



# DAY 3 - API INTEGRATION AND DATA MIGRATION (e-commerce)

Muhammad Sarim

## Overview

The work I accomplished on Day 3, highlighting the steps I took for API integration, data migration to Sanity, and the creation of dynamic pages.

**[API to Sanity Data Migration Documentation](#)**

## Day 3: API Integration and Data Migration to Sanity

### 1. Receiving the API:

- On Day 3, I received the API designed according to my template. This API contained the complete data required for my project.
- I started by thoroughly reading and understanding the structure and content of the API.

### 2. Migration Script:

- After analyzing the API, I reviewed the migration script provided. This script was essential for transferring the data from the API to Sanity.
- Before proceeding, I ensured my understanding of how the script interacted with both the API and Sanity.

### 3. Error Resolution During Migration:

- During the data migration process, I encountered several errors (at least 5 to 6 major issues). These errors ranged from data format mismatches to connectivity issues.
- Debugging and resolving these errors took significant time and effort (around 5-6 hours). This process involved revisiting the migration script, debugging data processing logic, and adjusting configurations in Sanity.
- Despite the challenges, I successfully migrated the data from the API to Sanity by the end of the day. This marked a major milestone in the project.

### Cleanup and Fetching Data from Sanity:

- I began the day by reviewing my setup files to ensure a clean and efficient development environment.
- I deleted all dummy data that was previously used for testing purposes.
- I fetched live data directly from Sanity, ensuring the project was now running with actual data.

### Creating Cards and Handling Data:

- Using the fetched data, I created visually appealing and functional cards for the products. These cards displayed product information dynamically based on the data retrieved from Sanity.
- Throughout this process, I encountered and resolved several minor errors to ensure the cards were functioning correctly.

### Comprehensive Data Management:

- I carefully reviewed and double-checked all the data to ensure accuracy and consistency.
- A detailed list of data items was created to keep track of all the products and their attributes.

### Dynamic Page Creation:

- Dynamic pages for the products were successfully implemented. Each product now had its own dedicated page displaying detailed information.
- This involved setting up routes, integrating Sanity's dynamic data fetching capabilities, and ensuring seamless navigation between pages.

### Final Review:

- I conducted a thorough review of the entire setup to ensure everything was functioning as expected.
- By the end of Day 4, the project had transitioned from dummy data to fully functional dynamic pages powered by Sanity, laying the groundwork for further development.

---

### Conclusion:

- The transition from API to Sanity was a challenging but rewarding process that involved solving multiple errors, optimizing the setup, and successfully migrating live data.
- Dynamic product pages were implemented, and the entire project's data flow was streamlined for efficiency and accuracy.

- 
- The experience provided valuable insights into API integration, data migration, error resolution, and dynamic page development.