop machine learning models to predict the likelihood of employee departures, enabling proactive retention strategies.

- **Identification of Key Attrition Factors:** Analyze the impact of variables such as job satisfaction, compensation, work-life balance, and career development opportunities on employee turnover.
- **Visualization and Reporting:** Create intuitive visualizations and dashboards to effectively communicate findings to stakeholders, facilitating data-driven decision-making.
- **Strategic Recommendations:** Provide actionable insights and recommendations to HR departments aimed at reducing attrition rates and enhancing employee engagement and satisfaction.

Objective

- **Technical Mastery:** To develop advanced proficiency in data collection, cleaning, analysis, and visualization using state-of-the-art tools.
- **Analytical Acumen:** To cultivate the ability to interpret complex datasets, recognize meaningful patterns, and forecast future trends.
- **Strategic Insight:** To generate data-backed recommendations that inform and enhance business strategies.
- **Professional Growth:** To integrate academic knowledge with practical industry experience, thereby preparing for a career in data science.

Dataset

Source: HR Attrition Data (CSV format)

Dataset Link: [HR_Attrition_Dataset](#)

Records: 1471 rows

Columns: Age, Attrition, BusinessTravel, DailyRate, Department, DistanceFromHome, Education, EducationField, EmployeeCount, EmployeeNumber, EnvironmentSatisfaction, Gender, HourlyRate, JobInvolvement, JobLevel, JobRole, JobSatisfaction, MaritalStatus, MonthlyIncome, MonthlyRate, NumCompaniesWorked, Over18, OverTime, PercentSalaryHike, PerformanceRating, RelationshipSatisfaction, StandardHours, StockOptionLevel, TotalWorkingYears, TrainingTimesLastYear, WorkLifeBalance, YearsAtCompany, YearsInCurrentRole, YearsSinceLastPromotion, YearsWithCurrManager, Attrition_Code

Data Processing

Data Cleaning & Transformation

- Standardized Department (Human Resources, Research & Development, Sales).
- Added new column Attrition Code for sum analysis.
- Handled missing values & ensured consistency in categorical fields.

Key Performance Indicators (KPIs)

- Total Employee – Total Count Employees.
- Leave Employee – Count of Leaved Employee.
- Average Monthly Income – Count of Average Monthly Income per Employee.
- Average Age – Count of Average Age.

1. Microsoft Excel

Although more advanced tools formed the backbone of the project, Microsoft Excel was essential for preliminary data exploration and quick computations. Excel's user-friendly interface enabled rapid data cleaning, the creation of pivot tables, and fundamental statistical analysis. This tool served as an accessible starting point for understanding the data before transitioning to more complex analytical environments.

Action Perform: Data Cleaning Techniques, Data Formatting, Pivot Tables

Microsoft Excel is a versatile tool for data analysis, enabling users to efficiently collect, organize, and interpret data. You can start by entering or importing data, then use functions and tools to clean and standardize it.



- **Data Cleaning Techniques:** Developed strategies to remove duplicates, correct errors, and standardize data, ensuring high-quality and accurate datasets for analysis.
- **Data Formatting:** Learned techniques to properly structure and present data, including cell formatting, conditional formatting, and alignment to ensure clarity.
- **Pivot Tables:** Explored methods to summarize large datasets quickly, creating dynamic pivot tables for efficient data analysis.

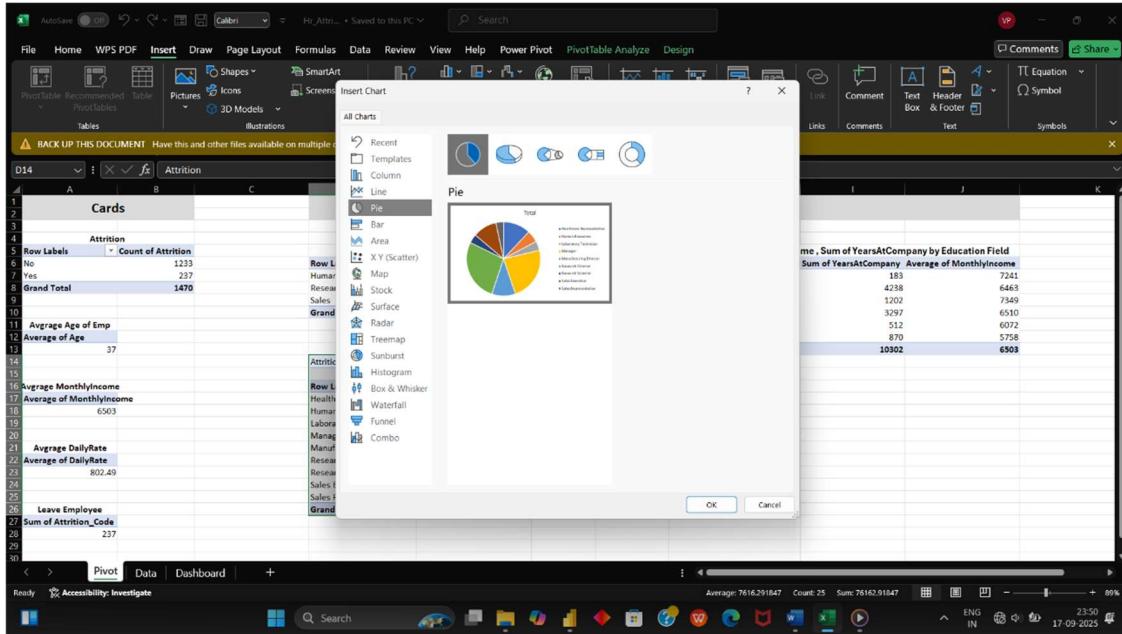
✍ Data Cleaning Images:

The screenshot shows a WPS Office spreadsheet with a 'Remove Duplicates' dialog box open. The dialog box has a title 'Remove Duplicates' and a message 'To delete duplicate values, select one or more columns that contain duplicates.' with a 'Select All' checkbox checked. Below this are checkboxes for 'Age', 'Attrition', 'BusinessTravel', 'DailyRate', and 'Department'. At the bottom are 'OK' and 'Cancel' buttons.

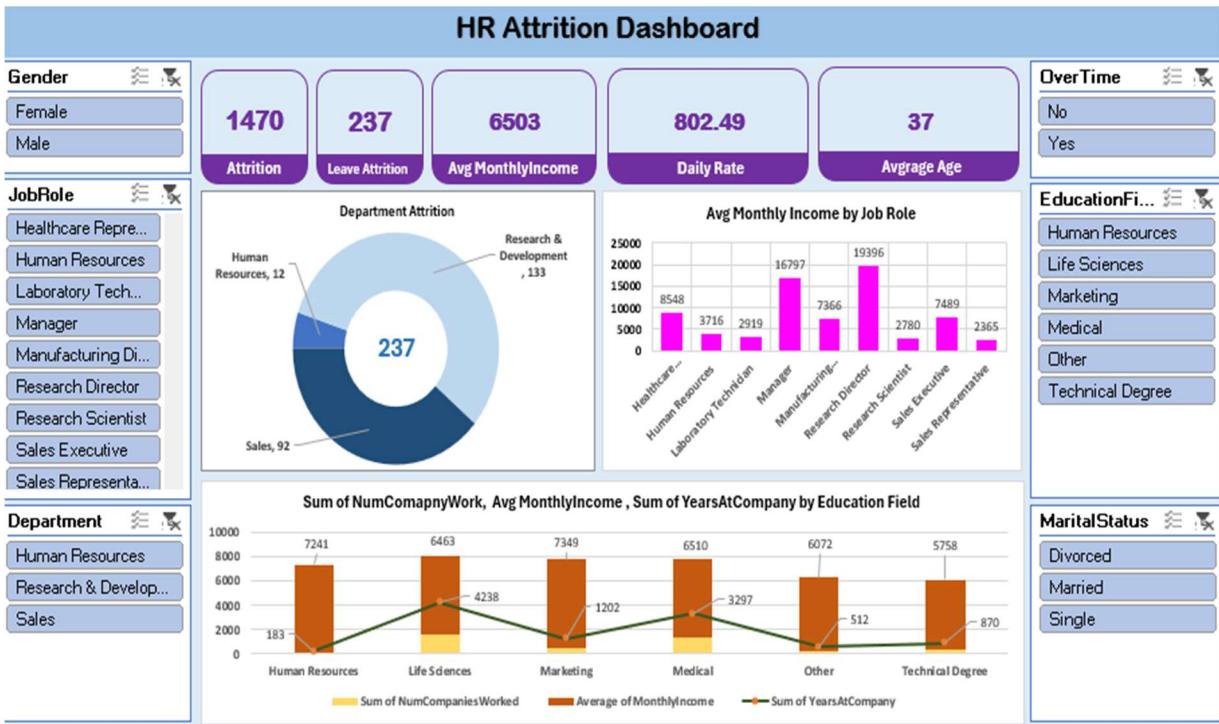
✍ Create Pivot Calculations Images:

The screenshot shows a WPS Office spreadsheet with a PivotTable setup. The PivotTable is divided into 'Cards' and 'Graphs' sections. The 'Cards' section contains summary data like 'Count of Attrition' (1476), 'Average Age of Emp' (37), and 'Average MonthlyIncome' (6503). The 'Graphs' section contains four charts: 'Sum of NumCompanyWork', 'Avg MonthlyIncome', 'Sum of YearsAtCompany', and 'Average of MonthlyIncome'. The 'Sum of NumCompanyWork' chart shows Human Resources (88), Life Sciences (1621), Manufacturing (467), Medical (1292), Other (230), and Technical Degree (341). The 'Avg MonthlyIncome' chart shows Healthcare Representative (8548), Human Resources (3716), Laboratory Technician (2919), Manager (16797), Manufacturing Director (7366), Research Director (19396), Research Scientist (7390), Sales Executive (7489), and Sales Representative (2365). The 'Sum of YearsAtCompany' chart shows the same categories with their respective average years. The 'Average of MonthlyIncome' chart shows the same categories with their respective average monthly income.

✍ Create Visualization Images:



✍ Excel HR Attrition Dashboard:



2. Python (Jupyter Notebook)

Libraries Used: Pandas, Matplotlib, Seaborn, Plotly.

🔗 Python Notebook Link: [HR_Attrition.ipynb](#)

Steps:

- Data loading & cleaning with Pandas.
- EDA (Exploratory Data Analysis) with advanced visualizations.
- Interactive plots with Plotly (time series, category analysis).

↗ Data Cleaning & Transformation Images:

```
[1]: df.head()
[2]: 
[3]: Age Attrition BusinessTravel DailyRate Department DistanceFromHome Education EducationField EmployeeCount EmployeeNumber ... RelationshipSatisfaction
0 41 Yes Travel_Rarely 1102 Sales 1 2 Life Sciences 1 1 ...
1 49 No Travel_Frequently 279 Research & Development 6 1 Life Sciences 1 2 ...
2 37 Yes Travel_Rarely 1373 Research & Development 2 2 Other 1 4 ...
3 33 No Travel_Frequently 1392 Research & Development 3 4 Life Sciences 1 5 ...
4 27 No Travel_Rarely 591 Research & Development 2 1 Medical 1 7 ...

5 rows × 35 columns

[4]: sns.set_style('whitegrid')
[5]: df.shape
[6]: (1470, 95)
[7]: df.columns
[8]: Index(['Age', 'Attrition', 'BusinessTravel', 'DailyRate', 'Department', 'DistanceFromHome', 'Education', 'EducationField', 'EmployeeCount', 'EmployeeNumber', 'EnvironmentSatisfaction', 'HourlyRate', 'JobInvolvement', 'JobLevel', 'JobRole', 'JobSatisfaction', 'MaritalStatus', 'MonthlyIncome', 'NumCompaniesWorked', 'OverTime', 'PercentSalaryHike', 'PerformanceRating', 'RelationshipSatisfaction', 'StockOptionLevel', 'TotalWorkingYears', 'TrainingTimesLastYear', 'WorkLifeBalance', 'YearsAtCompany', 'YearsInCurrentRole', 'YearsSinceLastPromotion', 'YearsWithCurrManager'], dtype='object')
[9]: df.describe()
[10]: 
[11]: Age DailyRate DistanceFromHome Education EmployeeCount EmployeeNumber EnvironmentSatisfaction HourlyRate JobInvolvement JobLevel
count 1470.000000 1470.000000 1470.000000 1470.0 1470.000000 1470.000000 1470.000000 1470.000000 1470.000000 1470.000000
mean 36.923811 802.485714 9.192517 2.912925 1.0 1024.865306 2.721769 65.891156 2.729932 2.0639
std 9.135373 403.509100 8.106864 1.024165 0.0 602.024335 1.093082 20.329428 0.711561 1.1069
min 18.000000 100.000000 1.000000 1.000000 1.000000 1.000000 1.000000 1.000000 1.000000 1.0000
25% 30.000000 463.000000 2.000000 2.000000 1.0 491.250000 2.000000 48.000000 2.000000 1.0000
50% 36.000000 802.000000 7.000000 3.000000 1.0 1020.500000 3.000000 66.000000 3.000000 2.0000
75% 43.000000 1157.000000 14.000000 4.000000 1.0 1555.750000 4.000000 83.750000 3.000000 3.0000
```

```
[1]: yearSinceLastPromotion Attrition
dtype: object
[2]: 
[3]: # drop unnecessary column
df = df.drop(['EmployeeCount', 'OverTime', 'StandardHours'], axis=1)
[4]: df.isnull().sum()
[5]: 
[6]: Age Attrition BusinessTravel DailyRate Department DistanceFromHome Education EducationField EmployeeCount EmployeeNumber ... RelationshipSatisfaction
0 41 Yes Travel_Rarely 1102 Sales 1 2 Life Sciences 1 1 ...
1 49 No Travel_Frequently 279 Research & Development 6 1 Life Sciences 1 2 ...
2 37 Yes Travel_Rarely 1373 Research & Development 2 2 Other 1 4 ...
3 33 No Travel_Frequently 1392 Research & Development 3 4 Life Sciences 1 5 ...
4 27 No Travel_Rarely 591 Research & Development 2 1 Medical 1 7 ...

[7]: df.duplicated().sum()
[8]: 0
[9]: 
[10]: # add Attrition column
df['Attrition_Code'] = df['Attrition'].map({'Yes':1, 'No':0})
[11]: print(df[['Attrition', 'Attrition_Code']].head())
Attrition Attrition_Code
0 Yes 1
1 No 0
2 Yes 1
3 No 0
4 No 0
```

4 Data Visualization Images:

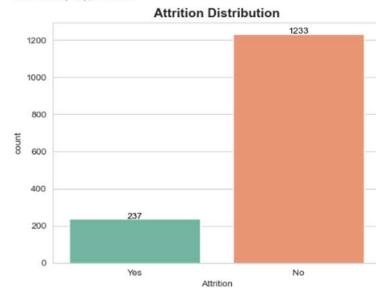
- Bar Chart by Attrition Count:

Attrition Count

```
[13]: # Check Attrition distribution
print(df['Attrition'].value_counts())
ax = sns.countplot(x='Attrition', data=df, palette='Set2')
plt.title('Attrition Distribution', fontsize=14, fontweight='bold')
```

```
#dots! Label
for p in ax.patches:
    height = p.get_height()
    ax.text(p.get_x() + p.get_width()/2., height + 2,
            int(height), ha='center', color='black')
plt.show()
```

```
No 1233
Yes 237
Name: count, dtype: int64
```



- Histogram for Attrition by Age:

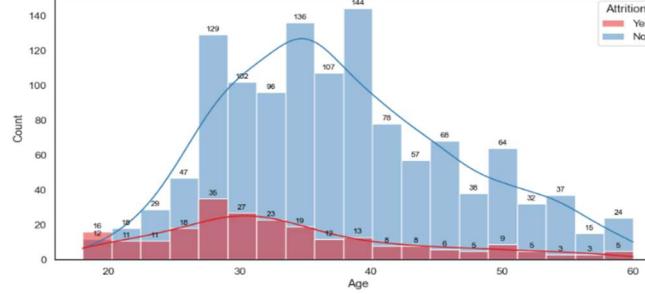
Histogram: Attrition by Age

```
[49]: # Attrition vs Age
plt.figure(figsize=(12,6))
ax1 = sns.histplot(data=df, x='Age', hue='Attrition', kde=True, palette='Set1')
plt.title('Attrition by Age', fontsize=14, fontweight='bold')
```

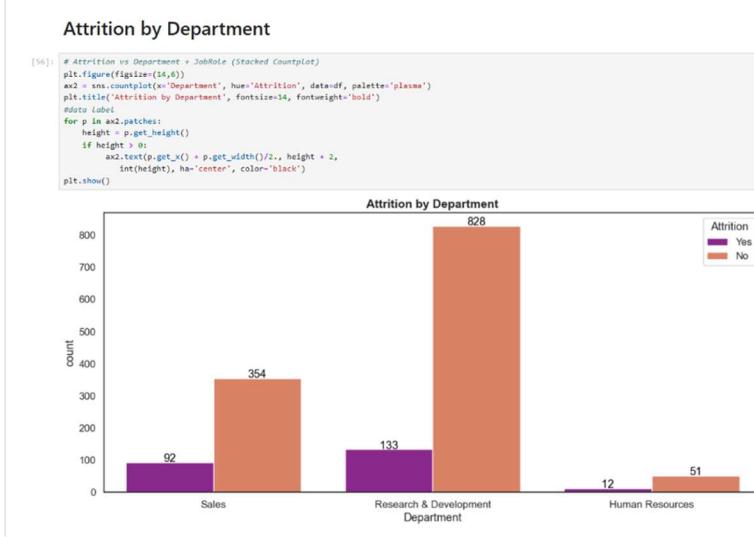
```
#dots! Label
for p in ax1.patches:
    height = p.get_height()
    if height > 0:
        ax1.text(p.get_x() + p.get_width()/2., height + 2,
                int(height), ha='center', fontsize=9, color='black')
plt.show()
```

```
C:\Users\ashish\anaconda3\lib\site-packages\seaborn\kde.py:1119: FutureWarning: use_inf_as_na option is deprecated and will be removed in a future
version. Convert inf values to NaN before operating instead.
    with np.seterr_context(more_use_inf_as_na=True):
```

Attrition by Age



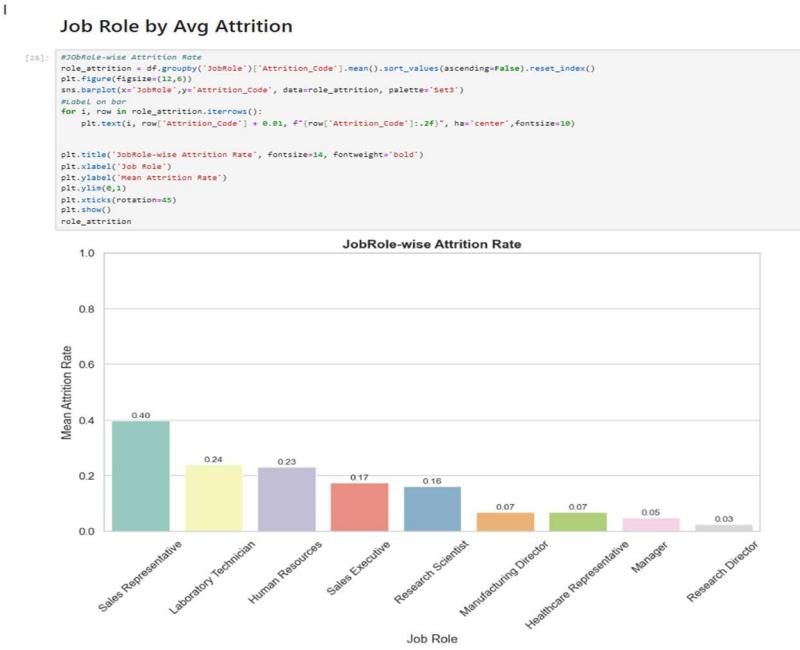
- Bar Chart for Attrition by Age:



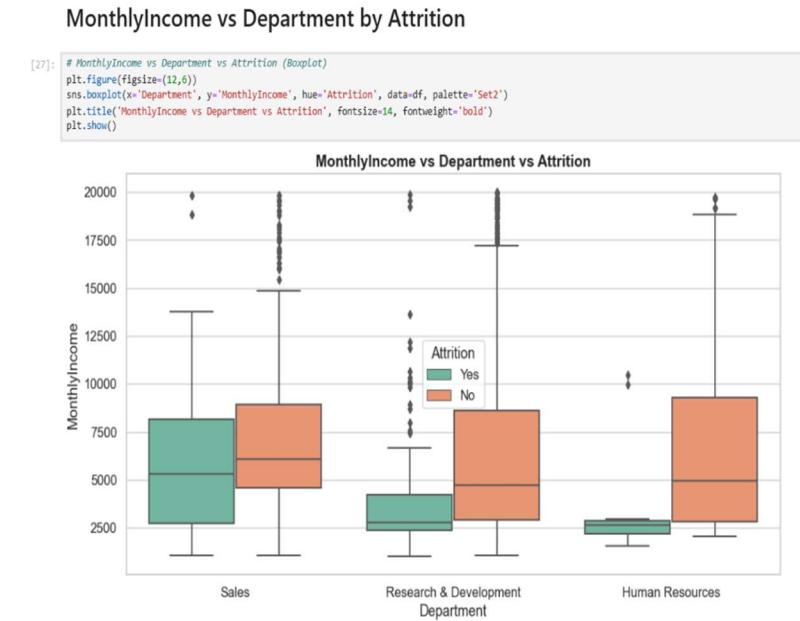
- Stacked Chart for Overtime by Attrition %:



- Bar Chart for Job Role by Avg Attrition:



- Boxplot for MonthlyIncome vs Department by Attrition:



- KPI Calculations: Total Employee, Leave Employee, Avg Monthly Income, Avg Age Employee:

KPI Calculations: Total Employee, Leave Employee, Avg Monthly Income, Avg Age Employee

```
[61]: TotalEmployees = df['EmployeeNumber'].value_counts().sum()
LeavedEmployee = df['Attrition_Code'].sum()
AverageMonthlyIncome = df['MonthlyIncome'].mean().round(2)
AvgrageAgeEmp = df['Age'].mean().round(2)

kpis_title = ['TotalEmployees', 'LeavedEmployee', 'AverageMonthlyIncome', 'AvgrageAgeEmp']
kpis = [
    f'{Employees:,}f",
    f'{EmployeesLeave:,}',
    f'{AverageMonthlyIncome}',
    f'{AvgrageAgeEmp}'
]
sns.set_style('white')
fig, axes = plt.subplots(1, 4, figsize=(20,3))
colors=['#2ecc71', '#3498db', '#e67e22', '#9b59b6']

for ax, title, value, color in zip(axes, kpis_title, kpis, colors):
    ax.set_facecolor(color)
    ax.text(0.5, 0.55, value, ha='center', va='center', fontsize=20, color='white', fontweight='bold')
    ax.text(0.5, 0.25, title, ha='center', va='center', fontsize=20, color='white', fontweight='bold')
    ax.set_xticks([])
    ax.set_yticks([])
    for spine in ax.spines.values():
        spine.set_visible(False)

plt.suptitle('KPI', fontsize=18, fontweight='bold')
plt.subplots_adjust(top=0.8, wspace=0.3)
plt.show()
```

KPI

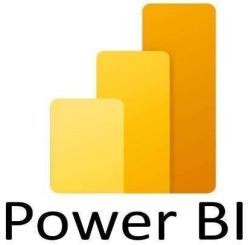


3. Power BI Dashboard

Power BI is a Microsoft tool for business analytics that helps users visualize and share data insights. It integrates with various data sources, offers diverse visualization options, and supports data modelling and transformation. Power BI enables interactive reports and dashboards, facilitates collaboration and sharing, incorporates AI for advanced insights, and provides mobile access for on-the-go data analysis.

↙ Features:

- Interactive slicers (Gender, Department, Job Role, Education Field).
- KPI Cards (Total Employee, Leave Employee, Avg Monthly Income, Avg Age Employee:).
- Dynamic bar/Pie charts for Department/Gender wise Attrition.
- Satisfaction preference insights by Jobrole & Environment.



↙ Insights:

- Research & Development Department are shows 133 Employees are exit.
- Employees who work overtime (overtime=Yes) have a higher percentage of attrition compared to those who don't.
- In Department where WorkLifeBalance is poor, attrition percentage is noticeably higher.

DAX Functions:

- Implemented Data Analysis Expressions (DAX) to create calculated columns, measures, and complex aggregations.
- Utilized DAX to derive additional insights from raw data, such as computing growth rates, trends, and other KPIs essential for decision-making.
- Optimized data models with DAX formulas to improve dashboard performance and ensure real-time data responsiveness.

What is DAX Query?

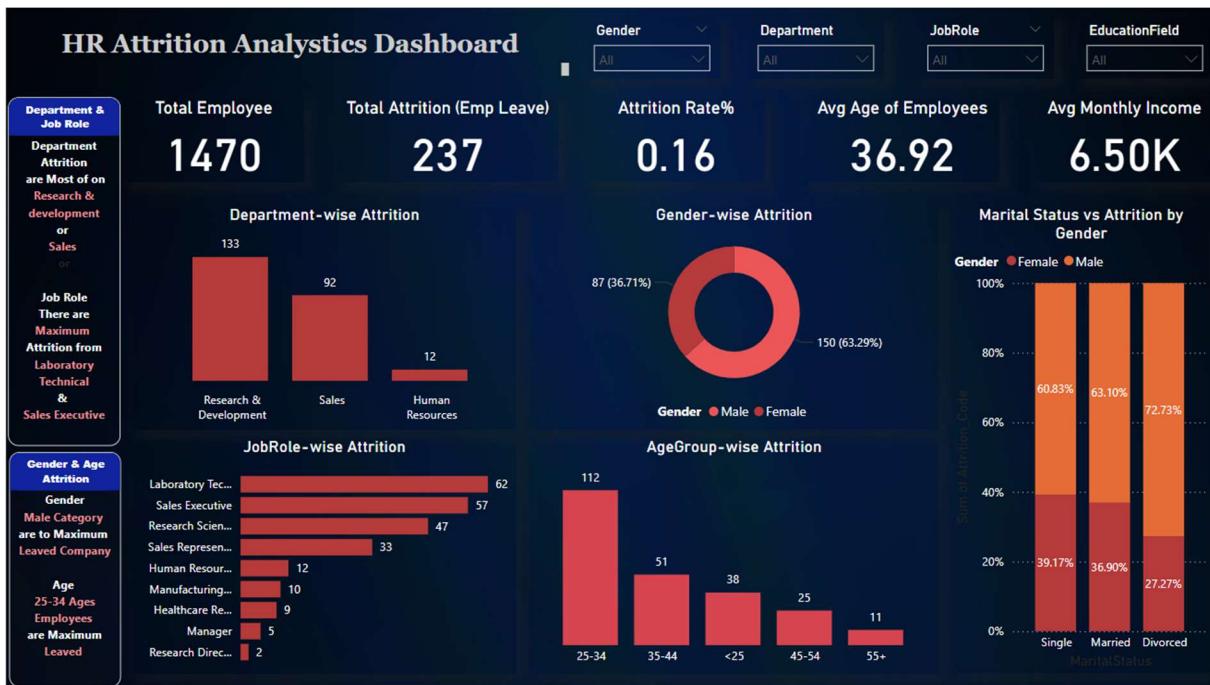
DAX (Data Analysis Expressions) is a powerful formula language used in Power BI, Excel, and SQL Server Analysis Services (SSAS) to create custom calculations and data aggregations. It allows users to define measures and calculated columns, enabling advanced data analysis and reporting.

```
Attrition_Code = IF(  
    HR_TABLE[Attrition] == "Yes", 1, 0  
)
```

```
Age Group = SWITCH(  
SWITCH(  
    TRUE(),  
    'HR_Attrition'[Age] < 25, "<25",  
    'HR_Attrition'[Age] >= 25 && 'HR_Attrition'[Age] <= 35, "25-35",  
    'HR_Attrition'[Age] >= 35 && 'HR_Attrition'[Age] <= 45, "35-45"  
    "45+"  
)
```

↗ Power BI Dashboard Link: [HR_Attrition_Power_BI](#)

❖ Dashboard Previews



❖ Tools & Technologies Used

- Microsoft Excel → Data cleaning, KPI calculation, and visualization.
- Python → Pandas, Matplotlib, Seaborn, Plotly for EDA & visualization.
- Power BI → Interactive dashboard creation with slicers & KPIs.

❖ Conclusion:

The analysis reveals clear trends in **HR Attrition Counts**, **peak Attrition by Categories and Attrition performance**, helping the HR team optimize performance and predictive strategies. By leveraging **Excel, Python and Power BI**, the project demonstrate a strong **multi-tool approach** to data analytics and provides actionable insights for better decision-making.