Backend Developer Homework Assignment

Objective

Build an API-only Rails application to manage companies and their addresses. The focus is on demonstrating your understanding of Rails 8.0, advanced validations, and building APIs with JSON responses.

Requirements

Project Overview

You will build a RESTful API with the following features:

- A 'Company' can have multiple 'Addresses'
- Add a company with multiple addresses in a single API request
- Bulk import multiple companies (with addresses) from a CSV file via a dedicated endpoint
- Validate the input and handle errors gracefully

Project Setup

Use:

- Ruby 3.3
- Rails 8.0
- Minitest or RSpec for automated testing

Data Models

Company

1. 'name' (string, required, max length 256)
2. 'registration_number' (number, required, unique)

Address

1. 'street' (string, required)
2. 'city' (string, required)
3. 'postal_code' (string, optional)
4. 'country' (string, required)

Relationships: A 'Company' has one or many 'Addresses'.
There are no 'Addresses' without 'Company'.

Endpoints

- Create a Company with Multiple Addresses
- Bulk Import Companies via CSV
- Input: A CSV file with the following structure:

```
'''name,registration_number,street,city,postal_code,country
Example Co,123456789,123 Main St,New York,10001,USA
Example Co,123456789,456 Elm St,Los Angeles,90001,USA
Another Co,987654321,789 Oak St,Chicago,60601,USA
'''
```

Requirements:

- The CSV may contain multiple rows for the same company (with different addresses).
- Parse the CSV and create companies with their associated addresses.
- Validate the input (e.g., no duplicate 'registration_number').
- Response (Success): return imported companies.
- Error Handling: Return meaningful error messages when validations fail.

Testing

- Write automated tests (Minitest or RSpec) for:
- Creating a company with addresses.
- Bulk importing companies from CSV.

Delivery

Submit your code as a public git repository (GitHub, Gitlab etc.).

Notes

- Focus on clean, readable code and clear API responses
- Use vanilla Rails or any gem you want
- Use "Rails Way" or not
- If something in task is not clear, then select the simplest solution in your opinion
- If you want to show off some additional skills or stand out when solving a task - feel free to do so
- The task was designed to last approximately 60 minutes