Part I

1. **Basic Indexing**:
   Given the list numbers = [10, 20, 30, 40, 50], what is the value at index 2?

2. **Negative Indexing**:
   Using the list fruits = ['apple', 'banana', 'cherry', 'date', 'elderberry'], what element is accessed using the index -1?

3. **Slicing a List**:
   Given the list colors = ['red', 'green', 'blue', 'yellow', 'orange'], how would you slice the list to get ['green', 'blue', 'yellow']?

4. **Slicing with Step**:
   For the string alphabet = 'abcdefghijklmnopqrstuvwxyz', write a slice expression to get every third letter starting from index 1.

5. **Slicing a Tuple**:
   Given the tuple coordinates = (4, 3, 2, 1, 0), how can you extract the sub-tuple (3, 2, 1)?

6. **Modifying List through Indexing**:
   Using the list fruits = ['apple', 'banana', 'cherry', 'date'], change the second element to 'blueberry' using indexing.
7. **Using Indexing with Strings**:
   Given the string text = "Python is awesome!", how would you extract the substring "is" using indexing?

8. **Slicing a List from Start to End**:
   For the list numbers = [1, 2, 3, 4, 5], how would you slice the entire list using a slice notation?

Edit with WPS Office

9. **Slicing with Negative Indices**:
   Using the list `months = ['January', 'February', 'March', 'April', 'May', 'June']`, how would you slice the list to get the last three months?

10. **Combining Indexing and Slicing**:
    Given the list `values = [10, 20, 30, 40, 50]`, how would you access the second-to-last element, and how would you slice out the last two elements?

Part II

1. **Slicing a List of Lists**:
   Given the list `matrix = [[1, 2], [3, 4], [5, 6], [7, 8]]`, how would you extract a sublist containing only the second and third inner lists?

2. **Advanced Step Slicing**:
   For the string `text = "abcdefghijklmnopqrstuvwxyz"`, how would you slice the string to get the characters starting from index 2 and ending at index 18, with a step of 4?

3. **Slicing with Ellipsis**:
   Given the 3D list `data = [[[1, 2], [3, 4]], [[5, 6], [7, 8]], [[9, 10], [11, 12]]]`, use slicing and ellipsis to extract all the inner lists from the second and third outer lists.

4. **Slicing with None**:
   How would you use `None` in slicing to extract a sublist starting from index 2 to the end, and every second element after that, from the list `numbers = [1, 2, 3, 4, 5, 6, 7, 8, 9]`?

5. **Slicing in Reverse**:
   Given the list `letters = ['a', 'b', 'c', 'd', 'e']`, write a slice that returns the list in reverse order, but without using the `reversed()` function or list slicing with a negative step.

Edit with WPS Office

6. **Nested List Indexing**:
   Consider the nested list `nested = [[1, 2, [3, 4]], [5, 6, [7, 8]]]`. How would you extract the number `7` from this list using indexing?

7. **Slicing and Indexing on a String with Multiple Occurrences**:
   Given the string `phrase = "Hello world, welcome to Python!"`, how would you extract the substring `"world"` using slicing and indexing (assuming only one occurrence)?

8. **Handling IndexError in Slicing**:
   Given the list `data = [10, 20, 30]`, how would you safely slice the list to get the last 5 elements without causing an `IndexError`?

9. **Multi-Dimensional Array Slicing**:
   For a NumPy array `arr = np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])`, write a slice to extract the last column from the 2D array.

10. **Reassigning Slices in Nested Lists**:
    Consider the nested list `matrix = [[10, 20], [30, 40], [50, 60]]`. How would you change the second element of the second inner list to `99` using indexing and slicing?

Part III

1. **Extract Middle Element from List**: Write a Python function that takes a list of odd length and returns the middle element. If the list has an even length, return the two middle elements.

2. **Reversing a String Without Built-in Functions**: Write a Python function that takes a string and returns it in reverse order using slicing (do not use the `reversed()` function).

Edit with WPS Office

3. **Remove Duplicates Using Slicing**: Write a Python function that takes a list and returns a new list with duplicates removed, keeping the original order of elements.

4. **Extract Every N-th Element from a List**: Write a Python function that takes a list and an integer `n`, and returns a new list containing every `n`-th element from the original list.

5. **Check if a String is a Palindrome**: Write a Python function that takes a string and returns `True` if the string is a palindrome (reads the same forward and backward), using slicing.

6. **Extract a Sublist Between Two Indices**: Write a Python function that takes a list and two indices (`start` and `end`) and returns a sublist containing elements from `start` index to `end` index (inclusive).

7. **Create a Copy of a List Using Slicing**: Write a Python function that takes a list and returns a copy of that list using slicing.

8. **Find the Second Largest Element in a List**: Write a Python function that takes a list of numbers and returns the second largest element using slicing and indexing.

9. **Rotate a List Left by N Elements**: Write a Python function that rotates a list `lst` to the left by `n` elements using slicing.

Edit with WPS Office

10. **Replace Elements in a Range of Indices**: Write a Python function that takes a list, two indices (start and end), and a new list of elements. Replace the sublist between the two indices with the new list.

Edit with WPS Office