

Chapter 93: Fluent API

JavaScript is great for designing fluent API - a consumer-oriented API with focus on developer experience. Combine with language dynamic features for optimal results.

Section 93.1: Fluent API capturing construction of HTML articles with JS

Version ≥ 6

```
class Item {
  constructor(text, type) {
    this.text = text;
    this.emphasis = false;
    this.type = type;
  }

  toHtml() {
    return `<${this.type}>${this.emphasis ? '<em>' : ''}${this.text}${this.emphasis ? '</em>' : ''}</${this.type}>`;
  }
}

class Section {
  constructor(header, paragraphs) {
    this.header = header;
    this.paragraphs = paragraphs;
  }

  toHtml() {
    return `<section><h2>${this.header}</h2>${this.paragraphs.map(p => p.toHtml()).join('')}</section>`;
  }
}

class List {
  constructor(text, items) {
    this.text = text;
    this.items = items;
  }

  toHtml() {
    return `<ol><h2>${this.text}</h2>${this.items.map(i => i.toHtml()).join('')}</ol>`;
  }
}

class Article {
  constructor(topic) {
    this.topic = topic;
    this.sections = [];
    this.lists = [];
  }

  section(text) {
    const section = new Section(text, []);
    this.sections.push(section);
    this.lastSection = section;
    return this;
  }
}
```

```

list(text) {
  const list = new List(text, []);
  this.lists.push(list);
  this.lastList = list;
  return this;
}

addParagraph(text) {
  const paragraph = new Item(text, 'p');
  this.lastSection.paragraphs.push(paragraph);
  this.lastItem = paragraph;
  return this;
}

addListItem(text) {
  const listItem = new Item(text, 'li');
  this.lastList.items.push(listItem);
  this.lastItem = listItem;
  return this;
}

withEmphasis() {
  this.lastItem.emphasis = true;
  return this;
}

toHtml() {
  return `<article><h1>${this.topic}</h1>${this.sections.map(s =>
s.toHtml()).join('')}${this.lists.map(l => l.toHtml()).join('')}</article>`;
}
}

Article.withTopic = topic => new Article(topic);

```

This allows the consumer of the API to have a nice-looking article construction, almost a DSL for this purpose, using plain JS:

Version ≥ 6

```

const articles = [
  Article.withTopic('Artificial Intelligence - Overview')
    .section('What is Artificial Intelligence?')
    .addParagraph('Something something')
    .addParagraph('Lorem ipsum')
    .withEmphasis()
    .section('Philosophy of AI')
    .addParagraph('Something about AI philosophy')
    .addParagraph('Conclusion'),

  Article.withTopic('JavaScript')
    .list('JavaScript is one of the 3 languages all web developers must learn:')
    .addListItem('HTML to define the content of web pages')
    .addListItem('CSS to specify the layout of web pages')
    .addListItem(' JavaScript to program the behavior of web pages')
];

document.getElementById('content').innerHTML = articles.map(a => a.toHtml()).join('\n');

```