1. Write a Python program to find words which are greater than given length k?

In [5]:
```
def find_words(string, k):
    # split the string into individual words
    words = string.split()

    # create an empty list to store the words greater than k
    result = []

    # iterate through the words and check their length
    for word in words:
        if len(word) > k:
            result.append(word)

    return result

# Example usage:
string = "This is a sample string with words of varying lengths"
k = 5
print(find_words(string, k)) # Output: ['sample', 'string', 'varying', 'lengths']
```

```
['sample', 'string', 'varying', 'lengths']
```

2. Write a Python program for removing i-th character from a string?

In [2]:
```
def remove_char(string, i):
    if i < 0 or i >= len(string):
        # index out of range
        return string
    else:
        return string[:i] + string[i+1:]

# example usage
s = "hello world"
i = 4
s = remove_char(s, i)
print(s)  # prints "helo world"
```

```
hell world
```

3. Write a Python program to split and join a string?

In [3]:
```
# Example string
my_string = "Hello, World! This is a Python program."

# Splitting the string using whitespace as delimiter
split_string = my_string.split()

# Printing the splitted string
print("Splitted string:", split_string)

# Joining the splitted string using space as separator
joined_string = " ".join(split_string)

# Printing the joined string
print("Joined string:", joined_string)
```

```
Splitted string: ['Hello,', 'World!', 'This', 'is', 'a', 'Python', 'program.']
Joined string: Hello, World! This is a Python program.
```

4. Write a Python to check if a given string is binary string or not?

In [6]:
```
def is_binary_string(s):
    """
    This function takes a string as input and returns True if it's a binary string, and False otherwise.
    """
    # Loop through each character in the string
    for char in s:
        # If the character is not '0' or '1', the string is not binary
        if char not in ['0', '1']:
            return False
    # If all characters are '0' or '1', the string is binary
    return True
```

5. Write a Python program to find uncommon words from two strings?

5. Write a Python program to find uncommon words from two Strings?

In [7]:
```
def uncommon_words(string1, string2):
    # Split the two strings into lists of words
    words1 = string1.split()
    words2 = string2.split()

    # Create sets of unique words from each string
    set1 = set(words1)
    set2 = set(words2)

    # Find the words that appear in only one of the sets
    uncommon_words = set1.symmetric_difference(set2)

    # Return the uncommon words
    return uncommon_words

# Example usage
string1 = "the quick brown fox jumps over the lazy dog"
string2 = "the lazy cat sleeps all day"
result = uncommon_words(string1, string2)
print(result)
```

```
{'fox', 'all', 'brown', 'dog', 'day', 'cat', 'over', 'quick', 'jumps', 'sleeps'}
```

6. Write a Python to find all duplicate characters in string?

In [8]:
```
def find_duplicate_characters(string):
    # create an empty dictionary to store character counts
    char_counts = {}

    # loop through each character in the string
    for char in string:
        # if the character is already in the dictionary, increment its count
        if char in char_counts:
            char_counts[char] += 1
        # otherwise, add the character to the dictionary with a count of 1
        else:
            char_counts[char] = 1

    # loop through the dictionary and print any characters with a count greater than 1
    for char in char_counts:
        if char_counts[char] > 1:
            print(char)

# example usage
find_duplicate_characters("hello world")
```

```
l
o
```

7. Write a Python Program to check if a string contains any special character?

In [9]:
```
import re

def contains_special_char(string):
    regex = re.compile('[@_!#$%^&*()<>?/\|}{~:]') # Regular expression to match special characters
    return regex.search(string) is not None

# Example usage
string1 = "Hello World!" # Contains a special character
string2 = "HelloWorld123" # Does not contain any special character

if contains_special_char(string1):
    print("String 1 contains a special character")
else:
    print("String 1 does not contain any special character")

if contains_special_char(string2):
    print("String 2 contains a special character")
else:
    print("String 2 does not contain any special character")
```

```
String 1 contains a special character
String 2 does not contain any special character
```

In [ ]: