## PROJECT DOCUMENTATION

**ON**

**<<STUDENT RESULT ANALYSIS>>**

**B.E.(IT) III -Sem**

BY

**JOHN(160122737180)**

**SHAFAAT (160122737181)**

**SAI BABA(160122737182)**

UNDER THE GUIDANCE OF

**<<DR.N. SUDHAKAR YADAV>>**

<<Associate Professor>>,

IT Department, CBIT.



**DEPARTMENT OF INFORMATION TECHNOLOGY**

**CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY (A)**

**(Affiliated to Osmania University; Accredited by NBA and NAAC, ISO 9001:2015 Certified Institution), GANDIPET, HYDERABAD – 500 075**

**Website:** [**www.cbit.ac.in**](http://www.cbit.ac.in)

**2023-2024**

**Introduction**

**The Student Result Analysis project is designed to provide comprehensive insights into student performance using Python programming and data analysis libraries. By leveraging the power of NumPy, pandas, Matplotlib, and Seaborn, this project offers tools for data manipulation, statistical analysis, and data visualization. Through these capabilities, educators, administrators, and stakeholders can gain valuable insights into student achievements, identify areas for improvement, and make informed decisions to enhance educational outcomes.**

**Installation**

**Before using the Student Result Analysis project, ensure that you have Python installed on your system. Additionally, you'll need to install the required Python libraries using pip, the Python package manager. Run the following command in your terminal or command prompt to install the dependencies:**

**pip install numpy pandas matplotlib seaborn**

**Usage**

**1. Data Loading**

**Begin by loading the student result data from a CSV file into a pandas DataFrame using the pd.read\_csv() function. This step prepares the data for further analysis and visualization.**

**import pandas as pd**

**# Load student result data from CSV file**

**df = pd.read\_csv("student\_result.csv")**

**2. Data Exploration**

**Explore the loaded data by examining the first few rows of the DataFrame using df.head(). This provides an initial overview of the dataset's structure, columns, and values.**

**print(df.head())**

**3. Data Information**

**Obtain detailed information about the DataFrame, including the data types, non-null counts, and memory usage, using df.info(). This helps in understanding the dataset's characteristics and identifying potential data quality issues.**

**df.info()**

**4. Handling Missing Values**

**Check for missing values in the dataset using df.isnull().sum(). This allows for identifying columns with missing data and determining appropriate strategies for handling them, such as imputation or removal.**

**print(df.isnull().sum())**

**5. Data Cleaning**

**Clean the data by removing any unnecessary columns or rows. For example, you can drop the "Unnamed: 0" column using the df.drop() method.**

**df = df.drop("Unnamed: 0", axis=1)**

**6. Data Transformation**

**Perform data transformations as needed to ensure consistency and correctness. For instance, you can replace specific values in a column using string manipulation techniques.**

**df["WklyStudyHours"] = df["WklyStudyHours"].str.replace("05-Oct", "5-10")**

**7. Data Visualization (Additional Insights)**

**Explore additional insights into the student data through various visualizations, including histograms, box plots, count plots, and heatmaps. These visualizations provide deeper insights into student demographics, academic performance, and the impact of different factors on student outcomes.**

**# Visualize Gender Distribution using a count plot**

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

**ax = sns.countplot(data=df, x="Gender")**

**ax.bar\_label(ax.containers[0])**

**plt.title("Gender Distribution")**

**plt.show()**

**# Explore the relationship between Parent's Education and Student Scores using a heatmap**

**gb = df.groupby("ParentEduc").agg({"MathScore": "mean", "ReadingScore": "mean", "WritingScore": "mean"})**

**sns.heatmap(gb, annot=True)**

**plt.title("Relationship between Parents Education and Student Scores")**

**plt.show()**

**# Visualize the distribution of Math, Reading, and Writing Scores using box plots**

**sns.boxplot(data=df)**

**plt.show()**

**sns.boxplot(data=df, x="MathScore")**

**plt.show()**

**sns.boxplot(data=df, x="ReadingScore")**

**plt.show()**

**# Explore the distribution of students across Ethnic Groups using a pie chart and count plot**

**ethnic\_groups = df["EthnicGroup"].unique()**

**group\_counts = [df.loc[df["EthnicGroup"] == group].count()["EthnicGroup"] for group in ethnic\_groups]**

**plt.pie(group\_counts, labels=ethnic\_groups, autopct="%1.2f%%")**

**plt.title("Ethnic Group Distribution")**

**plt.show()**

**sns.countplot(data=df, x="EthnicGroup")**

**plt.show()**

**# Visualize the impact of Sports Practice on Student Scores using a heatmap**

**gb2 = df.groupby("PracticeSport").agg({"MathScore": "mean", "ReadingScore": "mean", "WritingScore": "mean"})**

**sns.heatmap(gb2)**

**plt.title("Impact of Sports Practice on Student Scores")**

**plt.show()**

**Conclusion**

**The Student Result Analysis project offers a powerful toolkit for analyzing and visualizing student performance data. By following the provided usage instructions and exploring additional insights, users can gain valuable insights into student achievements, demographic trends, and factors influencing academic success. Through informed analysis and visualization, educators and stakeholders can make data-driven decisions to enhance educational outcomes and support student success.**