

jsoup News Bugs Discussion Download API Reference Cookbook Try jsoup

[jsoup](#) » [Cookbook](#) » [Introduction](#) » Parsing and traversing a Document

Parsing and traversing a Document

To parse a HTML document:

```
String html = "<html><head><title>First parse</title></head>"
+ "<body><p>Parsed HTML into a doc.</p></body></html>";
Document doc = Jsoup.parse(html);
```

(See [parsing a document from a string](#) for more info.)

The parser will make every attempt to create a clean parse from the HTML you provide, regardless of whether the HTML is well-formed or not. It handles:

- unclosed tags (e.g. `<p>Lorem <p>Ipsum` parses to `<p>Lorem</p> <p>Ipsum</p>`)
- implicit tags (e.g. a naked `<td>Table data</td>` is wrapped into a `<table><tr><td>...`)
- reliably creating the document structure (html containing a head and body, and only appropriate elements within the head)

The object model of a document

- Documents consist of Elements and TextNodes (and a couple of other misc nodes: see the [nodes package tree](#)).
- The inheritance chain is: **Document** extends **Element** extends **Node**. **TextNode** extends **Node**.
- An Element contains a list of children Nodes, and has one parent Element. They also have provide a filtered list of child Elements only.

See also

- Extracting data: [DOM navigation](#)
- Extracting data: [Selector syntax](#)

Cookbook contents

Introduction

1. Parsing and traversing a Document

Input

2. Parse a document from a String
3. Parsing a body fragment
4. Load a Document from a URL
5. Load a Document from a File

Extracting data

6. Use DOM methods to navigate a document
7. Use selector-syntax to find elements
8. Extract attributes, text, and HTML from elements
9. Working with URLs
10. Example program: list links

Modifying data

11. Set attribute values
12. Set the HTML of an element
13. Setting the text content of elements

Cleaning HTML

14. Sanitize untrusted HTML (to prevent XSS)