

PRODUCT REQUIREMENTS DOCUMENT (PRD)

1. Product Overview

Product Name: SmartClean

Product Type: Python Data Cleaning Library

Primary Domain: Data Engineering / Data Science Tooling

SmartClean is an open-source Python package designed to automatically detect and correct common data quality issues in tabular datasets. The library integrates directly with **pandas DataFrames** and provides both automatic and customizable cleaning pipelines.

The goal is to reduce the time data scientists spend performing repetitive preprocessing tasks by providing intelligent default strategies and a simple, intuitive API.

2. Problem Statement

In most data science workflows:

- **60–80% of time is spent cleaning data**
- Data cleaning scripts are repetitive
- Data quality problems vary across datasets
- Manual cleaning increases chances of human error

Common issues include:

- Missing values
- Duplicate records
- Incorrect data types
- Inconsistent column names

- Outliers
- Text inconsistencies

Existing tools like pandas provide primitives, but **do not provide a cohesive automated cleaning pipeline.**

SmartClean aims to address this gap.

3. Target Users

Primary Users

Data Scientists

Need quick dataset preparation before model training.

Data Analysts

Work with messy datasets from Excel, CSV exports, surveys.

Machine Learning Engineers

Need reliable preprocessing pipelines.

Students / Researchers

Require easy dataset preparation tools.

4. Product Goals

Primary Goals

1. Reduce time spent cleaning datasets.
2. Provide automatic dataset quality diagnostics.
3. Offer intelligent default cleaning strategies.
4. Provide transparent cleaning reports.

- 5. Maintain full compatibility with pandas.

Secondary Goals

- Encourage reproducible data cleaning workflows
 - Provide extensibility for advanced users
 - Enable automated data quality evaluation
-

5. Key Features

5.1 Dataset Ingestion

SmartClean should support loading datasets from common formats.

Supported formats

- CSV
- Excel
- JSON
- Pandas DataFrame

Capabilities

- delimiter detection
- encoding detection
- automatic type inference

Example usage:

```
df=sc.read("dataset.csv")
```

5.2 Dataset Profiling

SmartClean should inspect datasets before cleaning.

Profiling Metrics

- number of rows
- number of columns
- missing value counts
- duplicate records
- column data types
- potential outliers

Example:

Dataset Profile

Rows: 12,450

Columns: 10

Missing Values

age: 32

salary: 5

Duplicate Rows

3 detected

5.3 Missing Value Handling

The system must detect missing values and apply strategies.

Strategies

Numeric columns:

- mean
- median
- interpolation

Categorical columns:

- mode
- constant value ("unknown")

Datetime columns:

- forward fill
- backward fill

Auto strategy example:

```
numeric → median  
categorical → mode  
datetime → forward fill
```

5.4 Duplicate Detection

Capabilities:

- detect duplicate rows
- remove duplicates
- subset-based duplicate removal

Example:

```
sc.remove_duplicates(df)
```

5.5 Data Type Correction

SmartClean should detect incorrect types.

Examples:

Incorrect	Correct
"25"	int
"2023-10-10"	datetime
"TRUE"	boolean

Functions include:

- numeric conversion
 - datetime conversion
 - categorical detection
-

5.6 Text Cleaning

Text columns often contain formatting problems.

Features:

- whitespace trimming
- case normalization
- removal of special characters
- category normalization

Example:

```
" USA " → "USA"  
"female" → "Female"
```

5.7 Column Name Standardization

Columns should be standardized.

Example transformation:

```
First Name → first_name  
TOTAL SALES → total_sales  
customer-id → customer_id
```

5.8 Outlier Detection

SmartClean should detect extreme values.

Supported methods

- Interquartile Range (IQR)
- Z-score

Handling options

- remove outliers
 - cap outliers
 - flag outliers
-

5.9 Automatic Cleaning Pipeline

The core feature.

One command performs full cleaning.

Example:

```
df=sc.auto_clean(df)
```

Pipeline steps:

1. profile dataset
 2. clean column names
 3. fix data types
 4. handle missing values
 5. remove duplicates
 6. clean text fields
 7. detect and handle outliers
-

5.10 Cleaning Reports

After cleaning, the library generates a report.

Example:

Cleaning Summary

Missing values filled: 42

Duplicate rows removed: 5

Columns converted: 3

Outliers capped: 7

6. Non-Goals (Initial Release)

SmartClean will NOT include:

- machine learning models
- feature engineering
- deep data validation frameworks
- visualization tools

These may appear in later versions.

7. Success Metrics

The project will be considered successful if:

- users can clean datasets with **one function call**
- documentation allows usage within **5 minutes**
- codebase remains modular and maintainable
- package gains adoption on **PyPI and GitHub**