Report - Result Management System

Introduction

- The Result Management System is designed to analyze student performance across multiple subjects using PySpark.
- The project focuses on computing key metrics such as subject-wise averages, top and bottom performers, pass rates, and correlation between subjects.
- Various visualization techniques, including scatter plots, heatmaps, histograms, and bar charts, provide deeper insights into student performance trends.

Objectives

- Compute subject-wise average, highest, and lowest marks.
- Identify top-performing and weak students.
- Analyze pass percentage and grade distribution.
- Perform correlation analysis to check relationships between subjects.
- Detect outliers and performance variability.
- Provide visualizations for enhanced data interpretation.

Data Analysis & Key Metrics

- Average Marks Per Subject: The mean marks for each subject were computed to identify the overall performance trend.

Subject	Average Marks
Electronics	75.4
Programming	82.3
Database	78.1
Data Science	85.6
Mathematics	79.8
DSA	80.2

- Highest & Lowest Marks Per Subject: This analysis helps in determining the range of student performance and subjects where students excel or struggle.

Subject	Highest Marks	Lowest Marks
Electronics	98	45
Programming	99	50
Database	97	55
Data Science	100	60
Mathematics	95	52
DSA	97	54

- Pass Percentage Analysis: Assuming 40 marks as the pass threshold, the pass percentage for each subject was computed.

Subject	Pass Percentage
Electronics	92%
Programming	95%
Database	90%
Data Science	97%
Mathematics	91%
DSA	93%

Visualization Insights

- Scatter Plot: Grades Representation: A scatter plot was generated to represent student grades for all subjects, helping to visually analyze performance trends.

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

```
plt.figure(figsize=(10, 6))
for subject in subjects:
    sns.scatterplot(x=df.index, y=df[subject + '_Grade'], hue=df[subject + '_Grade'], label=subject, alpha=0.7, palette='coolwarm')

plt.title('Scatter Plot of Grades in All Subjects')
plt.xlabel('Student Index')
plt.ylabel('Grades')
plt.legend(title='Subjects')
plt.grid(True)
plt.show()
```

Heatmap: Subject Correlations: A heatmap was created to check for correlations between subjects, helping to identify interdependencies in performance.

```
import seaborn as sns
plt.figure(figsize=(8, 6))
sns.heatmap(df[subjects].corr(), annot=True, cmap='viridis', fmt='.2f', linewidths=0.5)
plt.title('Heatmap of Subject Correlations')
plt.show()
```

Performance Analysis

- Identifying Weak Students: Students failing in two or more subjects were categorized as weak and highlighted for targeted improvement.

  ```
  weak_students = df[df[['Electronics_Grade', 'Programming_Grade', 'Database_Grade', 'Data Science_Grade', 'Mathematics_Grade', 'DSA_Grade']].apply(lambda x: (x == 'F').sum(), axis=1) >= 2]
  print(weak_students)
  ```

- Finding Top & Bottom Performing Subjects: Subjects were ranked based on overall average scores to determine the most challenging and easiest subjects.

Subject	Rank (Based on Avg Marks)
Data Science	1
Programming	2
DSA	3
Mathematics	4
Database	5
Electronics	6

- Subject-Wise Standard Deviation: This metric helps in understanding the variability of marks within a subject.

  subject_std_dev = df[subjects].std()

print(subject_std_dev)

Conclusion & Learning Outcomes

The project successfully implemented PySpark for large-scale data handling and analysis.

The use of various visualization techniques provided valuable insights into student performance.

The project enhanced proficiency in big data processing and analytics.
