### Lesson 10: Introduction to Network Programming

* **Reading**: “The Rust Programming Language” for basic networking concepts, supplemented with targeted readings from “Programming Rust” by Jim Blandy on advanced network programming.
* **Assignments**: Build a chat server and client that communicate over TCP. Focus on handling multiple client connections and broadcasting messages.
* **Preflight**: Familiarize with the std::net module in Rust by creating simple TCP socket connections.
* **Lesson Goals**:
  + Introduce the fundamentals of network programming in Rust, including TCP and UDP socket programming.
  + Understand how to manage client-server connections and data transfer.
  + Explore asynchronous I/O in network programming to handle multiple clients efficiently.
* **Motivation**: Network programming is a cornerstone of many systems and applications, and Rust offers powerful, safe tools to build robust networked services.
* **Lecture**:
  + Basics of TCP/UDP networking, including socket creation, connection, and data transfer.
  + Techniques for handling multiple clients using non-blocking I/O or async/await patterns.
  + Introduction to higher-level networking crates available in the Rust ecosystem.
* **Lab**: Develop the chat application, with emphasis on asynchronous network communication and managing client states effectively.