

Semantic HTML

Dr Harshad Prajapati
7 Nov 2023

Semantics

- **Semantics** in programming refers to the **meaning** of a piece of code.
- In HTML,
 - **Focus** is on **what** is the **role** or **purpose** of an HTML **element**?
 - Instead of **what** does that **element look** like?

Example of Semantics in HTML

Page Header

HTML: HyperText Markup Language

HTML (HyperText Markup Language) is the most basic building block of the Web. It defines the meaning and structure of web content. Other technologies besides HTML are generally used to describe a web page's appearance/presentation ([CSS](#)) or functionality/behavior ([JavaScript](#)).

- Example of semantic element in HTML:
 - The **h1** element is a semantic element.
 - The **h1 element** gives the text it wraps a meaning/role that the text being wrapped is a top level heading of our page.
 - By default browser's user agent stylesheet will style an **h1** element with a large font size to make it look like a heading.

3

Use Right Element for Semantic Value


- By providing appropriate style, we could make any element look like a top level heading.
 - ```

 Not a top-level heading!

```

This code will display the text look like top level heading.
- The above code will render the text to look like a top level heading, but it has no semantic value.
- We need to use right HTML element for the right job.
- Why using right element for semantic value is important?
  - A webpage is read not only by humans.


4



Humans  
read this

# HTML: HyperText Markup Language

HTML (HyperText Markup Language) is the most basic building block of the Web. It defines the meaning and structure of web content. Other technologies besides HTML are generally used to describe a web page's appearance/presentation ([CSS](#)) or functionality/behavior ([JavaScript](#)).



Machines  
read this

```
<article class="main-page-content" lang="en-US"><header><h1>HTML: HyperText Markup Language</h1></header><div class="section-content"><p>HTML (HyperText Markup Language) is the most basic building block of the Web. It defines the meaning and structure of web content. Other technologies besides HTML are generally used to describe a web page's appearance/presentation (CSS) or functionality/behavior (JavaScript).</p>
```

5

## Benefits of Writing Semantic Markup

- **SEO** uses for **page's** search **rankings**.
- **Screen reader** can use it to help visually impaired users navigate a page.
- If **only divs** are used, it is very **difficult** to **find blocks** of meaningful code.
- **Suggests** to the **developer** the **type of data** that will be populated.

6

# SEO: Search Engine Optimization

- SEO (Search Engine Optimization) is the **process** of making a **website more visible** in **search engines' results pages**.
  - It is also referred to as **improving search rankings**.
- Search engines **crawl web, following links** from page to page, and index the content that is found.
- When we search, the **search engine displays** the **indexed content**.
- **Crawlers follow some rules**.
  - If **we follow those rules** when doing SEO for a website, we **can improve search optimization** of our website.

7

# SEO: Search Engine Optimization

- **SEO methods** fall into three categories:
  - **Technical**: Use semantic tags.
  - **Copywriting**: Use visitors' vocabulary.
  - **Popularity**: Other established sites link to our site.
- One of three methods is **Technical**:
  - We **tag** the **content** using **semantic HTML**.
  - When crawlers explore our website, they should find the content that we want indexed.

8

# Questions to Think When Deciding HTML Elements

- We need to ask questions to ourselves :
- What element(s) best describe/represent the data that we are populating.
- Is it a list of data?
  - Is data ordered?
  - Is data unordered?
- Is it an article with sections and an aside of related information?
- Does data represent list of definitions?
- Is it a figure or image that needs a caption?
- Should we have a header and footer in addition to the global site-wide header and footer.

9

## Semantic Elements

- There are some 100 semantics elements available. Some are
- <article>
- <aside>
- <details>
- <figcaption>
- <figure>
- <footer>
- <form>
- <header>
- <main>
- <mark>
- <nav>
- <section>
- <summary>
- <time>

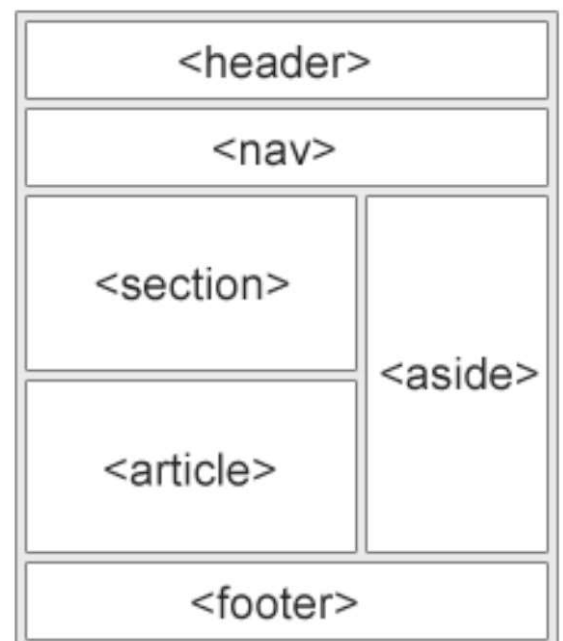
10

# Semantic Elements and Non-semantic Elements

- A **semantic element** clearly describes its **meaning** to both the **browser** and the **developer**.
- Examples of **non-semantic elements**: (Do not convey anything about content)
  - `<span>`
  - `<div>`
- Examples of **semantic elements**: (They clearly define content)
  - `<form>`
  - `<table>`
  - `<article>`
  - `<aside>`

11

## Typical Parts in a Webpage



12

# Example of a Webpage

Spring Framework Documentation6.0.13

SearchCTRL + k

Overview

Core Technologies

Testing

Data Access

Web on Servlet Stack

Web on Reactive Stack

Integration

Language Support

Appendix

Wiki

Spring Framework / Spring Framework Documentation

Spring Framework Documentation

OverviewHistory, Design Philosophy, Feedback, Getting Started.

CoreIoC Container, Events, Resources, i18n, Validation, Data Binding, Type Conversion, SpEL, AOP, AOT.

TestingMock Objects, TestContext Framework, Spring MVC Test, WebTestClient.

Data AccessTransactions, DAO Support, JDBC, R2DBC, O/R Mapping, XML Marshalling.

Web ServletSpring MVC, WebSocket, SockJS, STOMP Messaging.

Web ReactiveSpring WebFlux, WebClient, WebSocket, RSocket.

IntegrationREST Clients, JMS, JCA, JMX, Email, Tasks, Scheduling, Caching, Observability.

LanguagesKotlin, Groovy, Dynamic Languages.

AppendixSpring properties.

WikiWhat's New, Upgrade Notes, Supported Versions, additional cross-version information.

Edit this Page

Github Project

Stack Overflow

13

# Example of a Webpage

Spring Framework Documentation6.0.13

SearchCTRL + k

Overview

Core Technologies

Testing

Data Access

Web on Servlet Stack

Web on Reactive Stack

Integration

Language Support

Appendix

Wiki

Spring Framework / Spring Framework Documentation

Spring Framework Documentation

OverviewHistory, Design Philosophy, Feedback, Getting Started.

CoreIoC Container, Events, Resources, i18n, Validation, Data Binding, Type Conversion, SpEL, AOP, AOT.

TestingMock Objects, TestContext Framework, Spring MVC Test, WebTestClient.

Data AccessTransactions, DAO Support, JDBC, R2DBC, O/R Mapping, XML Marshalling.

Web ServletSpring MVC, WebSocket, SockJS, STOMP Messaging.

Web ReactiveSpring WebFlux, WebClient, WebSocket, RSocket.

IntegrationREST Clients, JMS, JCA, JMX, Email, Tasks, Scheduling, Caching, Observability.

LanguagesKotlin, Groovy, Dynamic Languages.

AppendixSpring properties.

WikiWhat's New, Upgrade Notes, Supported Versions, additional cross-version information.

Edit this Page

Github Project

Stack Overflow

14

# HTML <section> Element

- The **<section> element** is a generic container that is used to **group** together **content** that is **thematically related**.
  - For example, the **title of a chapter** and the **content of the chapter** are **related**, so we can group them using **<section>**
- It's often used to **divide** a document **into chapters, headers, footers**, or any other group of content.
- We can use **<section>** element for the following:
  - Chapters
  - Introduction
  - News items
  - Contact information

15

## Example of Section

**<section>** {  

### 1. Legal

Copyright © 2012-2023

Copies of this document may be made for your own use and for distribution to others, provided that you do not charge any fee for such copies and further provided that each copy contains this Copyright Notice, whether distributed in print or electronically.

---

**<section>** {  

### 2. Getting Help

If you have trouble with Spring Boot, we would like to help.

- Try the [How-to documents](#). They provide solutions to the most common questions.
- Learn the Spring basics. Spring Boot builds on many other Spring projects. Check the [spring.io](#) web-site for a wealth of reference documentation. If you are starting out with Spring, try one of the [guides](#).
- Ask a question. We monitor [stackoverflow.com](#) for questions tagged with [spring-boot](#).
- Report bugs with Spring Boot at [github.com/spring-projects/spring-boot/issues](#).

16



## Example of HTML <section> Element

<section>

<h2>Chapter 1</h2>

<p>Content of Chapter 1...</p>

</section>

<section>

<h2>Chapter 2</h2>

<p>Content of Chapter 2...</p>

</section>

17

## HTML <article> Element

- The **<article> element** is more specific and is used to define a **self-contained** piece of **content** that could be **distributed** and **reused independently**.
- It is **suitable** for content like
  - Blog posts.
  - Newspaper articles.
  - Forum posts.
  - User comments.
  - Product cards.
- The **content inside** an **<article>** should make **sense** on its **own** and **should not rely** on the **surrounding content** for **meaning**.

18

## Understanding Database Types



ALEX XU

APR 19, 2023 · PAID



638



11

Share

&lt;article&gt;

The success of a software application often hinges on the choice of the right databases. As developers, we're faced with a vast array of database options. It is crucial for us to understand the differences between these options and how to select the ones that best align with our project's requirements. A complex application usually uses several different databases, each catering to a specific aspect of the application's needs.

### Mastering the Art of Database Selection

#### Understanding Database Types

Relational Databases

NoSQL

4

4

6

19

## Example of HTML <article> Element

&lt;article&gt;

&lt;h2&gt;Article Title&lt;/h2&gt;

&lt;p&gt;Content of the article...&lt;/p&gt;

&lt;p&gt;Content of the article...&lt;/p&gt;

&lt;/article&gt;

# HTML <header> Element

- The **<header> element** represents a **container** for **introductory content** or a set of **navigational links**.
- A **<header> element** generally contains:
  - One or more **heading elements** (<h1> to <h6>)
  - **Logo** or icon.
  - **Authorship** information (Author or Organization or both or Timestamp)
- We can have **several <header> elements** in **one HTML document**.
  - However, we **cannot place** a **<header> within** another **<header>** or **<footer>** or **<address>** element.

21

**<header>**



The screenshot shows the top of the Spring Framework documentation page. A red box highlights the header area, which contains the Spring logo, navigation links (Why Spring, Learn, Projects, Academy, Solutions, Community), and a settings icon. Below the navigation bar, the page title 'Spring Framework Documentation' is visible, along with a sidebar menu on the left and a right-hand menu with links to 'Edit this Page', 'Github Project', and 'Stack Overflow'.

22

## Example of HTML <header> Element

```
<article>
 <header>
 <h1>Article Title</h1>
 <p>Published on <time datetime="2023-11-12">November 12, 2023</time> by
Author Name</p>
 </header>

 <p>This is the content of the article...</p>
 <p>More paragraphs and details...</p>

 <!-- Additional content of the article -->
</article>
```

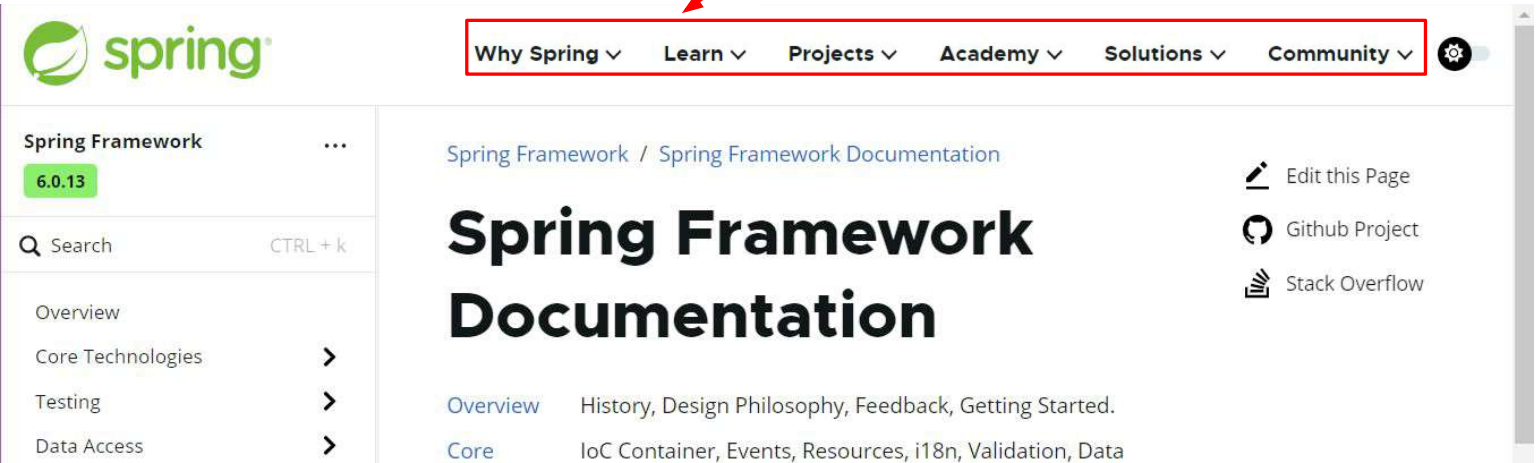
23

## HTML <nav> Element

- ✓ The <nav> element defines a set of navigation links.
- The <nav> element is to be used for major blocks of navigation links.
  - We should not place all links of a document inside <nav> element.
- Browsers have screen readers (disabled users use it):
  - Screen readers can use this element to determine whether to omit the initial reading of the content of <nav>.

24

`<nav>`



25

## Example of HTML `<nav>` Element

```
<header>
 <h1>Website Name</h1>
 <nav>

 Home
 About Us
 Services
 Contact Us

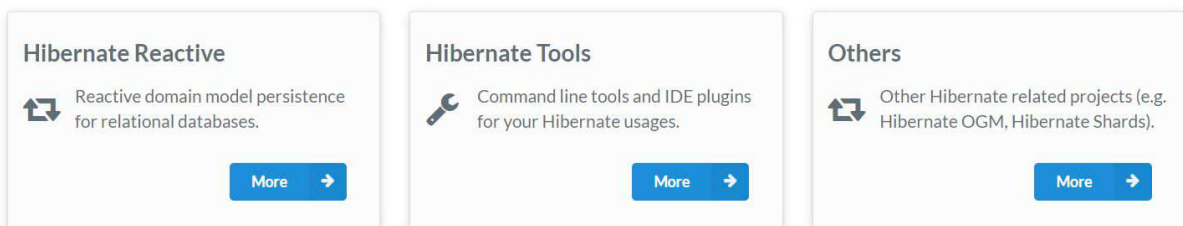
 </nav>
</header>
```

26

# HTML <footer> Element

- The **<footer>** element defines the **footer** of a **document** or **section**.
- A **<footer>** typically contains:
  - **Copyright** information.
  - **Authorship** information.
  - **Sitemap**.
  - **Contact** information.
  - **Back to top links**.
  - **Related documents**.
- We can have **several <footer> elements** in **one document**.
- This **<footer> element** and its **content** is **very important** for **SEO**.

27



## Projects

Hibernate ORM  
Hibernate Search  
Hibernate Validator  
Hibernate Reactive  
Hibernate Tools  
Other projects

## Follow us

Blog  
Twitter

## Contribute and community

Community resources  
Our GitHub organization  
Submit a bug  
Our forums  
Report a security issue  
License



Sponsored by  
**Red Hat**

**<footer>**

28

## Example of <footer> Element

```
<footer>
 <p>© 2023 Example Page. All rights reserved.</p>
 <nav>

 About Us
 Contact Us
 Privacy Policy

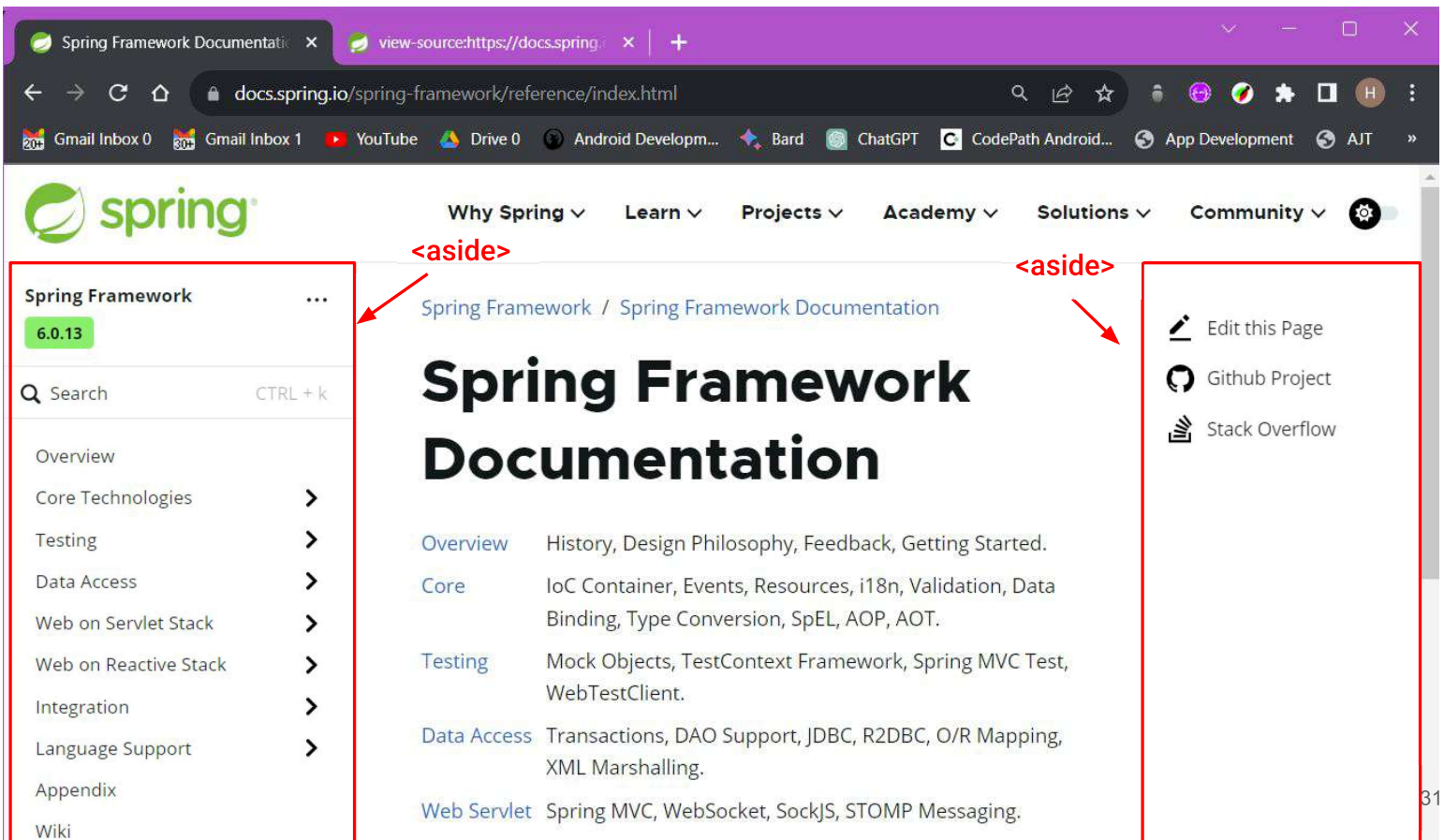
 </nav>
</footer>
```

29

## HTML <aside> Element

- ✓ The <aside> element defines some content aside from the content that is placed in (like sidebar).
- The content of <aside> should be directly related to the surrounding content.

30



## Example of HTML `<aside>` Element

`<aside>`

```
<section>
 <h2>Related Links</h2>

 Related Article 1
 Related Article 2
 Related Article 3

</section>
```

```
<section>
 <h2>Advertisement</h2>
 <p>Check out our special offers!</p>
 <!-- Additional ad content -->
</section>
```

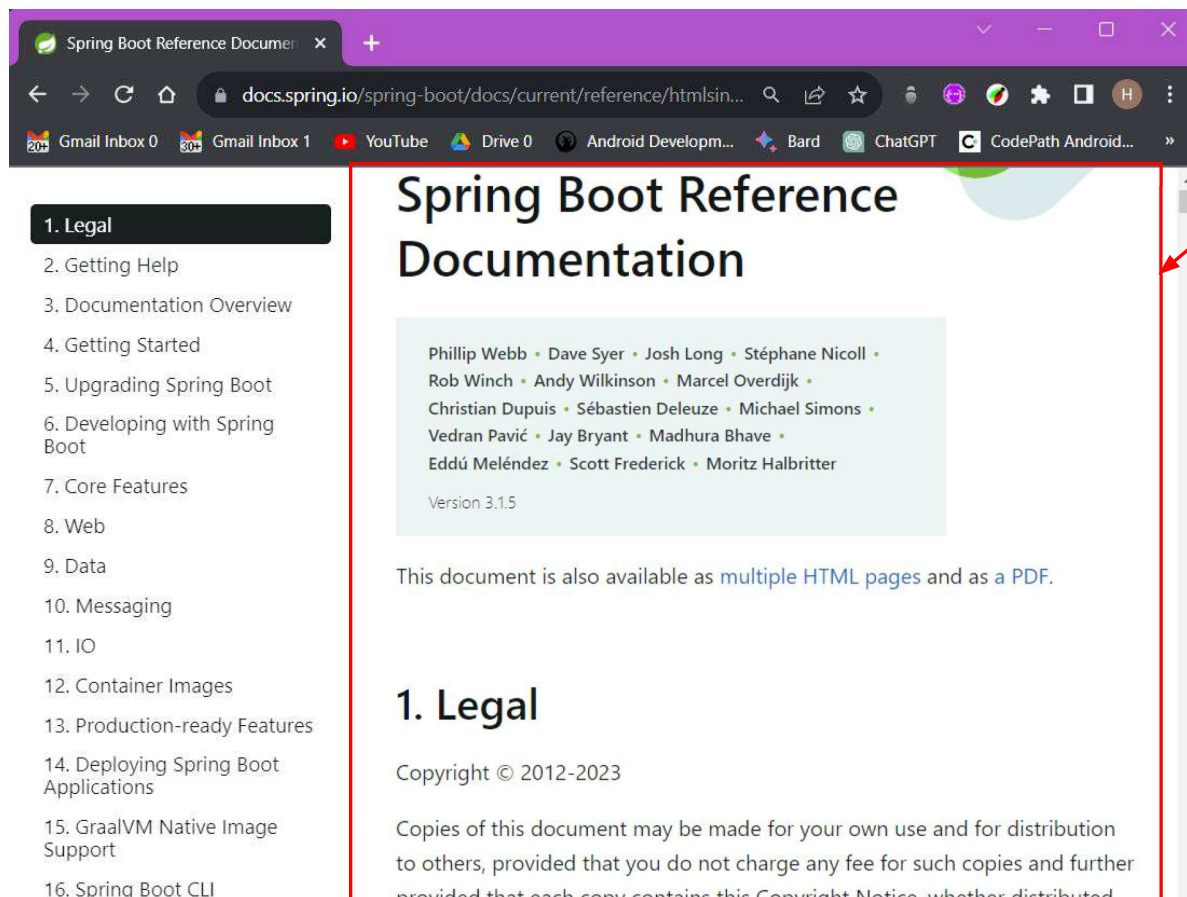
`</aside>`



# HTML <main> Element

- ✓ The **<main>** element specifies the **main content** of a document.
- ✓ The **content inside <main>** should be **unique** to **the document**.
  - It should **not contain** content that is **repeated** in **all documents**:
    - **Sidebars, Navigation links, Copyright information, Site logos., Search forms, etc.**
    - The must **not** be **more than one <main>** element in a document.
- The **<main>** element should **not** be a **child** of an **<article>**, **<aside>**, **<footer>**, **<header>**, or **<nav>** element.

33



<main>

34

## Example of HTML <main> Element

```
<body>
...
<main>
 <section>
 <h2>Welcome to Our Website</h2>
 <p>Thank you for visiting our website. Explore our content to learn more about who we are and what we offer.</p>
 </section>

 <section>
 <h2>Featured Services</h2>

 Service 1: Description of Service 1
 Service 2: Description of Service 2
 Service 3: Description of Service 3

 </section>
</main>
</body>
```

35

## HTML <figure> and <figcaption> Elements

- The **<figure>** element specifies **self-contained content**.
  - Code listings.
  - Illustrations.
  - Diagrams.
  - Photos.
- The **<figcaption>** element defines a **caption** for a **<figure>** element.
  - The **<figcaption>** element can be **placed** as the **first child** element or the **last child** element of **<figure>** element.
- The **actual image itself** is **defined** by **<img>** element.

36

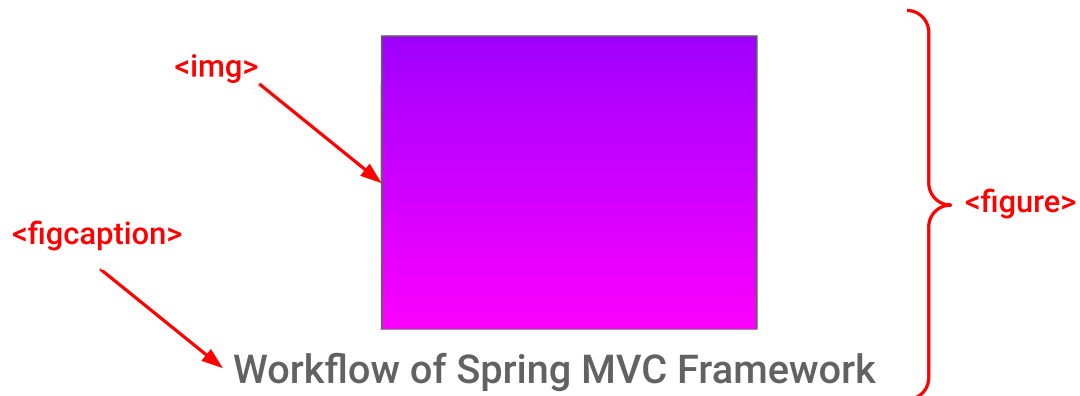
# Example of HTML <figure> and <figcaption> Elements

<figure>



<figcaption>Workflow of Spring MVC Framework</figcaption>

</figure>



37

## References

- <https://developer.mozilla.org/en-US/docs/Glossary/Semantics>
- [https://www.w3schools.com/html/html5\\_semantic\\_elements.asp](https://www.w3schools.com/html/html5_semantic_elements.asp)

38