**Assignment: Web Scraping and Crawling with Python**

**Introduction:**

In this assignment, you will learn how to create a Python program to scrape and crawl Wikipedia pages to download random images from them. You will use several libraries and functions to accomplish this task.

Libraries and Their Purpose:

1. **requests:** The `requests` library is used for making HTTP requests to web pages. It is essential for fetching web page content.
2. **BeautifulSoup (from bs4):** BeautifulSoup is a library for parsing HTML and XML documents. It helps extract data from web pages easily and efficiently.
3. **random**: Purpose: Generates random values for sampling during web crawling.
4. **os**: Purpose: Manages directories and file paths for storing downloaded images.
5. **urllib.parse**: Purpose: Parses and manipulates URLs to ensure proper formatting for web requests.

**Key Functions and Their Descriptions (from the libraries):**

- `requests.get(url)`: Sends an HTTP GET request to the specified URL and returns a response object containing the server's response.

- `BeautifulSoup(html\_content, 'html.parser')`: Initializes a BeautifulSoup object to parse HTML content using the 'html.parser' parser.

- `BeautifulSoup.find(name, attrs, recursive, string, \*\*kwargs)`: Searches for the first instance of a tag that matches the given criteria within the BeautifulSoup object.

- `BeautifulSoup.find\_all(name, attrs, recursive, string, limit, \*\*kwargs)`: Searches for all instances of a tag that match the given criteria within the BeautifulSoup object.

**String Functions (used in the code):**

- `rsplit()`: Splits a string into a list, starting from the right and working towards the left. Used to extract file extensions.

- `split()`: Splits a string into a list based on a specified delimiter. Used to extract image names.

- `replace()`: Replaces specified characters in a string with other characters. Used to handle special characters in image names.

**Main Ideas/Actions (without mentioning specific functions):**

1. Fetch the HTML content of a Wikipedia page.
2. Extract relevant information from the HTML, such as image URLs and page links.
3. Download images to a specified directory.
4. Implement a recursive crawling mechanism to explore multiple Wikipedia pages.

**Secondary Considerations (without mentioning specific functions):**

1. Error handling: Handle potential errors such as failed HTTP requests or file writing errors.
2. URL formatting: Ensure that URLs are properly constructed and joined when necessary.
3. Directory creation: Create directories for each Wikipedia page to organize downloaded images.
4. Argument handling: Implement a main function to accept user input for depth and width of the crawling operation using `sys.argv`.

**Assignment Tasks:**

Your task is to create a Python program that replicates the functionality described above. You should implement the following steps:

1. Import the necessary libraries: `requests`, `BeautifulSoup`, and any other libraries you require.
2. Define functions similar to the ones mentioned above to perform the required actions.
3. Implement error handling and proper URL handling to ensure the program is robust.
4. Create a main function that accepts `sys.argv` arguments for depth and width of crawling.
5. Test your program by crawling Wikipedia pages and downloading images. Ensure it works as expected.
6. Provide comments and explanations for each function and major code sections.
7. Submit your well-documented Python script along with a brief report describing your approach and any challenges you encountered.

**Note:** Be sure to adhere to ethical web scraping practices and respect website terms of use while completing this assignment.