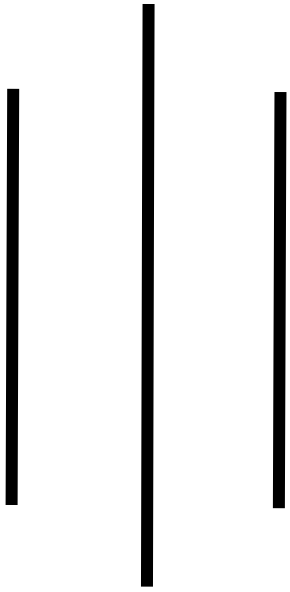
Assignment – I

Data Cleaning Using String Functions for Variables and Data Structures



Submitted by: Bikram Giri

Submitted to: Samriddha Pathak

Submission Date: 2025/07/06

Table of contents

Remove all leading digits from a string using lstrip().

14. Clean Nested List:

Remove spaces from each string in a nested list.

15. Capitalize After Cleaning:

Clean a string and capitalize the first letter.

16. Clean Dictionary Keys:

Remove trailing underscores from all dictionary keys.

17. Clean and Deduplicate Names:

Clean, capitalize, and deduplicate names in a list.

18. Remove Multiple Characters:

Remove *, -, and spaces from both ends of a string.

19. Conditional Cleaning in List:

Remove leading # only if present in each string in a list.

20. Clean and Group by First Letter:

Clean and group product names by their first letter (case-insensitive).'

Hard

21. Clean Set of Strings:

Clean all strings in a set of special-character-laden strings.

22. Complex Nested Cleaning:

Clean and title-case all strings in dictionary values (lists).

23. Custom Title Function:

Implement your own title() using only capitalize() and loops.

24. Clean and Format Emails:

Clean emails (strip spaces, lowercase all, except capitalize first letter before @).

25. Multi-Step Cleaning

Remove leading numbers, trailing punctuation, and title-case the string.

26. In-Place Cleaning:

Clean and title-case a list of strings in place (no new list).

27. Clean and Count Unique Words:

Clean sentences, split into words, count unique words.

28. Clean Dictionary Sentences:

Clean and capitalize only the first word of each dictionary value.

29. Selective Character Removal:

Remove only leading/trailing underscores and dashes.

30. Batch Clean and Sort:

Clean and title-case product codes, then sort.

Easy

1. Remove Spaces:

Remove leading and trailing spaces from a string using strip().

```
text = "  Jay Nepal  "
cleaned_text = text.strip()
print(cleaned_text)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs/Pyt
Jay Nepal
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

2. Remove Leading Characters:

Remove all leading asterisks from a string using lstrip().

```
text = "****Artificial Intelligence"
cleaned_text = text.lstrip("*")
print(cleaned_text)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs/Py
Artificial Intelligence
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

3. Remove Trailing Characters

Remove all trailing exclamation marks from a string using rstrip().

```
text = "Be honest!!!!"
cleaned_text = text.rstrip("!")
print(cleaned_text)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs/
Be honest
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

4. Capitalize a Sentence

Capitalize only the first letter of a lowercase sentence using capitalize().

```
sentence = "artificial intelligence is the future."
cleaned_text = sentence.capitalize()
print(cleaned_text)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs
Artificial intelligence is the future.
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

5. Title Case a Name

Convert a name to title case using title().

```
sentence = "discipline is the key to success"
cleaned_text = sentence.title()
print(cleaned_text)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs
I.py"
Discipline Is The Key To Success
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

6. Clean List of Names:

Remove leading and trailing spaces from each name in a list.

```
names = ["  Elephant  ", "  Tiger  ", "  Cow  ", "  Dog  ", "  Monkey  "]
cleaned_names = [name.strip() for name in names]
print(cleaned_names)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/
['Elephant', 'Tiger', 'Cow', 'Dog', 'Monkey']
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

7. Remove Custom Characters:

Remove # and $ from both ends of a string using strip().

```
text = "##Python Programming$$$"
cleaned_text = text.strip("#$")
print(cleaned_text)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs/
Python Programming
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

8. Capitalize All Names in List:

Capitalize each name in a list using capitalize().

```
names = ["elephant", "tiger", "cow", "dog", "monkey"]
capitalized_names = [name.capitalize() for name in names]
print(capitalized_names)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs/
['Elephant', 'Tiger', 'Cow', 'Dog', 'Monkey']
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

9. Clean Dictionary Values:

Remove trailing spaces from all values in a dictionary.

```python
data = {
    "name": "  Bidhan Karki   ",
    "age": "  28   ",
    "city": "  Dharan   "
}
cleaned_data = {key: value.strip() for key, value in data.items()}
print(cleaned_data)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs
{'name': 'Bidhan Karki', 'age': '28', 'city': 'Dharan'}
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

10. Title Case Sentences in List:

Convert each sentence in a list to title case.

```python
sentences = ["artificial intelligence is the future.", "machine learning is a subset of AI.",
             "natural language processing enables communication."]
title_cased_sentences = [sentence.title() for sentence in sentences]
print(title_cased_sentences)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs/Python/Python313/python.exe "d:/Digital Pa
['Artificial Intelligence Is The Future.', 'Machine Learning Is A Subset Of Ai.', 'Natural Language Processing Enables Communication.']
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

Intermediate

11. Clean and Title Case:

Remove spaces and convert to title case.

```python
sentences = "  Ram is a good boy.  "
cleaned_sentences = sentences.strip().title()
print(cleaned_sentences)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs/
Ram Is A Good Boy.
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

12. Clean List of Emails:

Remove leading/trailing spaces from each email in a list.

```
emails = ["  ram@example.com  ", "  shyam@example.com  ", "  sita@example.com  "]
cleaned_emails = [email.strip() for email in emails]
print(cleaned_emails)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs
['ram@example.com', 'shyam@example.com', 'sita@example.com']
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

13. Remove Leading Numbers:

Remove all leading digits from a string using lstrip().

```
text = "09265Hello World!"
cleaned_text = text.lstrip("09265")
print(cleaned_text)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs
Hello World!
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

14. Clean Nested List:

Remove spaces from each string in a nested list.

```
nested_list = [["  Ram  ", "  Shyam  "], ["  Sita  "], ["  Kiran  ", "  Himal  "]]
cleaned_nested_list = [[name.strip() for name in sublist] for sublist in nested_list]
print(cleaned_nested_list)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs
[['Ram', 'Shyam'], ['Sita'], ['Kiran', 'Himal']]
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

15. Capitalize After Cleaning:

Clean a string and capitalize the first letter.

```
sentence = "-& he *97is a && doctor. __%  "
cleaned_sentence = sentence.strip("-& _%  ").replace(" *97", " ").replace(" && ", " ").capitalize()
print(cleaned_sentence)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs
He is a doctor.
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

16. Clean Dictionary Keys:

Remove trailing underscores from all dictionary keys.

```
data = {
    "name_": "  Bidhan Karki   ",
    "age_": "  28   ",
    "city_": "  Dharan   "
}
cleaned_data = {key.rstrip("_"): value.strip() for key, value in data.items()}
print(cleaned_data)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs
{'name': 'Bidhan Karki', 'age': '28', 'city': 'Dharan'}
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

17. Clean and Deduplicate Names:

Clean, capitalize, and deduplicate names in a list.

```
planets = ["  Mercury  ", "  venus ", "  MERCURY", "Venus  "]
cleaned_planets = list(set(planet.strip().capitalize() for planet in planets))
print(cleaned_planets)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs
['Mercury', 'Venus']
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

18. Remove Multiple Characters:

Remove *, -, and spaces from both ends of a string.

```
text = "--*Python Programming**--   "
cleaned_text = text.strip("*-  ")
print(cleaned_text)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs
Python Programming
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

19. Conditional Cleaning in List:

Remove leading # only if present in each string in a list.

```
birds = ["Peacock", "#Crow", "#Parrot", "Eagle"]
cleaned_birds = [bird.lstrip("#") for bird in birds]
print(cleaned_birds)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs
['Peacock', 'Crow', 'Parrot', 'Eagle']
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

20. Clean and Group by First Letter:

Clean and group product names by their first letter (case-insensitive).’

```
names = ["  Ram  ", "  binod  ", "  Ramesh  ", "  Kamal  ", "  Bishal  ", " Manish"]
cleaned_names = [name.strip().capitalize() for name in names]
grouped_names = {}
for name in cleaned_names:
    first_letter = name[0]
    if first_letter not in grouped_names:
        grouped_names[first_letter] = []
    grouped_names[first_letter].append(name)
print(grouped_names)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs/
{'R': ['Ram', 'Ramesh'], 'B': ['Binod', 'Bishal'], 'K': ['Kamal'], 'M': ['Manish']}
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

Hard

21. Clean Set of Strings:

Clean all strings in a set of special-character-laden strings.

```
vegetables = {" --@Tomato  !", "  #&potato__  ", "$carrot  ", "  %Cabbage(*)"}
cleaned_vegetables = {veg.strip(" -@ !  #&_  $   %(*)").capitalize() for veg in vegetables}
print(cleaned_vegetables)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs/
{'Tomato', 'Cabbage', 'Potato', 'Carrot'}
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

22. Complex Nested Cleaning:

Clean and title-case all strings in dictionary values (lists).

```
data = {
    "birds": ["  &**peacock  ", "Crow^^  ", " parrot89  "],
    "animals": ["  __Elephant#@  ", "  tiger12", "44Lion  "],
}
cleaned_data = {key: [value.strip("  &**  ^^   89    _#@    1244  ").title() for value in values]
                for key, values in data.items()}
print(cleaned_data)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs
{'birds': ['Peacock', 'Crow', 'Parrot'], 'animals': ['Elephant', 'Tiger', 'Lion']}
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

23. Custom Title Function:

Implement your own title() using only capitalize() and loops.

```
def title(t):
    return ' '.join(word.capitalize() for word in t.split())
text = "hello world, this is a test."
title_cased_text = title(text)
print(title_cased_text)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs
Hello World, This Is A Test.
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

24. Clean and Format Emails:

Clean emails (strip spaces, lowercase all, except capitalize first letter before @).

```
emails = ["  ram@example.com  ", "  shyam@example.com  ", "  sita@example.com  "]
cleaned_emails = []
for email in emails:
    email = email.strip().lower()
    local_part, domain = email.split("@")
    local_part = local_part.capitalize()
    cleaned_emails.append(f"{local_part}@{domain}")
print(cleaned_emails)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs/
['Ram@example.com', 'Shyam@example.com', 'Sita@example.com']
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

25. Multi-Step Cleaning

Remove leading numbers, trailing punctuation, and title-case the string.

```
text = " 8346Good Evening !!!"
cleaned_text = text.lstrip(" 8346").rstrip(" !!!").title()
print(cleaned_text)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs/
Good Evening
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

26. In-Place Cleaning:

Clean and title-case a list of strings in place (no new list).

```
fruits = ["MANGO#$  ", "  45Orange --", " banana$# "]
for i in range(len(fruits)):
    fruits[i] = fruits[i].strip("#$    45 - $# ").title()
print(fruits)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs/
['Mango', 'Orange', 'Banana']
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

27. Clean and Count Unique Words:

Clean sentences, split into words, count unique words.

```
sentences = [" 88Good Morning  ", "good afternon! --%  ", "  @good Evening (*)_"]
cleaned_sentences = [sentence.strip(" 88   --%    @  (*)_").title() for sentence in sentences]
unique_words = set(word for sentence in cleaned_sentences for word in sentence.split())
print(unique_words)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs
{'Morning', 'Afternon!', 'Evening', 'Good'}
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

28. Clean Dictionary Sentences:

Clean and capitalize only the first word of each dictionary value.

```
data = {
      "s1": "  --hello world! 99",
      "s2": "##ai is amazing.__",
      "s3": "  77ai and ML are the future.  **"
      }
cleaned_data = {key: value.strip("  -- 99##__  77  **").capitalize() for key, value in data.items()}
print(cleaned_data)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs
{'s1': 'Hello world!', 's2': 'Ai is amazing.', 's3': 'Ai and ml are the future.'}
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

29. Selective Character Removal:

Remove only leading/trailing underscores and dashes.

```
text = "--__Python Programming__--"
cleaned_text = text.strip("-_")
print(cleaned_text)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs/
Python Programming
PS D:\Digital Pathshala\AI ML With Python\New folder>
```

30. Batch Clean and Sort:

Clean and title-case product codes, then sort.

```
product_codes = ["**pROD123  ", "  --88prod456--  ", "  0-0PROD789 --__  ", " PROD999 (&)"]
cleaned_product_codes = [code.strip("**    --88--    0-0  --__    (&)").title() for code in product_codes]
cleaned_product_codes.sort()
print(cleaned_product_codes)
```

Output:

```
PS D:\Digital Pathshala\AI ML With Python\New folder> & C:/Users/Admin/AppData/Local/Programs/
['Prod123', 'Prod456', 'Prod789', 'Prod999']
PS D:\Digital Pathshala\AI ML With Python\New folder>
```