Data cleaning on Titanic Dataset:

1. PassengerId column

Cleaning:
Ensure PassengerId is unique and not missing. It serves as a unique identifier.
a. # Check and remove any duplicate entries based on PassengerId
b. # Drop rows with missing PassengerId (shouldn’t happen in Titanic dataset)
c. # Convert PassengerId to integer type if not already

---

2. Name column

Cleaning:
Name values should be complete, clean, and consistently formatted.
a. # Remove leading and trailing whitespaces using str.strip()
b. # Drop rows where Name is missing
c. # Remove or replace non-ASCII / special characters
d. # Ensure each name contains at least first and last name (basic validation)

---

3. Survived column

Cleaning:
Ensure Survived contains only 0 or 1 (no nulls or invalid entries).
a. # Check for null values and decide how to handle them (if any)
b. # Validate that values are either 0 or 1
c. # Convert column to integer type if needed

---

4. Pclass column

Cleaning:
Passenger class should contain only 1, 2, or 3.
a. # Fill or remove missing values in Pclass
b. # Check and remove invalid values (outside 1, 2, 3)
c. # Convert to integer or categorical type for consistency

---

5. Sex column

Cleaning:
Ensure values are only “male” or “female” (case-insensitive).
a. # Convert all text to lowercase for consistency
b. # Handle missing values by imputing or dropping
c. # Check for spelling mistakes or unexpected values (e.g., “femail”, “m”)
d. # Map to standard categories (e.g., ‘male’, ‘female’)

---

6. Age column

Cleaning:
Ensure ages are numeric and reasonable.
a. # Convert Age to numeric type
b. # Identify and impute missing ages (e.g., with median or mean)
c. # Remove or flag unrealistic values (e.g., negative ages or > 100)
d. # Check for outliers that might require treatment

---

7. SibSp column

Cleaning:
Ensure sibling/spouse count is numeric and valid.
a. # Convert to integer type
b. # Check for negative values and correct/remove them
c. # Fill missing values with 0 or appropriate imputation

d. # Validate value distribution (e.g., no extremely high unrealistic values)

---

8. Parch column

Cleaning:
Parent/children count should be non-negative integers.
a. # Convert to integer type
b. # Fill missing values
c. # Check for negative values
d. # Validate distribution (e.g., 0–6 range is expected)

---

9. Ticket column

Cleaning:
Ticket numbers should be clean and consistent.
a. # Strip leading/trailing whitespaces
b. # Handle missing values appropriately
c. # Remove unnecessary special characters or extra spaces
d. # Standardize ticket prefixes (e.g., “A/5 21171” vs “A/5 21171 ”)

---

10. Fare column

Cleaning:
Fares should be numeric and non-negative.
a. # Convert Fare to numeric type
b. # Fill missing fares with median or mean of the class
c. # Remove negative or unrealistic fare values
d. # Round to reasonable decimal places if needed

---

11. Cabin column

Cleaning:
Cabin should have consistent formatting; many are missing.
a. # Handle missing values (e.g., fill with “Unknown” or keep as NaN)
b. # Strip whitespaces and standardize cabin format (e.g., “C85”)
c. # Remove any special characters or inconsistent entries
d. # Extract deck letter (optional for further processing)

---

12. Embarked column

Cleaning:
Embarkation point should only be “C”, “Q”, or “S”.
a. # Fill missing values with the mode (most frequent value)
b. # Convert to uppercase and remove spaces
c. # Remove invalid entries (e.g., lowercase or unknown letters)
d. # Standardize categorical values (C, Q, S only)