Assignment 1 - Exploring Distributed Databases through Microservices

Part 1: Theoretical Understanding (20 points)

1. Provide a brief explanation of Distributed Databases and their significance in modern applications.
2. Compare and contrast different distributed database architectures (eg: Homogeneous vs. Heterogeneous, Client-Server vs. Peer-to-Peer)
3. Compare and contrast different techniques for managing data in distributed databases (e.g., replication, sharding, partitioning).
4. Explain the concepts of data consistency, data availability, and data partitioning in the context of distributed databases.

Part 2: Practical Implementation (80 points)

Task 1: Setting Up Microservices (30 points)

1. Choose a programming language and framework for implementing microservices (e.g., Node.js with Express, Spring Boot, Django, etc.).
2. Create three separate microservices (Microservice A, B, and C), each representing a different functionality.
    a. Microservice A - User Management
    b. Microservice B - Order Placement
    c. Microservice C - Inventory Management
3. Implement RESTful APIs for each microservice, including endpoints for basic CRUD operations.
    a. Microservice A - Create, Update, View, Delete Users
    b. Microservice B - Create, Update, View, Delete Orders
    c. Microservice C - Create, Update, View, Delete Products

Task 2: Database Integration (40 points)

1. Select three different types of databases (e.g., MySQL, MongoDB, Microsoft SQL Server, DynamoDB) for each microservice.
2. Configure the databases for each microservice and establish connections.
3. Implement database operations for each microservice's functionality (Create, View, Update, Delete).

Task 3: Microservices Communication (10 points)

- Implement communication mechanisms between the microservices.
- Ensure that microservices can exchange relevant data and maintain consistency across their interactions.

Submission Guidelines:

Prepare a report that includes:

1. **Theoretical explanations of distributed databases and microservices architecture (Part 1).**

2. **Detailed documentation of the implementation process, including code snippets, for each microservice (Part 2).**

3. **A summary of how microservices communicate and exchange data (Task 3).**

4. **Postman script to test the workflow.**

5. **Submit the report along with the source code of the microservices on the designated platform (e.g., GitHub, GitLab).**