part-5

Looping Techniques in Python

Using For Loops

- For loops in Python are used to iterate over items or elements in a sequence (e.g.,list,tuple,dictionary, string)
- The basic syntax for a for loop is:
  ```
  for iterator_variable in sequence:    # statements to be executed
  ```
- You can use the `end` parameter in the `print()`function to specify the character to be printed at the end of the output
- You can iterate over a list using a for loop
- You can calculate the average of a list of numbers using a for loop
- You can iterate over a tuple using a for loop
- You can use the `range()` function in a for loop to iterate over a range of numbers
- You can use a nested for loop to iterate over a 2D list or a list of dictionaries
- You can use the `else` clause with a for loop to execute a block of code after the loop has completed
- You can use the `break` statement to exit a for loop before it has completed all iterations
- You can use the `continue` statement to skip the current iteration of a for loop and move to the next iteration

Using While Loops

- While loops in Python are used to execute a block of code as long as a certain condition is true
- The basic syntax for a while loop is:
  ```
  while condition:    # statements to be executed
  ```
- You can use a while loop to accept positive numbers from the user and calculate the average
- You can use a while loop to guess a random number
- You can use a while loop to remove elements from the end of a list
- You can use a while loop with the `else` clause to execute a block of code after the loop has completed
- You can use a while loop to print a sequence of numbers in a single line

Sorting and Filtering

- You can use the `sorted()` function with a for loop to sort the elements in a list in ascending or descending order
- You can use the `sorted()`function with the `reverse=True` parameter to sort the elements in a list in descending order
- You can use the `sorted()`function to sort the keys or values in a dictionary
- You can use the `filter()` function with a lambda function to filter elements in a list based on a specific condition

Enumerate and Zip Functions

- The `enumerate()` function in Python assigns an index to each item in a sequence, returning a tuple of the index and th e item
- You can use the `start` parameter in the `enumerate()` function to specify the starting index
- The `zip()`function in Python creates an iterator of tuples, where each tuple contains the corresponding elements fro m each iterable

Table

Looping Technique	Description
For Loop	Used to iterate over items or elements in a sequence
While Loop	Used to execute a block of code as long as a certain condition is true
Sorting	Use `sorted()` function to sort elements in a list or dictionary
Filtering	Use `filter()` function with a lambda function to filter elements in a list
Enumerate	Assigns an index to each item in a sequence, returning a tuple of the index and the item
Zip	Creates an iterator of tuples, where each tuple contains the corresponding elements from each ite rable