Assignment 2: Web Search

Name : Sohaib Ahmed

CMS ID : 373937

***Crawler Details:***

**URL Queue:** The URL queue functionality is embedded in the global variable $startUrl, which represents the seed URL. The crawlLinks function is responsible for crawling URLs from the queue.

**Crawling**: The crawlLinks function sends HTTP requests to the URLs in the queue, retrieves HTML content using file\_get\_contents, and saves it in the output.json file.

**HTML Parsing:** HTML parsing is achieved using the DOMDocument class in PHP. The loaded HTML content is processed using DOMDocument to extract information such as title, headings, paragraphs, spans, anchors, list items, table cells, labels, and buttons.

**URL Extraction**: URL extraction is part of the crawling process, where hyperlinks (URLs) are extracted from the crawled HTML using DOMDocument, and valid URLs are added to the URL queue for further crawling.

**Depth Limit:** The depth limit is controlled by the $depth variable, which is decremented with each recursive call to crawlLinks. The function stops recursive crawling when the depth limit is reached.

**Output:** The extracted information, including URLs, titles, and various HTML elements, is stored in the output.json file using file\_put\_contents.

**Content Search Module:** The search functionality is provided by the searchStringInJson and searchStringInJsonObject functions. They search for a specified string within the crawled content and display or log the URLs that contain the search string along with the matched content.

**Robots.txt Compliance:** The robots.txt compliance is checked by retrieving and parsing the robots.txt file using the parseRobotsTxt function. Disallowed paths are extracted and respected during the crawling process.

**Error Handling:** Error handling is implemented using try-catch blocks in the crawlLinks function. It catches exceptions and displays an error message when issues arise during crawling.

**Bonus Things:**

**Filtering:** Filtering is not explicitly implemented in the provided code. However, you can extend the isDisallowed function to include additional criteria for excluding certain URLs from crawling.

**Persistent Storage:** Persistent storage is achieved by storing the crawled data in the output.json file. For more advanced persistent storage, you can consider using a local database or other file formats.

**Advanced Search Features:** The current search module is basic and looks for an exact match. To implement more advanced search features, you can enhance the searchStringInJson and searchStringInJsonObject functions to support case sensitivity, regular expressions, or searching within specific HTML elements.