**🔍 1. Understand the Dataset**

Before diving into code:

* **Know the context**: What is the dataset about? What’s the business problem?
* **Understand column meanings**: Are there descriptions? Units? Categorical or numerical?

**📦 2. Structural Checks**

These checks ensure the data is loaded and formatted correctly.

| **Check** | **Why It Matters** |
| --- | --- |
| ✅ **Headers Present & Named Properly** | Missing or ambiguous headers make analysis difficult. |
| 📏 **Dimensions (rows × columns)** | Gives an idea of the dataset’s size and complexity. |
| 📂 **File format & encoding** | Ensure the file opens correctly (UTF-8, delimiters, etc.). |
| 🔁 **Duplicates (rows or IDs)** | Duplicated entries can skew results. |
| 📌 **Indexing issues** | Check for properly set indices, especially in time series or relational data. |

**🧼 3. Data Cleaning / Quality Checks**

| **Check** | **Why It Matters** |
| --- | --- |
| 🕳️ **Missing values (NaNs)** | Missing data can affect models and summaries. |
| ⚖️ **Data types (int, float, object, datetime)** | Ensures math/statistics are applied correctly. |
| 🔀 **Mixed types in same column** | E.g., numbers + strings in the same column break aggregations. |
| 🔂 **Constant Columns** | No information gain; often redundant. |
| 🎭 **Outliers & Anomalies** | May be data entry errors or true but rare cases. |
| 🔤 **String normalization** | Whitespace, casing, special characters need standardization. |
| 🔁 **Date parsing & standardization** | Inconsistent formats (e.g., dd-mm-yyyy vs yyyy-mm-dd). |
| 🌐 **Encoding Issues** | Special characters not displayed correctly (Ã© instead of é). |

**📊 4. Data Distribution & Summary**

| **Task** | **Purpose** |
| --- | --- |
| 📉 **Descriptive stats (mean, std, min, max)** | Spot strange values or potential outliers. |
| 📈 **Distribution plots (histograms, boxplots)** | Understand spread and skewness of numerical variables. |
| 📋 **Value counts for categoricals** | Spot typos or unusual values. |
| 📚 **Cardinality check** | High-cardinality categoricals may need encoding decisions. |
| ⏳ **Time-based patterns (if datetime present)** | Check trends, seasonality, or missing periods. |

**🔗 5. Relationship Checks**

| **Task** | **Why It’s Useful** |
| --- | --- |
| 🤝 **Correlation matrix** | Understand linear relationships and multicollinearity. |
| 🔀 **Cross-tabulations** | Compare two categorical variables. |
| 🧮 **GroupBy summaries** | Spot patterns across groups. |
| 🔁 **Target leakage** | Ensure no variables leak future information into models. |

**🧠 6. Assumptions for Modeling**

When preparing for ML/statistical modeling:

| **Check** | **Reason** |
| --- | --- |
| 🎯 **Target variable balance (for classification)** | Imbalanced classes need special handling. |
| 🔁 **Feature redundancy** | Highly correlated features may be redundant. |
| 📏 **Scale of variables** | Some algorithms are sensitive to feature scale. |
| 🧹 **Encoding requirements** | Categorical → numeric (one-hot, label encoding). |
| 📆 **Time dependency** | For time series or causal modeling. |

**🧪 7. Business-Specific Checks**

| **Check** | **Example** |
| --- | --- |
| 🧾 **Rule violations** | e.g., Age can’t be negative, delivery date must be after order date. |
| 🔢 **Expected ranges** | Salary, grades, temperatures should lie within sensible bounds. |
| 🎓 **Domain validation** | E.g., known locations, product categories, or industry norms. |

**🧰 Tools Commonly Used**

* **Pandas** (Python)
* **NumPy** for numerical checks
* **Matplotlib / Seaborn / Plotly** for visualization
* **Missingno** (missing data patterns)
* **Sweetviz / Pandas Profiling / ydata-profiling** (automated EDA reports)