1. Multiple Inheritance and Method Resolution Order

Create a Python class structure with multiple inheritance, similar to the **Father**, **Mother**, **Child** example, but with different classes and attributes. Ensure that your subclass inherits from two parent classes. Demonstrate how Python's Method Resolution Order (MRO) works in your example.

2. Global vs Local Variables

Write a Python script where:

* You define a global variable.
* You create a function that attempts to modify this global variable without using the **global** keyword, and observe what happens.
* Then, modify the same global variable within a function using the **global** keyword.

3. Set Operations

Given two sets:

* **set\_a = {10, 20, 30, 40, 50}**
* **set\_b = {30, 40, 60, 70}**

Perform the following operations:

* Union
* Intersection
* Difference (both **set\_a - set\_b** and **set\_b - set\_a**)
* Symmetric Difference

Explain the results of each operation.

4. Tuple Manipulations

Given a tuple **tuple\_a = (100, 200, 300)**, perform the following:

* Access the first and last elements.
* Unpack the tuple into three variables.
* Concatenate another tuple **(400, 500)** to **tuple\_a**.
* Repeat **tuple\_a** twice.

Explain the immutability of tuples in the context of these operations.

5. Class Attributes and Instance Attributes

Create a class **Vehicle** with a class attribute **category** set to 'Transport'. Then:

* Create an instance attribute **name** in the **\_\_init\_\_** method.
* Create a method to display both the class attribute and the instance attribute.
* Create an instance of **Vehicle** and call the display method.

Discuss the difference between class attributes and instance attributes with examples.

6. Exception Handling in Inheritance

Extend the multiple inheritance example by adding a method that could potentially raise an exception. Implement exception handling in this method. Demonstrate how this method can be called safely in the subclass.

**Submission Requirements**

For each problem:

* Write the Python code required to solve the problem.
* Include comments in your code to explain your logic.
* Provide a brief explanation of your solution and any Python concepts used.