In [40]:
```python
import re
from collections import Counter


def main():
    # Input string from the user
    input_string = input("Enter a string: ")

    # Use Regex to extract words - split by non-alphabetic characters
    word_pattern = re.split(r"[^A-Za-z]+", input_string)

    # Filter out empty strings and normalize case for word processing
    words = [word for word in word_pattern if word]  # All words (case-sensitive
    normalized_words = [word.lower() for word in words]  # Words in lowercase

    # Count unique words and calculate word frequencies
    unique_words = set(normalized_words)  # Unique words (case-insensitive)
    word_frequencies = Counter([word.title() for word in normalized_words])  # F

    # Display results
    print(f"\nExtracted Words: {' '.join(words)}")
    print(f"Number of Words: {len(words)}")
    print(f"Number of unique Words: {len(unique_words)}")
    print("Word Frequencies:")
    for word, count in word_frequencies.items():
        print(f"{word}: {count}")


if __name__ == "__main__":
    main()
```

```
Extracted Words: Hello Python Hello World python is amazing Is it not
Number of Words: 10
Number of unique Words: 7
Word Frequencies:
Hello: 2
Python: 2
World: 1
Is: 2
Amazing: 1
It: 1
Not: 1
```

Another Approach Without Using `Counter` function from Collections

In [44]:
```python
import re


def main():
    # Input string from the user
    input_string = input("Enter a string: ")

    # Use Regex to extract words - split by non-alphabetic characters
    word_pattern = re.split(r"[^A-Za-z]+", input_string)

    # Filter out empty strings and normalize case for word processing
```

```
    words = [word for word in word_pattern if word]  # All words (case-sensitive
    normalized_words = [word.lower() for word in words]  # Words in lowercase

    # Count unique words and calculate word frequencies
    unique_words = set(normalized_words)  # Unique words (case-insensitive)
    word_frequencies = {}  # Frequency with title case
    for word in normalized_words:
        if word in word_frequencies.keys():
            word_frequencies[word] += 1
        else:
            word_frequencies[word] = 1

    # Display results
    print(f"\nExtracted Words: {' '.join(words)}")
    print(f"Number of Words: {len(words)}")
    print(f"Number of unique Words: {len(unique_words)}")
    print("Word Frequencies:")
    for word, count in word_frequencies.items():
        print(f"{word}: {count}")


if __name__ == "__main__":
    main()
```

```
Extracted Words: Hello Python Hello World python is amazing Is it not
Number of Words: 10
Number of unique Words: 7
Word Frequencies:
hello: 2
python: 2
world: 1
is: 2
amazing: 1
it: 1
not: 1
```

In [ ]: