1) Accept a string and display whether it is palindrome or not.

```
s = input("Enter a string: ")
if s == s[::-1]:
    print("Palindrome")
else:
    print("Not Palindrome")
```

2) Accept a string and display it. Now accept slicing positions from and to, slice the string and display it.

```
s = input("Enter a string: ")
start = int(input("Enter start position: "))
end = int(input("Enter end position: "))
print("Original String:", s)
print("Sliced String:", s[start:end])
```

3) Accept a string and display how many vowel characters are there inside it.

```
s = input("Enter a string: ")
vowels = "aeiouAEIOU"
count = sum(1 for ch in s if ch in vowels)
print("Number of vowels:", count)
```

4) Accept a string and display the characters which are repeated in the string.

```
s = input("Enter a string: ")
repeated = []
for ch in set(s):
    if s.count(ch) > 1:
        repeated.append(ch)
print("Repeated characters:", repeated)
```

5) Accept a string and find out how many words are there in it.

```
s = input("Enter a string: ")
words = s.split()
print("Number of words:", len(words))
```

6) Accept a sentence and reverse it.

```
s = input("Enter a sentence: ")
words = s.split()
reversed_sentence = " ".join(reversed(words))
print("Reversed Sentence:", reversed_sentence)
```

7) A pangram is a sentence that contains all the alphabets. Write a function to check a sentence to see if it is a pangram or not.

```
import string

def is_pangram(sentence):
    alphabet = set(string.ascii_lowercase)
    return alphabet <= set(sentence.lower())

s = input("Enter a sentence: ")
if is_pangram(s):
    print("The sentence is a pangram.")
else:
    print("The sentence is not a pangram.")
```