Module 8 Cheat Sheet

BeautifulSoup()

- **Definition:** Creates a BeautifulSoup object for parsing HTML or XML documents.

Syntax:
python
Copy code
```
from bs4 import BeautifulSoup
soup = BeautifulSoup(html_content, 'html.parser')
```

find()

- **Definition:** Finds the first occurrence of a tag in the document.

Syntax:
python
Copy code
```
tag = soup.find('tag_name')
```

find_all()

- **Definition:** Finds all occurrences of a tag in the document and returns a list.

Syntax:
python
Copy code
```
tags = soup.find_all('tag_name')
```

get_text()

- **Definition:** Extracts all the text from a tag, stripping away the HTML tags.

Syntax:
python
Copy code
```
text = tag.get_text()
```

attrs

- **Definition:** Accesses the attributes of a tag.

Syntax:
python
Copy code
```
attributes = tag.attrs
```

find_parents()

- **Definition:** Finds all parent tags of a specific tag.

Syntax:
python
Copy code
```
parents = tag.find_parents('parent_tag_name')
```

find_parent()

- **Definition:** Finds the immediate parent tag of a specific tag.

Syntax:
python
Copy code
```
parent = tag.find_parent('parent_tag_name')
```

find_next_sibling()

- **Definition:** Finds the next sibling tag of a specific tag.

Syntax:
python
Copy code
```
next_sibling = tag.find_next_sibling('sibling_tag_name')
```

find_previous_sibling()

- **Definition:** Finds the previous sibling tag of a specific tag.

Syntax:
python
Copy code
```
previous_sibling = tag.find_previous_sibling('sibling_tag_name')
```

find_all_next()

- **Definition:** Finds all tags that appear after a specific tag.

Syntax:
python
Copy code
```
next_tags = tag.find_all_next('next_tag_name')
```

find_all_previous()

- **Definition:** Finds all tags that appear before a specific tag.

Syntax:
python
Copy code
```
previous_tags = tag.find_all_previous('previous_tag_name')
```

select()

- **Definition:** Finds tags using a CSS selector.

Syntax:
python
Copy code
```
selected_tags = soup.select('css_selector')
```

select_one()

- **Definition:** Finds the first tag that matches a CSS selector.

Syntax:
python
Copy code
```
selected_tag = soup.select_one('css_selector')
```

decompose()

- **Definition:** Removes a tag from the parse tree.

Syntax:
python
Copy code
```
tag.decompose()
```

replace_with()

- **Definition:** Replaces a tag with another tag or string.
- **Syntax:**
  python
  Copy code
  ```
  tag.replace_with(new_tag_or_string)
  ```