Python Coding Interview Questions with Examples

# Python Lists

1. Reverse a list in Python. Example: Given the list [1, 2, 3], after reversing, it should become [3, 2, 1].
2. Find the sum of elements in a list. Example: For [1, 2, 3], the sum is 6.
3. Check if a list is empty or not. Example: [] should return True, whereas [1] returns False.
4. Find the number of occurrences of an element in a list. Example: In [1, 1, 2, 3], the number of occurrences of 1 is 2.
5. Merge two sorted lists into a new sorted list. Example: Merging [1, 3, 5] and [2, 4, 6] should result in [1, 2, 3, 4, 5, 6].

# Python Dictionaries

1. Count Characters in a String. Example: 'hello' results in {'h': 1, 'e': 1, 'l': 2, 'o': 1}.
2. Merge Two Dictionaries. Example: {'a': 1, 'b': 2} and {'b': 3, 'c': 4} becomes {'a': 1, 'b': 3, 'c': 4}.
3. Invert a Dictionary. Example: {'a': 1, 'b': 2} becomes {1: 'a', 2: 'b'}.
4. Find the Key with the Maximum Value. Example: {'a': 100, 'b': 200} should return 'b'.
5. Group Anagrams. Example: ['eat', 'tea', 'tan', 'ate', 'nat', 'bat'] groups into [['eat', 'tea', 'ate'], ['tan', 'nat'], ['bat']].

# Python Tuples and Sets

1. Swap Two Elements in a Tuple. Example: Swapping indices 1 and 3 in (1, 2, 3, 4) results in (1, 4, 3, 2).
2. Count the Elements in a Tuple. Example: In (1, 2, 2, 3), 2 occurs twice.
3. Convert a List of Tuples into a Dictionary. Example: [('a', 1), ('b', 2)] becomes {'a': 1, 'b': 2}.
4. Find the Symmetric Difference Between Two Sets. Example: {1, 2, 3} and {3, 4, 5} results in {1, 2, 4, 5}.
5. Check if a Set is a Subset of Another Set. Example: {1, 2} is a subset of {1, 2, 3}.

# Python Methods

1. String Capitalization: Write a method that capitalizes the first letter of each word in a string. Example: 'hello world' becomes 'Hello World'.
2. Finding Unique Elements in a List: Write a method that returns a new list with unique elements of the first list. Example: [1,2,2,3] becomes [1,2,3].
3. Sorting a List of Tuples by the Second Item: Write a function that takes a list of tuples and returns a list sorted by the second item in each tuple. Example: [(1, 3), (3, 2), (2, 1)] becomes [(2, 1), (3, 2), (1, 3)].
4. Reverse Dictionary Mapping: Write a function that inverts a dictionary (swaps keys and values). Example: {'a': 1, 'b': 2} becomes {1: 'a', 2: 'b'}.

# Python Classes

1. Define a Basic Class: Define a class `Circle` with a method `area` that calculates the area of the circle.
2. Class Inheritance: Create a class `Cylinder` that inherits from `Circle` and has a method to calculate the volume.
3. Method Overriding: Override a method in the `Cylinder` class to adjust the volume calculation.
4. Encapsulation: Modify the `Circle` class so that its radius is a private attribute.
5. Static and Class Methods: Add a static method to the `Circle` class that calculates the circumference of a circle given its radius.

# Python Exception Handling

1. Basic Try-Except Block: Write a function that handles a division by zero error.
2. Handling Multiple Exceptions: Modify the above function to handle multiple types of exceptions, including division by zero and type errors.
3. Using the Else Clause: Include an `else` clause in the try-except block to execute code when no exceptions are raised.
4. Implementing Finally: Add a `finally` clause to the try-except block to execute cleanup code.
5. Raising Custom Exceptions: Define a custom exception and demonstrate raising it within a function.