1. **وارد کردن کتابخانه‌ها**:
   * std::io::{Read, Write}: برای خواندن و نوشتن داده‌ها از/به TCP Stream.
   * std::net::{TcpListener, TcpStream}: برای ایجاد و مدیریت اتصالات TCP.
   * std::thread: برای ایجاد تردها (رشته‌های پردازشی) جداگانه جهت مدیریت هر کلاینت به صورت موازی.
2. **تابع handle\_client**:
   * این تابع وظیفه پردازش داده‌های دریافتی از کلاینت‌ها را دارد.
   * داده‌های دریافتی از کلاینت (با استفاده از متد read) در یک بافر ذخیره می‌شوند.
   * سپس داده‌ها به همان صورت (بازتاب داده می‌شوند) به کلاینت بازمی‌گردد (write\_all).
   * در صورت قطع اتصال یا خطا، حلقه پایان می‌یابد.
3. **تابع main**:
   * یک سرور TCP بر روی آدرس 127.0.0.1:7878 راه‌اندازی می‌کند.
   * حلقه بی‌نهایت برای پذیرش اتصالات ورودی از کلاینت‌ها (با استفاده از listener.incoming()).
   * وقتی یک کلاینت به سرور متصل می‌شود، یک ترد جدید با فراخوانی تابع handle\_client برای پردازش داده‌ها ایجاد می‌شود.
4. **ویژگی‌های مهم**:
   * سرور به صورت **echo server** عمل می‌کند؛ یعنی هر داده‌ای که از کلاینت دریافت کند، دقیقاً همان را به کلاینت بازمی‌گرداند.
   * برای هر کلاینت جدید یک **ترد جدید** ایجاد می‌شود تا بتواند به صورت همزمان چندین کلاینت را پردازش کند.

این سرور به طور ساده و ابتدایی به درخواست‌ها پاسخ می‌دهد و می‌تواند به صورت موازی با چندین کلاینت ارتباط برقرار کند.