Scrapy Spider Autorepair

How to run the code?
1. Clone this repository
2. Go to the Scrapy-Spider-Autorepair directory using command line interface
3. type: python auto_repair_code.py
4. Hit Enter

How to interpret the output?

There are 3 examples. Each example consists of 2 files. First file consists of html code and the second file consists of the html code of the subtree to be exracted from the first file.

Let's try to understand the output generated by the code. Let's look at example 1 in the generated output.

This output consists of the results from 2 functionalities:
1. Subtree Extraction:
If you look at the file named query1.html, it contains the html code for the subtree to be extracted. The file named 1.html consists of the html code from which the subtree in query1.html is extracted.

Path Representation:

Consider the simple code::

```
<html>
   <body>
      <p>Browsers usually insert quotation marks around the q element.</p>
      <q>Build a future where people live in harmony with nature.</q>
   </body>
</html>
```

The <body>...</body> subtree is the 0th child of <html></html>. Similarly, the ... subtree is child 0 of <body>...</body> and <q>...</q> is child 1 of <body>...</body>. Thus the path to <q> is [0, 1]. It shows the child numbers that we need to select to reach element <q>...</q> as we traverse this tree from top to bottom.

XPath Representation:

Consider example 3.
query3.html contains:

```
<div>
   <div>
         <div>
            <p>
               <p>scrapy</p>
            </p>
         </div>
   </div>
   <div>
         <p>portia</p>
   </div>
   <p>slybot</p>
</div>
```

and

3.html contains:

```
<html>
   <body>
      <p>scrapy</p>
      <p>slybot</p>
      <p>portia</p>
   </body>
</html>
```

Now our goal was to find the subtree in query3.html in 3.html. Clearly, the subtree cannot be found in 3.html. Hence, the code, recursively, tries to find every subtree of the subtree in query3.html in the tree given in 3.html. Now, as there are no tags in 3.html, only the 3 leaf nodes of query3.html matches with the 3 leaf nodes in 3.html.
xpaths(in output) is a list of tuples(path of the position in query3.html where the subtree extracted from 3.html is to be inserted, path of the position in 3.html which is to be extracted).
For example, if xpaths = [([0, 0, 0, 0], [0, 0]), ([1, 0], [0, 2]), ([2], [0, 1])], the 2nd tuple in this list shows that we need to insert subtree portia present at path [0, 2] in 3.html at position [1, 0] in query3.html.