**What is NGIX**

**Steps**

Could start by reading the NGINX source code. So study the source code of NGINX itself, understanding how it works, is structured and how it handles various tasks such as HTTP request processing, event handling, and condiguration parsing as this will be valukbale insights for building your own server

Make sure you know C very well.

Familarize yourself with socket programming in C as it forms the foundation of building network server like NGINX.

Study [NET.PDF (cornell.edu)](https://www.cs.cornell.edu/~kvikram/HTMLS/MLA/NET.PDF)

Also study the HTTP protocol specification (HTTP/1.1 or HTTP/2) to understand how web servers communicate with clients. The RFC documents for HTTP are available online and provides detailed information about request and response formats, status codes, headers and more.

Event Driven programming – So NGINX is known for its efficient event driven architecture. Learn about even driven programming concepts and techniques such as epoll (On Linux) or Kqueue (on BSD) which are used for handling I/O events effieicenlt.

Concurrency and Multithreading – NGINX employs a highly concurrent and asynchronous model for handling multiple client connetinos simultaneously. Study concurrency and multithreading concepts to understand how to implement a similar model in your sevre

Memory management – NGINX is optimised for performance, incudling efficient memory management. Lean about memory allocation and management techniques in C as well as strategies for minimizing memory overhead and avoiding memory leaks

Building prototyoe – Start by building a simple prototype of your server, focusing on basic functionality such as accepting incoming connectinos, parsing HTTP requestings, and serving static conent. You can add more featuresa and optimizations. Explore open source web server written in C such as Lighthttpd or Cherokee to see how other developers have implemented similar functionality.