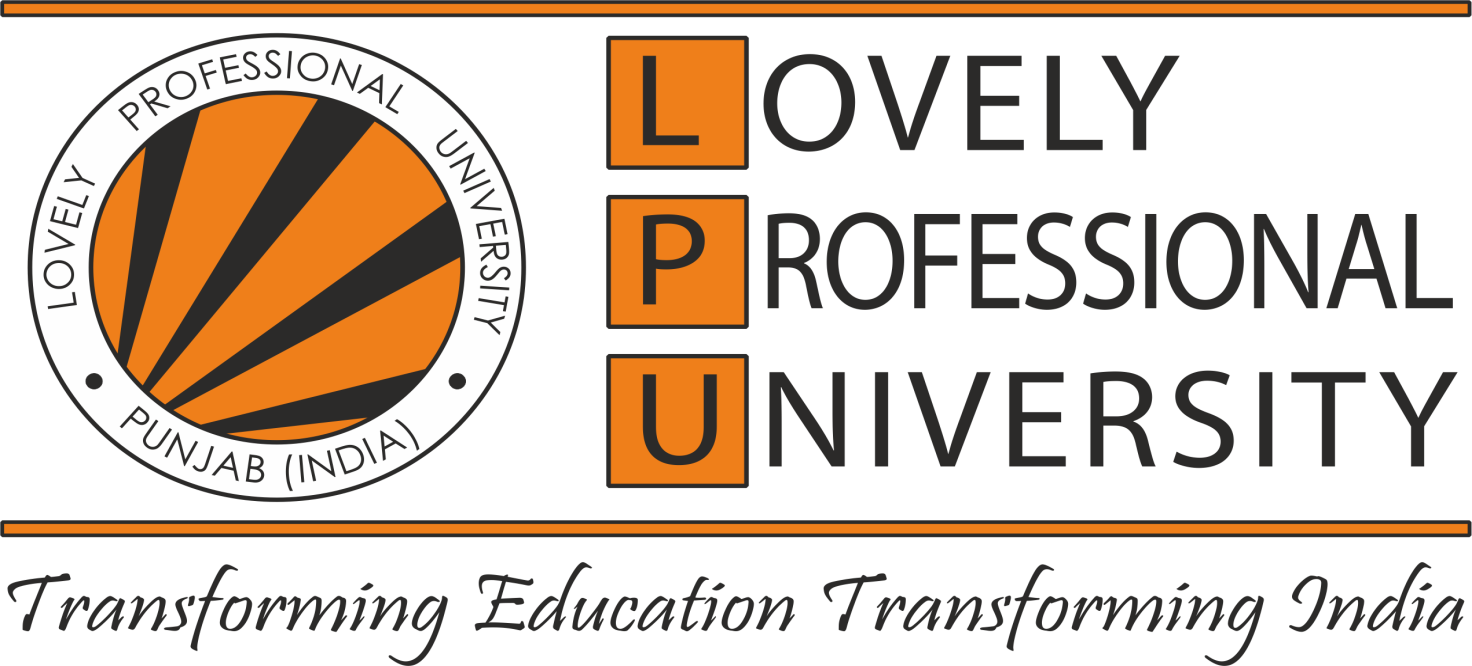
**[Big Data Project - Mentored by Industry SDE](https://www.gradskey.com/discussions/opinion/291)**

**Title: - RESULT MANAGEMENT SYSTEM Report**

|  |  |  |
| --- | --- | --- |
| Registration No | Section | Name of Student |
| 12206055 | 9W084 | Diwakar Sharma |



### **Submitted To Gradskey.**

Lovely Professional University

Jalandhar, Punjab, India

**RESULT MANAGEMENT SYSTEM**

**Project Report**

**Introduction** The Result Management System (RMS) is an academic analytics tool designed to manage and evaluate student performance across a large university. With 10,000 students enrolled in six core subjects, the system efficiently processes marks, performs statistical analysis, and generates insightful visualizations. The primary objective of RMS is to automate result management, provide real-time feedback, and facilitate decision-making for students and faculty.

**Objectives**

* To develop an automated system for processing and analyzing student results.
* To ensure efficient handling of large-scale academic data using modern data processing tools.
* To provide statistical insights into student performance across multiple subjects.
* To visualize data through various graphical representations for better understanding.
* To allow students to review and provide feedback on their results.

**Technologies Used**

To ensure high efficiency and scalability, the RMS leverages modern data processing tools and frameworks:

* **Python**: Used for scripting, automation, and data manipulation.
* **Pandas**: Employed for handling tabular data, performing transformations, and cleaning datasets.
* **PySpark**: A powerful tool for distributed computing, enabling large-scale data analysis.
* **Matplotlib & Seaborn**: Used for visual representation of student performance.

**Implementation Steps**

1. **Data Generation**:
   * Generated synthetic student profiles, including unique roll numbers and names.
   * Created six academic subjects: Electronics, Programming, Database, Data Science, Mathematics, and DSA.
   * Assigned each student random marks between 27 and 100.
   * Stored the dataset in a structured CSV file (students.csv) for processing.
2. **Data Processing**:
   * Imported data using Pandas for initial inspection.
   * Identified and handled missing values, ensuring data integrity.
   * Converted data into a PySpark DataFrame to leverage distributed computing capabilities.
   * Applied transformations and aggregations to prepare the dataset for analysis.
3. **Statistical Analysis**:
   * Computed subject-wise and student-wise averages.
   * Calculated essential metrics, including median, standard deviation, and percentile distributions.
   * Identified top-performing and underperforming students.
   * Generated a summary report showcasing performance trends and key insights.
4. **Data Visualization**:
   * **Bar Chart**: Illustrated the average marks across different subjects, making it easy to compare overall student performance.
   * **Pie Chart**: Represented the proportion of students falling into different grade categories (A, B, C, etc.).
   * **Box Plot**: Visualized mark distributions and identified outliers in student scores.
   * **Histogram**: Depicted the frequency of student scores within specified ranges, offering insights into score distribution.

**Results and Analysis**

The system successfully processed and analyzed student marks, providing valuable insights:

* **Overall Performance**: The average marks across all subjects showed a balanced distribution with minor variations.
* **Top Performers**: The highest-performing students consistently scored above 90% in all subjects.
* **Lowest Scores**: A small percentage of students scored below 40%, indicating the need for academic support.
* **Subject Trends**: Mathematics and Database had the lowest average marks, while Programming had the highest.
* **Grade Distribution**: The pie chart revealed that the majority of students secured grades B and C, with fewer in grade A.
* Efficient data handling using database sharding.

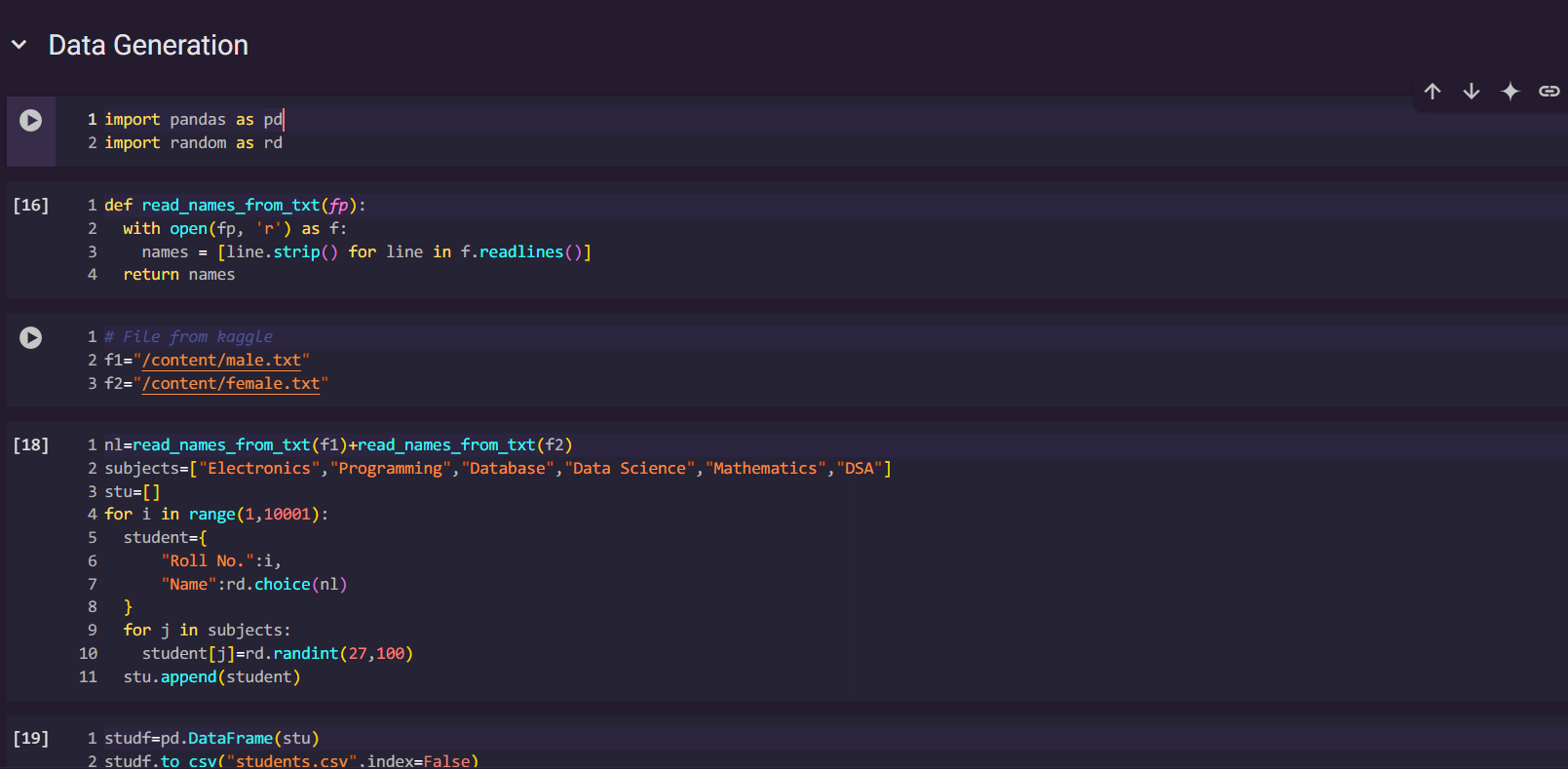
These insights enable institutions to identify struggling students and provide targeted academic interventions.

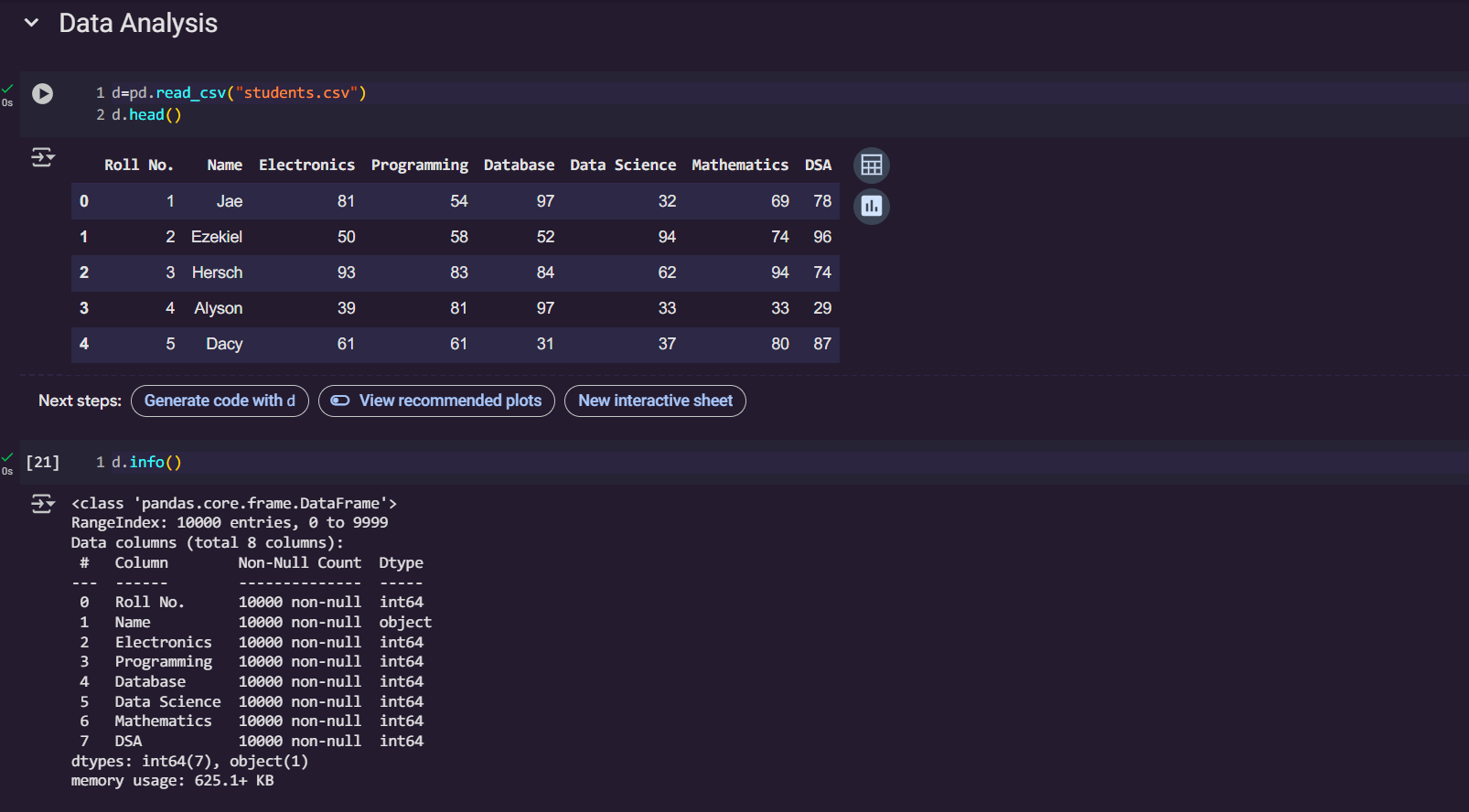
**Future Enhancements**

* Students can review their grades and provide feedback.
* Feedback is stored in a NoSQL database using a key-value format where the key is the student ID, and the value is the feedback comment.

**Conclusion** The Result Management System streamlines student performance evaluation, offering a scalable and data-driven approach to academic analytics. By automating data collection, processing, and visualization, it reduces administrative workload while providing valuable insights. Future enhancements could incorporate machine learning models for automated grading predictions and sentiment analysis on student feedback. Additionally, integrating cloud storage can improve accessibility and scalability for multi-institution deployments.

**CODE SCREENSHOTS**





A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.