

❤️ JavaScript Learning Journey Continues!

Today I explored **Callback Functions** — one of the most important concepts in JavaScript, especially when handling asynchronous operations like API calls, delays, and event-driven tasks. ↗

❖ Focus of the Day:

- What callbacks are and why we use them
- Passing functions as arguments
- Using callbacks in setTimeout()
- Understanding callback flow
- Using callbacks with array methods like forEach()
- Building custom callback functions

🔍 What I Learned:

- JavaScript treats functions as first-class citizens
- Callbacks help control the order in which tasks run
- setTimeout() is a simple but powerful asynchronous example
- Callbacks are the foundation of promises and async/await
- Understanding them improves my grasp of real asynchronous code

💬 Reflection:

Day 26 completed!

Callbacks helped me understand how JavaScript handles tasks that don't run immediately.

This concept is key for APIs, events, timers, and modern JS frameworks — a major step forward! 🎉

🎯 Next Target (Day 27):

➡️ Learn **Arrow Functions** — a cleaner, modern, and more concise way to write functions.

#JavaScript #Programming #WebDevelopment #100DaysOfCode #LearningInPublic #FrontendDeveloper
#CodingJourney