Python Basic Programming Assignment 11

Karan Shah

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

```
def find_long_words(string, k):
    # Split the string into a list of words
    words = string.split()

    # Initialize an empty list of results
    results = []

    # Iterate over the list of words
    for word in words:
        # Check if the length of the word is greater than k
        if len(word) > k:
            # If it is, append it to the list of results
            results.append(word)

    # Return the list of results
    return results
string = "The quick brown fox jumps over the lazy dog"
k = 5
print(find_long_words(string, k))  # Output: ['quick', 'brown', 'jumps', 'over', 'lazy']
```

```
[]
```

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

```
def remove_ith_char(string, i):
    # Convert the string to a list of characters
    chars = list(string)

    # Use the del statement to remove the i-th character
    del chars[i]

    # Convert the list of characters back to a string
    return "".join(chars)
string = "The quick brown fox"
i = 4
print(remove_ith_char(string, i))  # Output: "The quick brownox"
```

```
The uick brown fox
```

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

```
string = "The quick brown fox jumps over the lazy dog"
words = string.split()
print(words)  # Output: ['The', 'quick', 'brown', 'fox', 'jumps', 'over', 'the', 'lazy', 'dog']
```

```
['The', 'quick', 'brown', 'fox', 'jumps', 'over', 'the', 'lazy', 'dog']
```

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

```
def is_binary_string(string):
    # Iterate over the characters in the string
    for char in string:
        # If the character is not '0' or '1', return False
        if char != '0' and char != '1':
            return False

    # If all characters are '0' or '1', return True
    return True
string = "01010101"
print(is_binary_string(string))  # Output: True
```

```
string = "01010101a"
print(is_binary_string(string))  # Output: False
```

```
True
False
```

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

```
def find_uncommon_words(string1, string2):
    # Split the strings into lists of words
    words1 = string1.split()
    words2 = string2.split()

    # Create sets from the lists of words
    set1 = set(words1)
    set2 = set(words2)

    # Find the uncommon words by taking the difference between the sets
    uncommon = set1.difference(set2).union(set2.difference(set1))

    # Convert the resulting set back into a list
    uncommon_words = list(uncommon)

    return uncommon_words
string1 = "The quick brown fox jumps over the lazy dog"
string2 = "The quick black cat jumps over the lazy dog"

uncommon_words = find_uncommon_words(string1, string2)
print(uncommon_words)  # Output: ['black', 'brown']
```

```
['brown', 'cat', 'fox', 'black']
```

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

```
def find_duplicate_characters(string):
    # Create an empty dictionary to store the count of each character
    count = {}

    # Iterate over the characters in the string
    for char in string:
        # Increment the count for the character
        count[char] = count.get(char, 0) + 1

    # Create a list to store the duplicate characters
    duplicates = []

    # Iterate over the keys in the dictionary
    for key, value in count.items():
        # If the count is greater than 1, add the key to the list of duplicates
        if value > 1:
            duplicates.append(key)

    return duplicates
string = "Hello, World!"
duplicates = find_duplicate_characters(string)
print(duplicates)  # Output: ['l']
```

```
['l', 'o']
```

✓ 0s completed at 2:00 AM