Phase 1: Quality Control Notebook Report

This report summarizes each step performed in the `Quality_Control.ipynb` notebook, including code snippets and descriptions of their purpose.

---

1. Setup and Imports

Purpose: Load the libraries required for data manipulation and schema validation.

```
import pandera as pa
import pandas as pd
from pandera import Column, DataFrameSchema, Check, Index
from pandera.errors import SchemaErrors
```

---

2. Data Loading

Purpose: Read the processed SSH log dataset into a pandas DataFrame.

```
file_path = r"C:\ZC\Data
Governance\DataGovernanceWorkflow\data\ssh_logs_processed.csv"
df = pd.read_csv(file_path)
```

---

3. Initial Data Inspection

Purpose: Examine the structure and summary statistics of the dataset.

```
df.info()
df.describe(include='all')
```

- `df.info()` displays column names, non-null counts, and data types.
- `df.describe(include='all')` provides count, unique, top, and frequency for categorical columns, and basic statistics for numeric ones.

---

4. Data Cleaning

4.1 Removing Duplicate Rows

```
df.drop_duplicates(inplace=True)
```

4.2 Checking for Missing Values

```
if df.isnull().sum().sum() > 0:
    # Identify columns and their data types
    dtype_map = df.dtypes.to_dict()
    num_cols = [col for col, dt in dtype_map.items() if dt in ['int64',
'float64']]
    cat_cols = [col for col, dt in dtype_map.items() if dt == 'object']

    # Handle missing in numeric columns
    for col in num_cols:
        df[col].fillna(df[col].median(), inplace=True)

    # Handle missing in categorical columns
    for col in cat_cols:
        df[col].fillna(df[col].mode()[0], inplace=True)
```

- Numeric missing values filled with median.
- Categorical missing values filled with mode.

4.3 Handling Outliers

```
for col in num_cols:
    Q1 = df[col].quantile(0.25)
    Q3 = df[col].quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    df = df[(df[col] >= lower_bound) & (df[col] <= upper_bound)]
```

- Removes data points outside 1.5 × IQR.

---

5. Preview of Cleaned Dataset

```
print("Sample of cleaned dataset:")
print(df.head())
```

---

6. Data Validation with Pandera Schema

Purpose: Define and enforce a schema for the dataset.

```
schema = pa.DataFrameSchema(
    {
        "Date": Column(pa.DateTime, coerce=True, nullable=True),
        "User": Column(str, nullable=False,
checks=Check.str_length(min_value=1)),
        "Host": Column(str, nullable=False,
checks=Check.str_length(min_value=1)),
        "Action": Column(str, nullable=True),
        # ... additional columns as needed
    }
)

try:
    validated_df = schema.validate(df, lazy=True)
    print("✅ Validation passed.")
    print(validated_df)
except SchemaErrors as err:
    print("❌ Validation failed. Issues found:")
    print(err.failure_cases)
```

End of report.