📅 Day 2 — HTML URL Extraction + DFS Traversal Function

User Story

As a **developer**, I want to **extract all valid HTML URLs from a given HTML document and traverse them recursively using a DFS approach**, so that I can explore a website's structure up to a specified depth and link count limit.

Acceptance Criteria

1. Extract all `<a href="...">` URLs from a raw HTML string **without using external libraries**.
2. Skip URLs pointing to **non-HTML resources** (e.g., `.pdf`, `.jpg`, `.png`, `.css`, `.js`).
3. Normalize **relative URLs**:
4. Handle `./`, `../`, and root-relative (`/path`) formats.
5. Implement a recursive `crawl(url, depth)` function that:
6. Fetches and parses HTML (local stubbed content allowed for now).
7. Extracts links from the HTML.
8. Calls itself recursively for each extracted link.
9. The crawler must:
10. Respect **maximum depth**.
11. Respect **maximum total link count**.
12. No deduplication of URLs is required at this stage.

Outcome

- A working DFS-based link traversal tool that:
- Extracts valid HTML links.
- Traverses them recursively.
- Respects depth and link count limits.
- Basic URL normalization implemented.
- No deduplication logic yet.