String Based Practice Questions

Practice Questions

PW SKILLS

TOPIC: String Based Assignment Problem

1. Write a program to reverse a string.

2. Check if a string is a palindrome.

3. Convert a string to uppercase.

4. Convert a string to lowercase.

5. Count the number of vowels in a string.

6. Count the number of consonants in a string.

7. Remove all whitespaces from a string.

8. Find the length of a string without using the `len()` function.

9. Check if a string contains a specific word.

10. Replace a word in a string with another word.

11. Count the occurrences of a word in a string.

12. Find the first occurrence of a word in a string.

13. Find the last occurrence of a word in a string.

14. Split a string into a list of words.

15. Join a list of words into a string.

16. Convert a string where words are separated by spaces to one where words are separated by underscores.

17. Check if a string starts with a specific word or phrase.

18. Check if a string ends with a specific word or phrase.

19. Convert a string to title case (e.g., "hello world" to "Hello World").

20. Find the longest word in a string.

21. Find the shortest word in a string.

22. Reverse the order of words in a string.

23. Check if a string is alphanumeric.

24. Extract all digits from a string.

25. Extract all alphabets from a string.

26. Count the number of uppercase letters in a string.

27. Count the number of lowercase letters in a string.

28. Swap the case of each character in a string.

29. Remove a specific word from a string.

30. Check if a string is a valid email address.

31. Extract the username from an email address string.

32. Extract the domain name from an email address string.

33. Replace multiple spaces in a string with a single space.

34. Check if a string is a valid URL.

35. Extract the protocol (http or https) from a URL string.

36. Find the frequency of each character in a string.

37. Remove all punctuation from a string.

38. Check if a string contains only digits.

39. Check if a string contains only alphabets.

40. Convert a string to a list of characters.

41. Check if two strings are anagrams.

42. Encode a string using a Caesar cipher.

43. Decode a Caesar cipher encoded string.

44. Find the most frequent word in a string.

45. Find all unique words in a string.

46. Count the number of syllables in a string.

47. Check if a string contains any special characters.

48. Remove the nth word from a string.

49. Insert a word at the nth position in a string.

50. Convert a CSV string to a list of lists.

List Based Practice Problem :

1. Create a list with integers from 1 to 10.

2. Find the length of a list without using the `len()` function.

3. Append an element to the end of a list.

4. Insert an element at a specific index in a list.

5. Remove an element from a list by its value.

6. Remove an element from a list by its index.

7. Check if an element exists in a list.

8. Find the index of the first occurrence of an element in a list.

9. Count the occurrences of an element in a list.

10. Reverse the order of elements in a list.

11. Sort a list in ascending order.

12. Sort a list in descending order.

13. Create a list of even numbers from 1 to 20.

14. Create a list of odd numbers from 1 to 20.

15. Find the sum of all elements in a list.

16. Find the maximum value in a list.

17. Find the minimum value in a list.

18. Create a list of squares of numbers from 1 to 10.

19. Create a list of random numbers.

20. Remove duplicates from a list.

21. Find the common elements between two lists.

22. Find the difference between two lists.

23. Merge two lists.

24. Multiply all elements in a list by 2.

25. Filter out all even numbers from a list.

26. Convert a list of strings to a list of integers.

27. Convert a list of integers to a list of strings.

28. Flatten a nested list.

29. Create a list of the first 10 Fibonacci numbers.

30. Check if a list is sorted.

31. Rotate a list to the left by `n` positions.

32. Rotate a list to the right by `n` positions.

33. Create a list of prime numbers up to 50.

34. Split a list into chunks of size `n`.

35. Find the second largest number in a list.

36. Replace every element in a list with its square.

37. Convert a list to a dictionary where list elements become keys and their indices become values.

38. Shuffle the elements of a list randomly.

39. Create a list of the first 10 factorial numbers.

40. Check if two lists have at least one element in common.

41. Remove all elements from a list.

42. Replace negative numbers in a list with 0.

43. Convert a string into a list of words.

44. Convert a list of words into a string.

45. Create a list of the first `n` powers of 2.

46. Find the longest string in a list of strings.

47. Find the shortest string in a list of strings.

48. Create a list of the first `n` triangular numbers.

49. Check if a list contains another list as a subsequence.

50. Swap two elements in a list by their indices.

Tuple Based Practice Problem :

1. Create a tuple with integers from 1 to 5.

2. Access the third element of a tuple.

3. Find the length of a tuple without using the `len()` function.

4. Count the occurrences of an element in a tuple.

5. Find the index of the first occurrence of an element in a tuple.

6. Check if an element exists in a tuple.

7. Convert a tuple to a list.

8. Convert a list to a tuple.

9. Unpack the elements of a tuple into variables.

10. Create a tuple of even numbers from 1 to 10.

11. Create a tuple of odd numbers from 1 to 10.

12. Concatenate two tuples.

13. Repeat a tuple three times.

14. Check if a tuple is empty.

15. Create a nested tuple.

16. Access the first element of a nested tuple.

17. Create a tuple with a single element.

18. Compare two tuples.

19. Delete a tuple.

20. Slice a tuple.

21. Find the maximum value in a tuple.

22. Find the minimum value in a tuple.

23. Convert a string to a tuple of characters.

24. Convert a tuple of characters to a string.

25. Create a tuple from multiple data types.

26. Check if two tuples are identical.

27. Sort the elements of a tuple.

28. Convert a tuple of integers to a tuple of strings.

29. Convert a tuple of strings to a tuple of integers.

30. Merge two tuples.

31. Flatten a nested tuple.

32. Create a tuple of the first 5 prime numbers.

33. Check if a tuple is a palindrome.

34. Create a tuple of squares of numbers from 1 to 5.

35. Filter out all even numbers from a tuple.

36. Multiply all elements in a tuple by 2.

37. Create a tuple of random numbers.

38. Check if a tuple is sorted.

39. Rotate a tuple to the left by `n` positions.

40. Rotate a tuple to the right by `n` positions.

41. Create a tuple of the first 5 Fibonacci numbers.

42. Create a tuple from user input.

43. Swap two elements in a tuple.

44. Reverse the elements of a tuple.

45. Create a tuple of the first `n` powers of 2.

46. Find the longest string in a tuple of strings.

47. Find the shortest string in a tuple of strings.

48. Create a tuple of the first `n` triangular numbers.

49. Check if a tuple contains another tuple as a subsequence.

50. Create a tuple of alternating 1s and 0s of length `n`.

Set Based Practice Problem :

1. Create a set with integers from 1 to 5.

2. Add an element to a set.

3. Remove an element from a set.

4. Check if an element exists in a set.

5. Find the length of a set without using the `len()` function.

6. Clear all elements from a set.

7. Create a set of even numbers from 1 to 10.

8. Create a set of odd numbers from 1 to 10.

9. Find the union of two sets.

10. Find the intersection of two sets.

11. Find the difference between two sets.

12. Check if a set is a subset of another set.

13. Check if a set is a superset of another set.

14. Create a set from a list.

15. Convert a set to a list.

16. Remove a random element from a set.

17. Pop an element from a set.

18. Check if two sets have no elements in common.

19. Find the symmetric difference between two sets.

20. Update a set with elements from another set.

21. Create a set of the first 5 prime numbers.

22. Check if two sets are identical.

23. Create a frozen set.

24. Check if a set is disjoint with another set.

25. Create a set of squares of numbers from 1 to 5.

26. Filter out all even numbers from a set.

27. Multiply all elements in a set by 2.

28. Create a set of random numbers.

29. Check if a set is empty.

30. Create a nested set (hint: use frozenset).

31. Remove an element from a set using the discard method.

32. Compare two sets.

33. Create a set from a string.

34. Convert a set of strings to a set of integers.

35. Convert a set of integers to a set of strings.

36. Create a set from a tuple.

37. Convert a set to a tuple.

38. Find the maximum value in a set.

39. Find the minimum value in a set.

40. Create a set from user input.

41. Check if the intersection of two sets is empty.

42. Create a set of the first 5 Fibonacci numbers.

43. Remove duplicates from a list using sets.

44. Check if two sets have the same elements, regardless of their count.

45. Create a set of the first `n` powers of 2.

46. Find the common elements between a set and a list.

47. Create a set of the first `n` triangular numbers.

48. Check if a set contains another set as a subset.

49. Create a set of alternating 1s and 0s of length `n`.

50. Merge multiple sets into one.

NOTE: These are the practice problems created for you so that you can practice more and more and deal with the logic building. Solution of these problems will be provided to you by next monday, till then try to solve these problems as much as you can. There is no deadline to submit this assignment. You can take your own time and try to solve it.