QA Team Productivity Tracker

Objective

To analyze tester performance metrics and automation coverage to:

- Explore productivity trends across testers and modules (EDA + visualizations).
- Identify relationships between workload, defect count, and closure time.
- Predict the **average defect closure time** using regression analysis.

Step-by-Step Workflow

1. Data Generation

- Use numpy and pandas to create random data for all columns.
- Introduce correlations (e.g., higher Defects_Found → higher Avg_Closure_Time).
- Save as qa_team_productivity.csv.

---

2. Data Exploration

- Use .info(), .describe(), .isnull().sum() to inspect structure and completeness.
- Introduce a few missing values in Automation_Coverage and Defect_Severity_Avg to practice imputation.

---

3. Handling Missing Values

- Numeric columns: replace with mean() or median().
- If any categorical columns are added later (like Module), fill with mode.

---

4. Outlier Detection

- Identify outliers in numeric columns (Execution_Hours, Defects_Found, Avg_Closure_Time) using:
    - Z-score
    - IQR method
- Remove or cap extreme values to avoid model distortion.

---

5. Feature Engineering

- Create new ratio features:
    - Defects_per_Test = Defects_Found / Total_TestCases_Executed
    - Execution_per_Test = Execution_Hours / Total_TestCases_Executed
- Normalize numeric columns using StandardScaler, MinMaxScaler, and RobustScaler for comparison.

- Encode categorical columns (Module) using pd.get_dummies().

---

6. EDA & Visualization

Visualize productivity relationships using:
- **Correlation Heatmap** — identify key dependencies between variables.
- **Boxplot:** Defect_Severity_Avg vs Avg_Closure_Time.
- **Bar Chart:** Average closure time per module.
- **Scatter Plot:** Total_TestCases_Executed vs Execution_Hours to visualize workload.
- **Histogram + KDE:** Distribution of closure time and execution hours.

Use both matplotlib and seaborn for plots.

---

7. Model Building (Linear Regression)

Objective: Predict Avg_Closure_Time based on workload, severity, automation coverage, and execution effort.

---

8. Insights & Interpretation

- Identify which features most influence closure time using regression coefficients.
- Example insights:
    - "Testers handling highly automated modules spend **30% less time per closure**."
    - "Closure time increases non-linearly once testers execute more than 50 test cases per week."
    - "Higher average defect severity leads to longer closure times."
    - "Balanced workload (30–40 test cases) yields optimal closure efficiency."