Test Task: Generate Combinations and Store in MySQL

Objective

Create an API using Node.js and MySQL to generate combinations from a list of items and store them in the database. The API should respect a rule where items starting with the same letter cannot be combined.

Problem Description

You are given a list of items of different types. Each item type is identified by a prefix letter, and items with the same prefix (starting letter) cannot be selected together in a combination. Your task is to build an API that receives a list of numbers representing item types and a required combination length, generates all valid combinations, stores them in a MySQL database, and returns the results in the response.

Example Input

You are given an input array like `[1, 2, 1]`, which corresponds to items `A1`, `B1`, `B2`, `C1`. The condition is that no two items with the same starting letter (prefix) can be part of the same combination.

For a combination length of 2, the valid combinations would be:

```
["A1", "B1"], ["A1", "B2"], ["A1", "C1"],
["B1", "C1"], ["B2", "C1"]
```

API Task

Create a POST request that receives an input array (like `[1, 2, 1]`) and a combination length (e.g., 2). The API should generate valid combinations and store them in a MySQL database. Each combination should be associated with a unique ID.

Requirements

Request:

- POST `/generate`
- Body:

```
{
  "items": [1, 2, 1],
  "length": 2
}
```

Database Schema

- MySQL (without ORMs)
- Insertions should be done using MySQL transactions.
- The following tables should be created:
  - `items`: Stores the items (e.g., `A1`, `B1`, etc.).
  - `combinations`: Stores the generated combinations with their unique IDs.
  - `responses`: Stores the responses sent to the client.

Response

The API should return the stored combinations with their unique IDs in the following format:

```
{
    "id": 1,
    "combination": [
        ["A1","B1"], ["A1", "B2"], ["A1", "C1"],
        ["B1", "C1"], ["B2", "C1"]
    ]
}
```

Rules

- Items with the same starting letter (e.g., `A1` and `A2`) cannot be combined together.
- Insertions into the database must be done using MySQL transactions to ensure consistency.

Evaluation Criteria

1. **Correctness**: The API should return the correct combinations and store them in the database.
2. **Database Design**: Ensure that the items, combinations, and responses are stored in separate tables.
3. **Use of Transactions**: Ensure that insertions are wrapped in MySQL transactions.
4. **Code Quality**: Code should be clean, modular, and well-documented.
5. **Efficiency**: Handle database operations and combination generation efficiently.

Additional Information

- You can use mysql2 for MySQL connections.
- Do not use any ORM stick to raw SQL queries.