



## HTML Parser - Part 2 ★

88/115 challenges solved

Rank: 34240 | Points: 1545 ⓘ

[Problem](#)[Submissions](#)[Leaderboard](#)[Editorial](#)

\*This section assumes that you understand the basics discussed in [HTML Parser - Part 1](#)

### [.handle\\_comment\(data\)](#)

This method is called when a comment is encountered (e.g. `<!--comment-->`).

The data argument is the content inside the comment tag:

```
from html.parser import HTMLParser

class MyHTMLParser(HTMLParser):
    def handle_comment(self, data):
        print("Comment :", data)
```

### [.handle\\_data\(data\)](#)

This method is called to process arbitrary data (e.g. text nodes and the content of `<script>...</script>` and `<style>...</style>`).

The data argument is the text content of HTML.

```
from html.parser import HTMLParser

class MyHTMLParser(HTMLParser):
    def handle_data(self, data):
        print("Data      :", data)
```

### Task

You are given an HTML code snippet of ***N*** lines.

Your task is to print the single-line comments, multi-line comments and the data.

Print the result in the following format:

```
>>> Single-line Comment
Comment
>>> Data
My Data
>>> Multi-line Comment
Comment_multiline[0]
Comment_multiline[1]
>>> Data
My Data
>>> Single-line Comment:
```

**Note:** Do not print data if `data == '\n'`.

### Input Format

The first line contains integer ***N***, the number of lines in the HTML code snippet.

The next ***N*** lines contain HTML code.

### Constraints

**$0 < N < 100$**

### Output Format

Print the single-line comments, multi-line comments and the data in order of their occurrence from top to bottom in the snippet.

Format the answers as explained in the problem statement.

## Sample Input

```
4
<!--[if IE 9]>IE9-specific content
<![endif]-->
<div> Welcome to HackerRank</div>
<!--[if IE 9]>IE9-specific content<![endif]-->
```

## Sample Output

```
>>> Multi-line Comment
[if IE 9]>IE9-specific content
<![endif]
>>> Data
Welcome to HackerRank
>>> Single-line Comment
[if IE 9]>IE9-specific content<![endif]
```

Change Theme Language Python 3



```
1 from html.parser import HTMLParser
2
3 class MyHTMLParser(HTMLParser):
4     def handle_comment(self, data):
5         if '\n' in data:
6             print('>>> Multi-line Comment', data, sep='\n')
7         else:
8             print('>>> Single-line Comment', data, sep='\n')
9
10    def handle_data(self, data):
11        if data.strip():
12            print('>>> Data', data, sep='\n')
13
14
15
16
17
18
19
20
21
22    html = ""
23    for i in range(int(input())):
24        html += input().rstrip()
25        html += '\n'
26
27    parser = MyHTMLParser()
28    parser.feed(html)
29    parser.close()
30
```

EMACS

Line: 30 Col: 1

Run Code


Submit Code

[Upload Code as File](#)☐ Test against custom input

## Fetching Results

✔ Test case 0  ✔ Test case 4 

✔ Test case 1  ✔ Test case 5 

✔ Test case 2 

✔ Test case 3 

[Blog](#) | [Scoring](#) | [Environment](#) | [FAQ](#) | [About Us](#) | [Helpdesk](#) | [Careers](#) | [Terms Of Service](#) | [Privacy Policy](#)