

✓ Day 3: Environments & Variables – Reuse & Simplify

👤 Created by: Deep Ghinaiya

💼 Software Developer

🔗 [linkedin.com/in/deep-ghinaiya](https://www.linkedin.com/in/deep-ghinaiya)

📌 What You'll Learn:

- What are environments in Postman
- How to create and manage environments
- Use of environment, global, and local variables
- Dynamically switch between development, staging, and production setups
- Reduce repetition with variables in headers, bodies, and URLs

🌍 What Are Environments?

Environments allow you to manage different sets of variables (like base URLs, auth tokens, IDs) depending on your API stage — development, staging, or production.

Example:

{{base_url}} → https://api.example.com

{{auth_token}} → Bearer yourToken123

🔧 How to Set Up an Environment:

1. Click the **Environments** icon (top right gear icon) → **Manage Environments**
2. Click **Add**
3. Name your environment (e.g., “Dev”)
4. Add key-value pairs like:
 - base_url: https://jsonplaceholder.typicode.com

- user_id: 1

5. Click **Add**

Now in your request, you can use:

`{{base_url}}/posts/{{user_id}}`

🌟 Types of Variables:

- **Global:** Used across all collections and environments
- **Environment:** Scoped to specific environments
- **Collection:** Scoped within a collection
- **Local:** Temporary in a single request

Example Setup:

Request URL:

`{{base_url}}/posts/{{post_id}}`

Header:

Authorization: Bearer `{{auth_token}}`

Pro Tips:

- ✓ Use `{{variable}}` syntax anywhere: URL, headers, body
- ✓ Hit `{{` in a field to auto-suggest your defined variables
- ✓ Switch environments from the top-right dropdown
- ✓ Helps avoid hardcoded values and keeps requests clean

Full Series:

 github.com/Ghinaiya-Deep/Skill-Stack-by-DeepLogic-Labs

👏 If this helped, star 🌟 the repo & share with your tech circle!