IRON
HACK

HTML: Introduction

Juan “Harek” Urrios
@xharekx33

What is HTML?

HTML stands for HyperText Markup Language. Is the language that describes the semantic structure of a website so web browsers can render the pages and users can view or hear them.

It is the brainchild of Sir Tim Berners-Lee who in 1991 came up with a document titled "HTML tags" describing 18 elements that could be used to describe web pages.

IRON HACK

How to create a web page

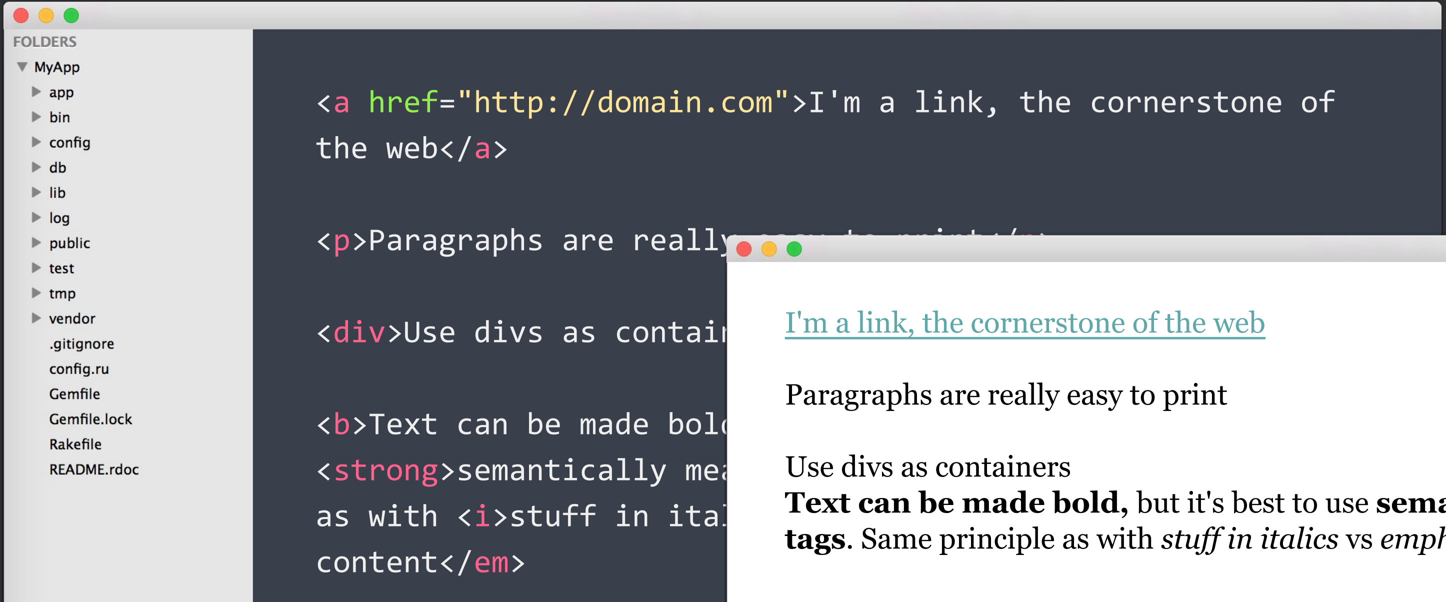
To create a simple web page all you need is a text editor and a browser.

Create a new file, foo.html, add some HTML code, save it and open the file in your browser. Whenever you change something, just reload the page in the browser to see them.

For a more convenient and quick option for running your tests, you can also use tools such as codepen.io or jsfiddle.net

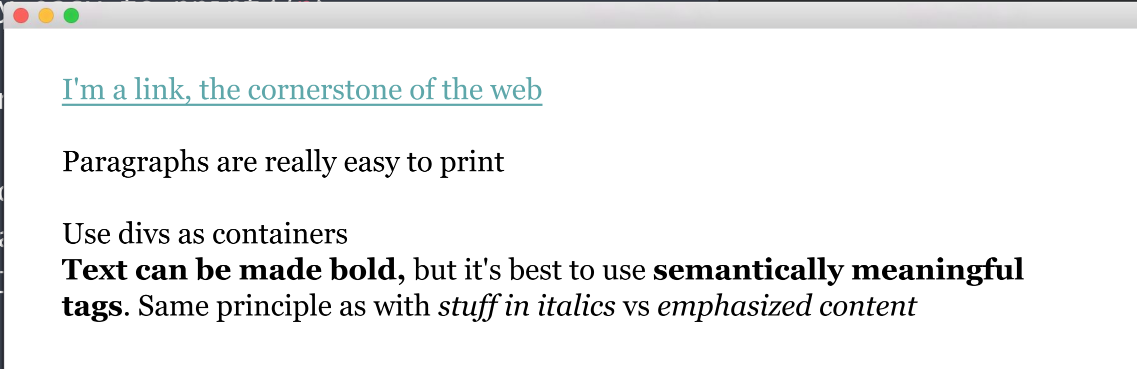
Tags and attributes

HTML consists of tags, that describe the elements in the page and its structure and attributes that will provide values to enable or modify their functionality.



FOLDERS
- MyApp
  - app
  - bin
  - config
  - db
  - lib
  - log
  - public
  - test
  - tmp
  - vendor
  - .gitignore
  - config.ru
  - Gemfile
  - Gemfile.lock
  - Rakefile
  - README.rdoc

```
<a href="http://domain.com">I'm a link, the cornerstone of
the web</a>

<p>Paragraphs are really

<div>Use divs as contai

<b>Text can be made bol
<strong>semantically me
as with <i>stuff in ita
content</em>
```



I'm a link, the cornerstone of the web

Paragraphs are really easy to print

Use divs as containers
Text can be made bold, but it's best to use **semantically meaningful tags**. Same principle as with *stuff in italics* vs *emphasized content*

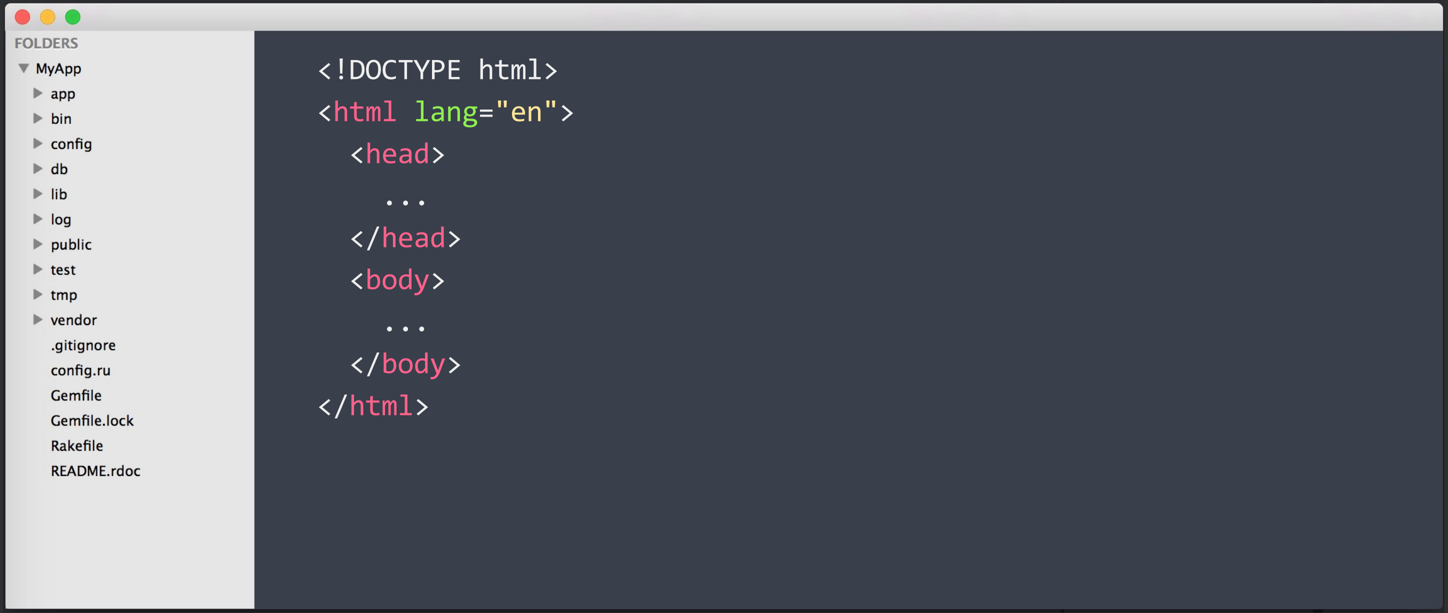
What are tags and attributes good for?

Using tags and attributes you can:

- Add content to the page.
- Wrap it and add structure and meaning to it.
- Style it.
- Load assets (images, video, stylesheets, javascript...).
- Make it interactive.
- Tell the browser how your page should be rendered and how it should behave.

Browsers will not display the actual tags, but will use them to render the page.

A basic HTML page



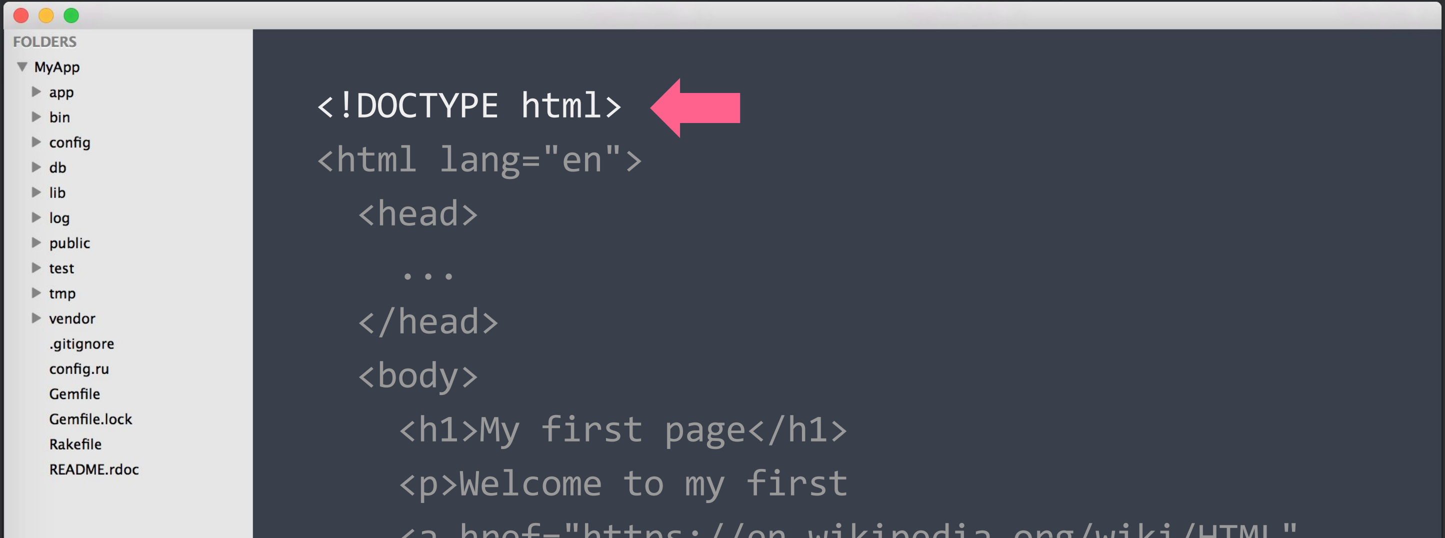
```html
<!DOCTYPE html>
<html lang="en">
  <head>
    ...
  </head>
  <body>
    ...
  </body>
</html>
```

IRON HACK

@xharekx33

The doctype

The doctype (Document Type Declaration) specifies the type (and version) of markup a document is written in, so the browser know what to expect and renders it properly. Always include it at the beginning of your pages.

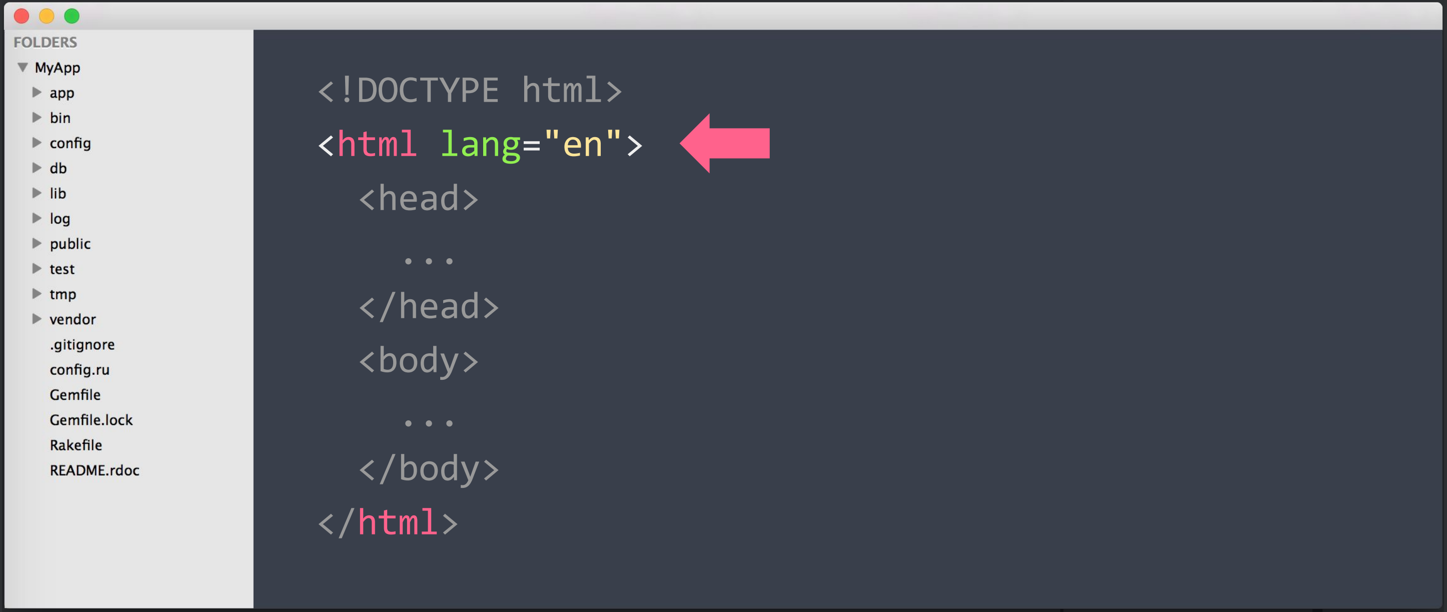


```
<!DOCTYPE html>
<html lang="en">
  <head>
    ...
  </head>
  <body>
    <h1>My first page</h1>
    <p>Welcome to my first
```

@xharekx33

The <html> tag

Tells the browser that this is a HTML document: it represents the start of the document and serves as a container for all the rest of the elements in the page.

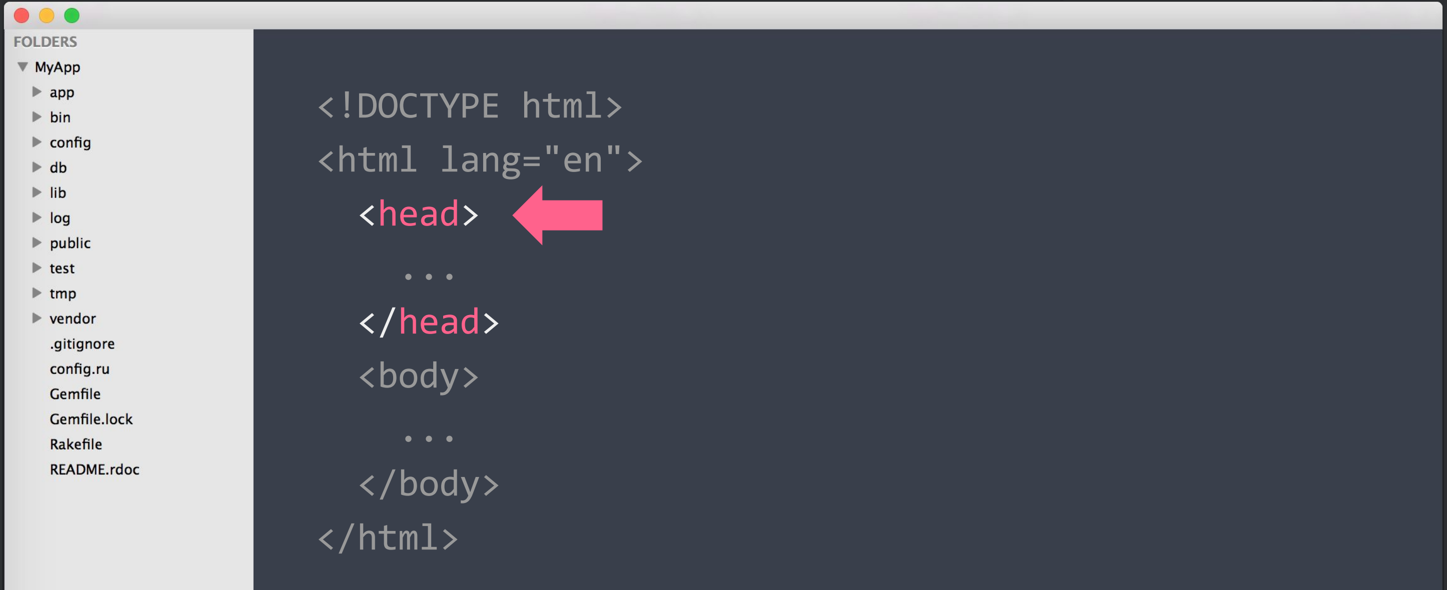


```
<!DOCTYPE html>
<html lang="en">
  <head>
    ...
  </head>
  <body>
    ...
  </body>
</html>
```

@xharekx33

The <head> tag

Contains all the properties and metadata of the document including its title, base url and links and or definitions for stylesheets.



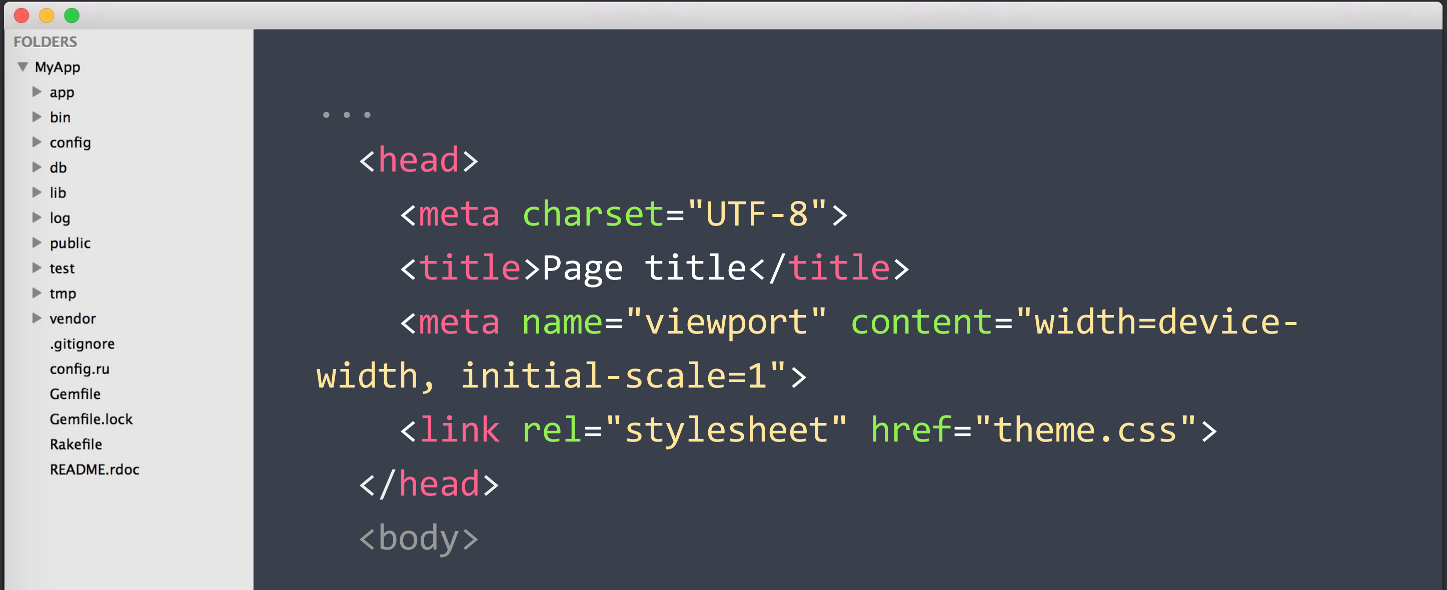
```
<!DOCTYPE html>
<html lang="en">
  <head>

    ...

  </head>
  <body>

    ...

  </body>
</html>
```

IRON HACK

@xharekx33

<head>: What goes inside?

Important stuff to include in your <head>: title, charset, viewport, styles...



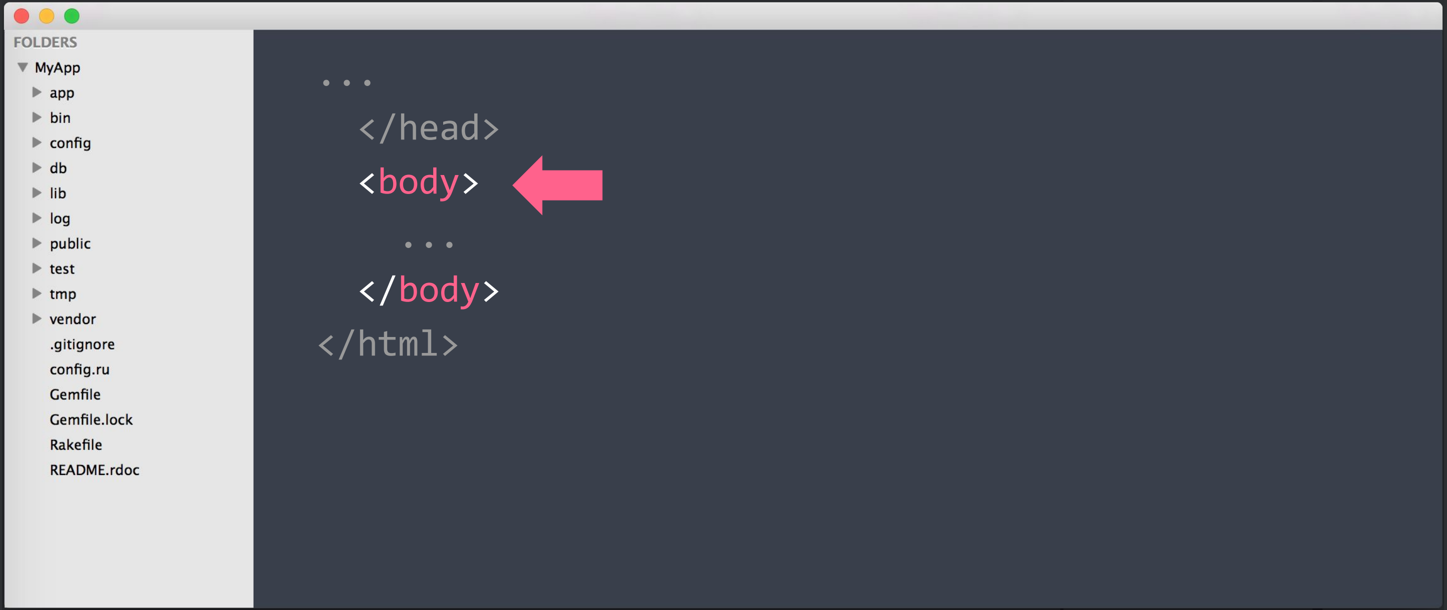
```
...
  <head>
    <meta charset="UTF-8">
    <title>Page title</title>
    <meta name="viewport" content="width=device-
width, initial-scale=1">
    <link rel="stylesheet" href="theme.css">
  </head>
  <body>
```

IRON HACK

@xharekx33

The <body> tag

Defines the section where the main content of the page will be placed. It contains everything that will be displayed on the viewport.



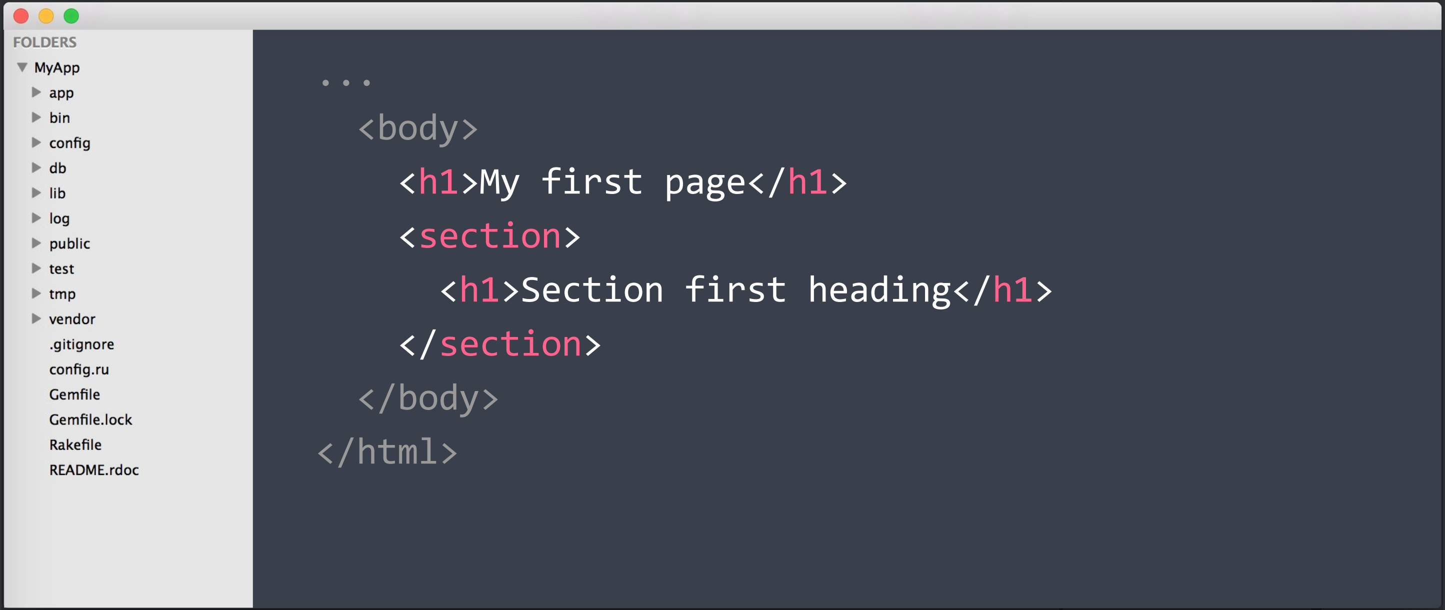
```
...
  </head>
  <body>

    ...

  </body>
</html>
```

IRON HACK

@xharekx33

Basic tags: Headings - <h1> to <h6>

The <h1> to <h6> tags are used to define text-only labels for sections of content on the page, defined implicitly or explicitly.

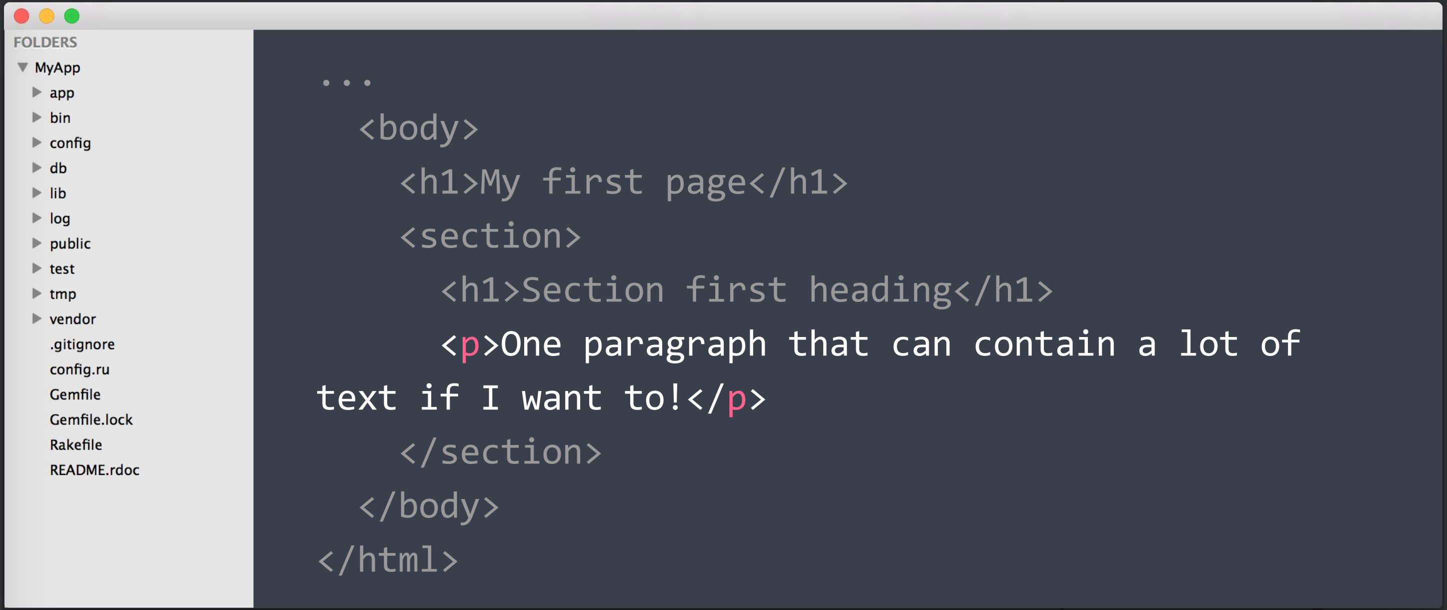


```
...
  <body>
    <h1>My first page</h1>
    <section>
      <h1>Section first heading</h1>
    </section>
  </body>
</html>
```

@xharekx33

Basic tags: Paragraphs - <p>

Used to define paragraphs. Browsers will automatically add space before and after to separate them from the rest of content.

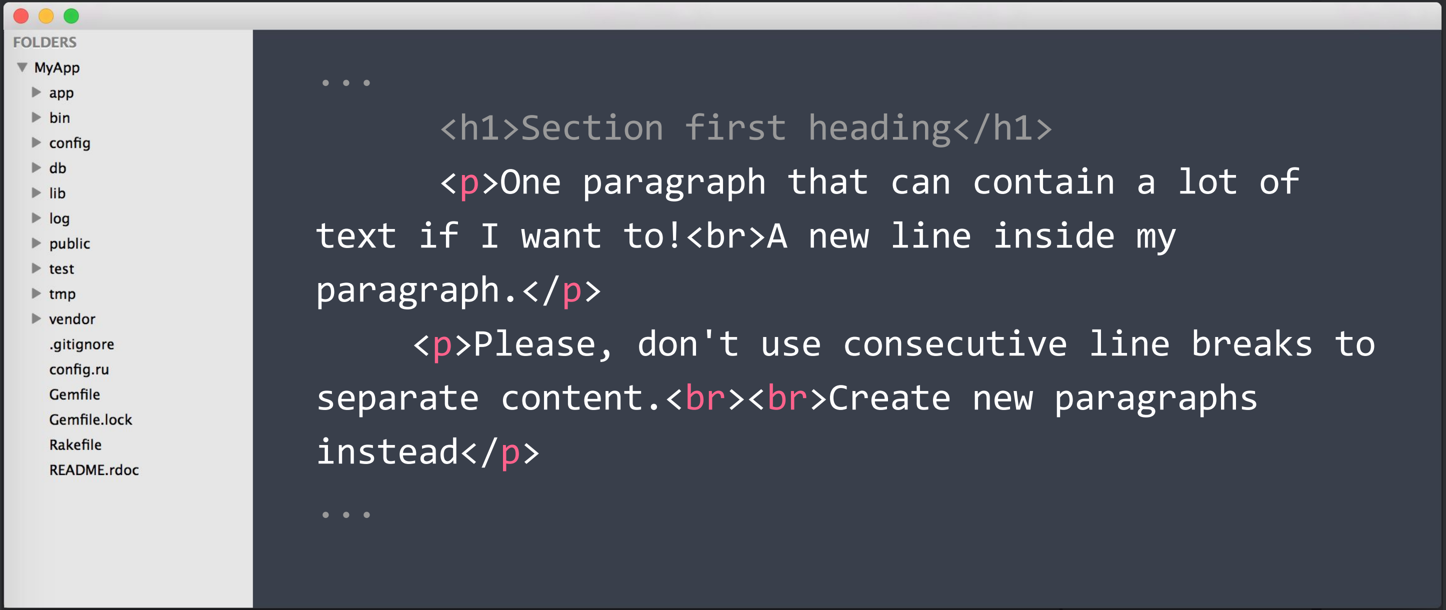


```
...
  <body>
    <h1>My first page</h1>
    <section>
      <h1>Section first heading</h1>
      <p>One paragraph that can contain a lot of
text if I want to!</p>
    </section>
  </body>
</html>
```

@xharekx33

Basic tags: Line breaks -

Line breaks are used to make text jump to the next line, usually for formatting purposes.



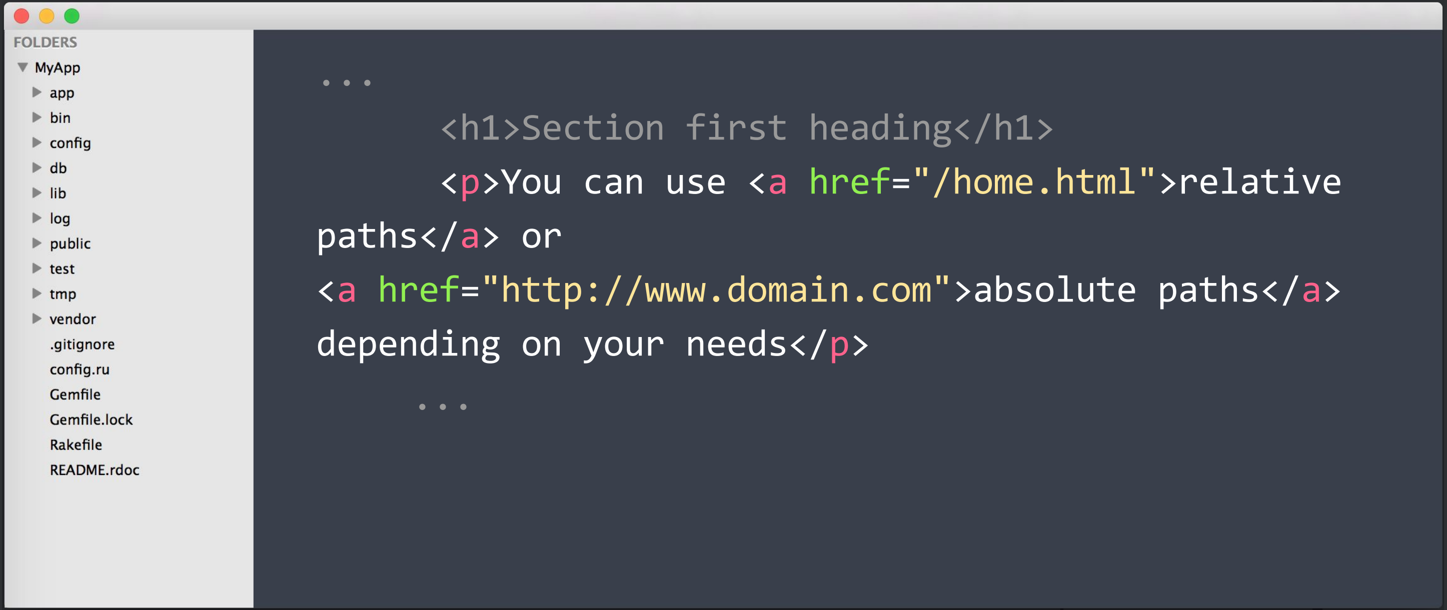
```
...
     <h1>Section first heading</h1>
     <p>One paragraph that can contain a lot of text if I want to!<br>A new line inside my paragraph.</p>
    <p>Please, don't use consecutive line breaks to separate content.<br><br>Create new paragraphs instead</p>
...
```

IRON HACK

@xharekx33

Basic tags: Links - <a>

The cornerstone of the web. They create hypertext links.

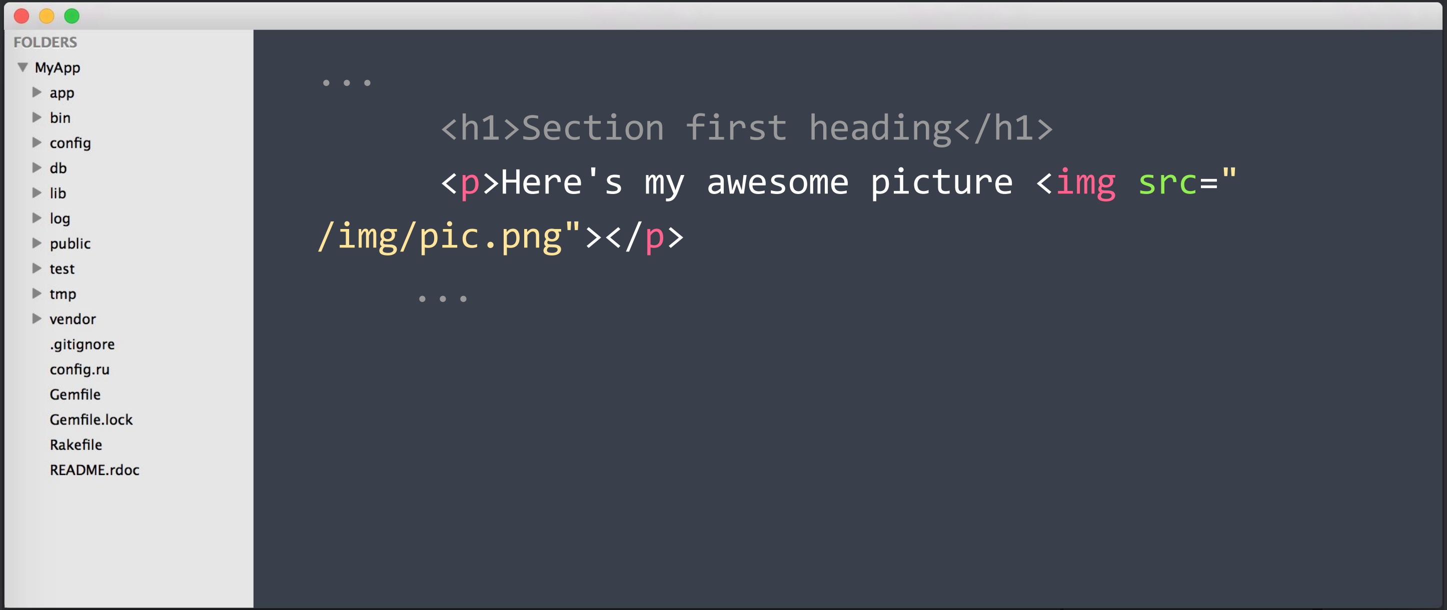


```
...
      <h1>Section first heading</h1>
      <p>You can use <a href="/home.html">relative
paths</a> or
<a href="http://www.domain.com">absolute paths</a>
depending on your needs</p>
      ...
```

@xharekx33

Basic tags: Images - <img>

Defines an image in the page.



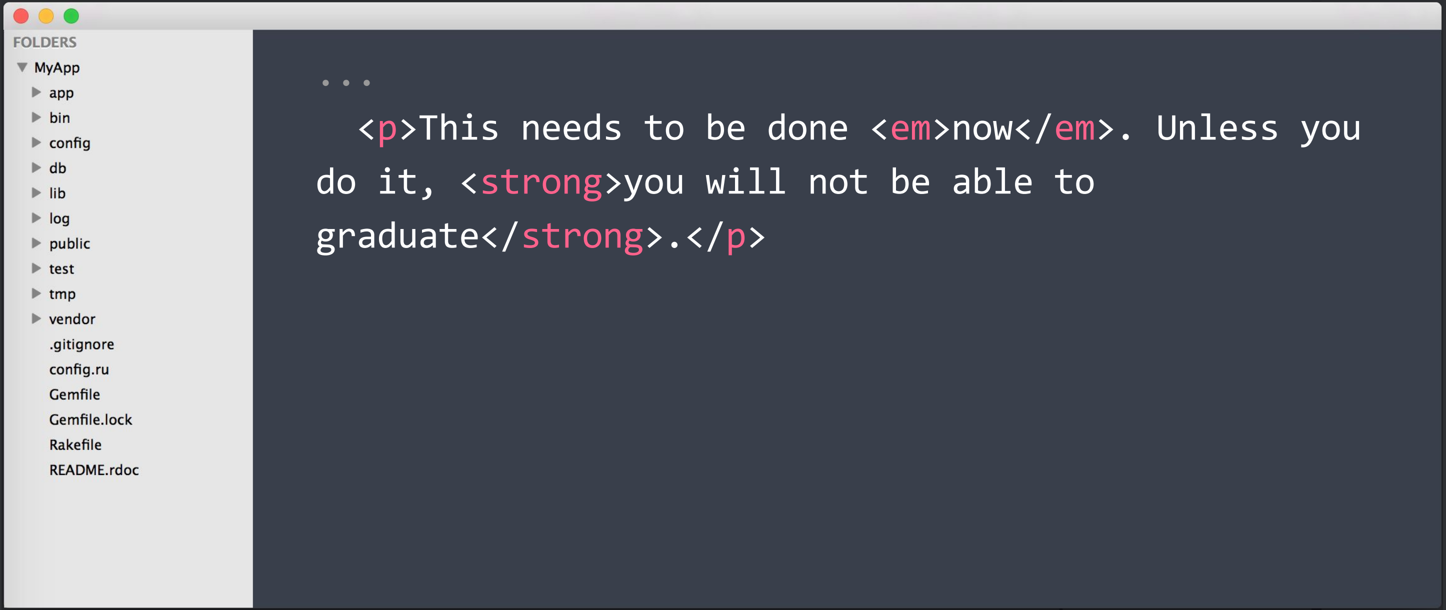
```
...
      <h1>Section first heading</h1>
      <p>Here's my awesome picture <img src="
/img/pic.png"></p>
     ...
```

IRON HACK

@xharekx33

Basic tags: Emphasis - <em> and <strong>

<em> is used to stress emphasis of its contents and <strong> to stress importance.



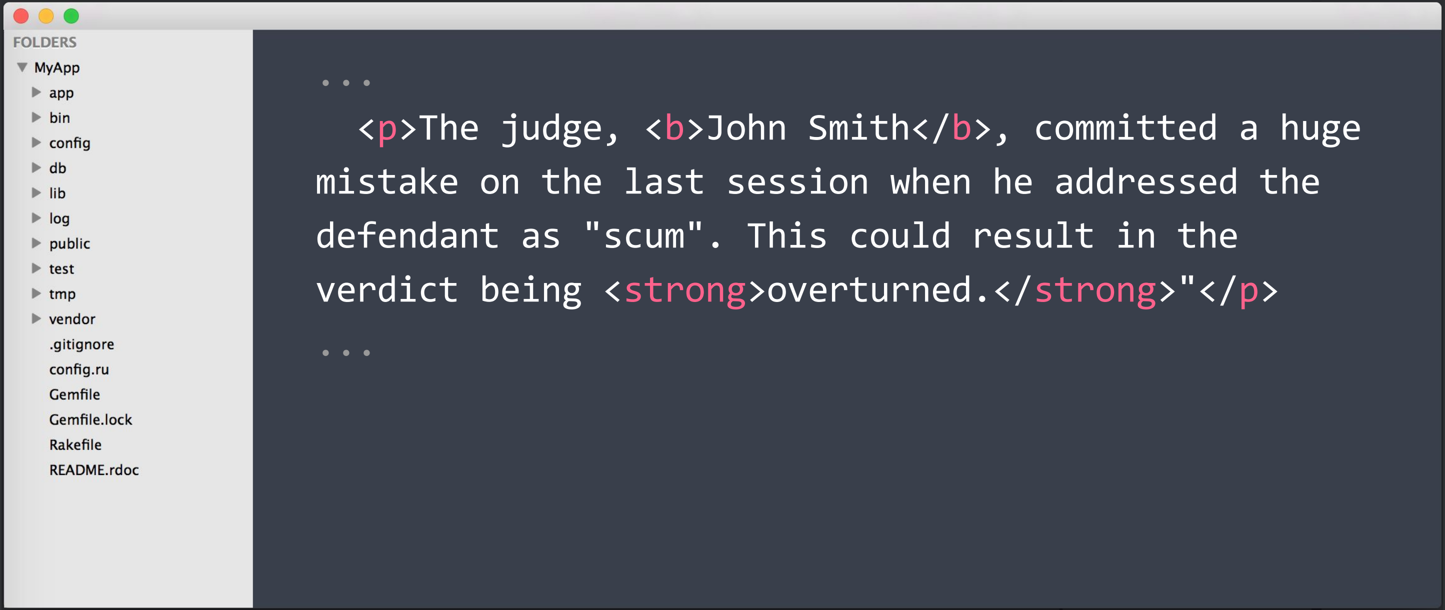
```
...
  <p>This needs to be done <em>now</em>. Unless you
do it, <strong>you will not be able to
graduate</strong>.</p>
```

IRON HACK

@xharekx33

<em> and <strong> VS <i> and <b>

Although by default they may look the same stylistically, <i> and <b> are strictly presentational, while <em> and <strong> carry semantic meaning

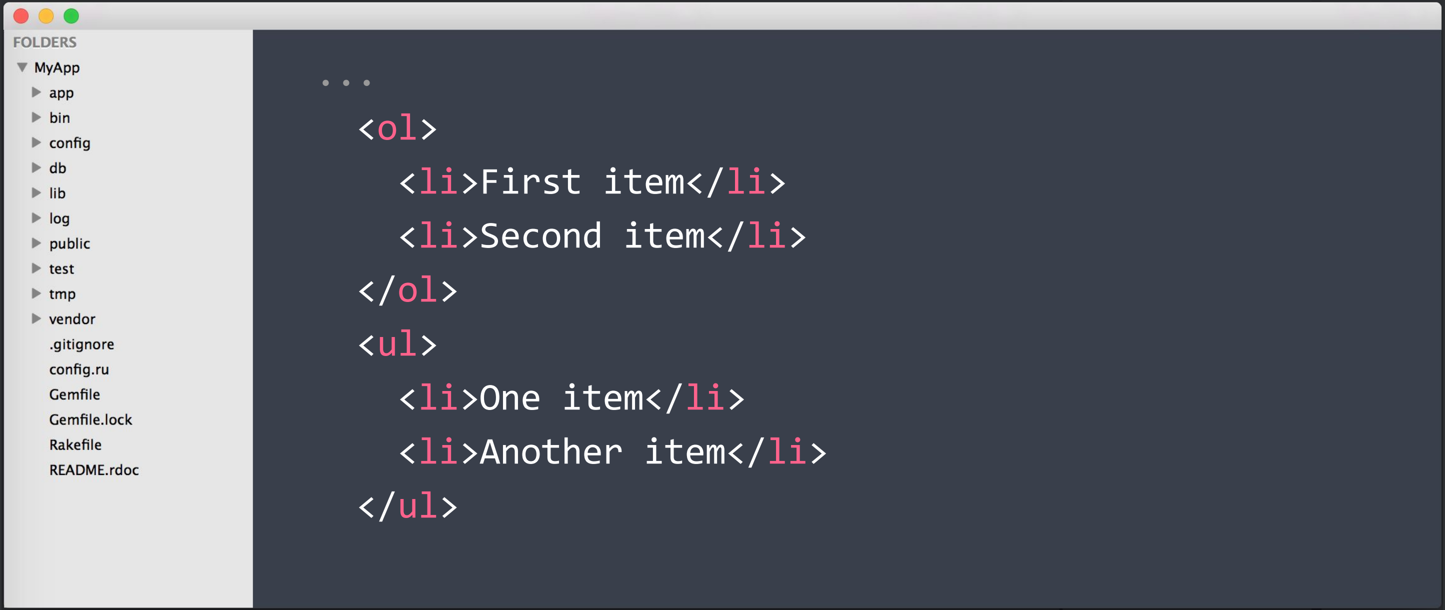


```
...
  <p>The judge, <b>John Smith</b>, committed a huge
mistake on the last session when he addressed the
defendant as "scum". This could result in the
verdict being <strong>overturned.</strong>"</p>
...
```

@xharekx33

Basic tags: Lists - `<ol>` and `<ul>`

Define ordered and unordered lists. Use this instead of adding numbers or bullet points by hand.



```
FOLDERS
▼ MyApp
  ▶ app
  ▶ bin
  ▶ config
  ▶ db
  ▶ lib
  ▶ log
  ▶ public
  ▶ test
  ▶ tmp
  ▶ vendor
    .gitignore
    config.ru
    Gemfile
    Gemfile.lock
    Rakefile
    README.rdoc
```

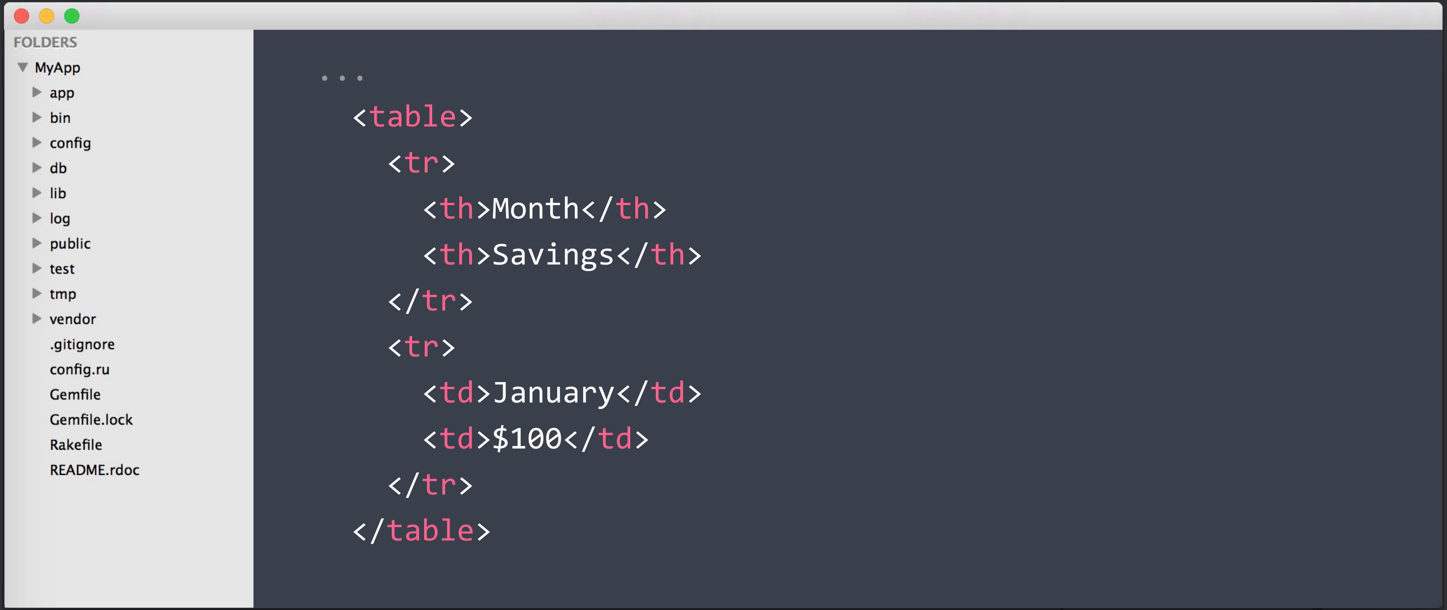
```
...
  <ol>
    <li>First item</li>
    <li>Second item</li>
  </ol>
  <ul>
    <li>One item</li>
    <li>Another item</li>
  </ul>
```

IRON HACK

@xharekx33

Basic tags: Tables - <table>

Defines a table to represent tabular data. DON'T use them to create layouts! This is not the 1990s!

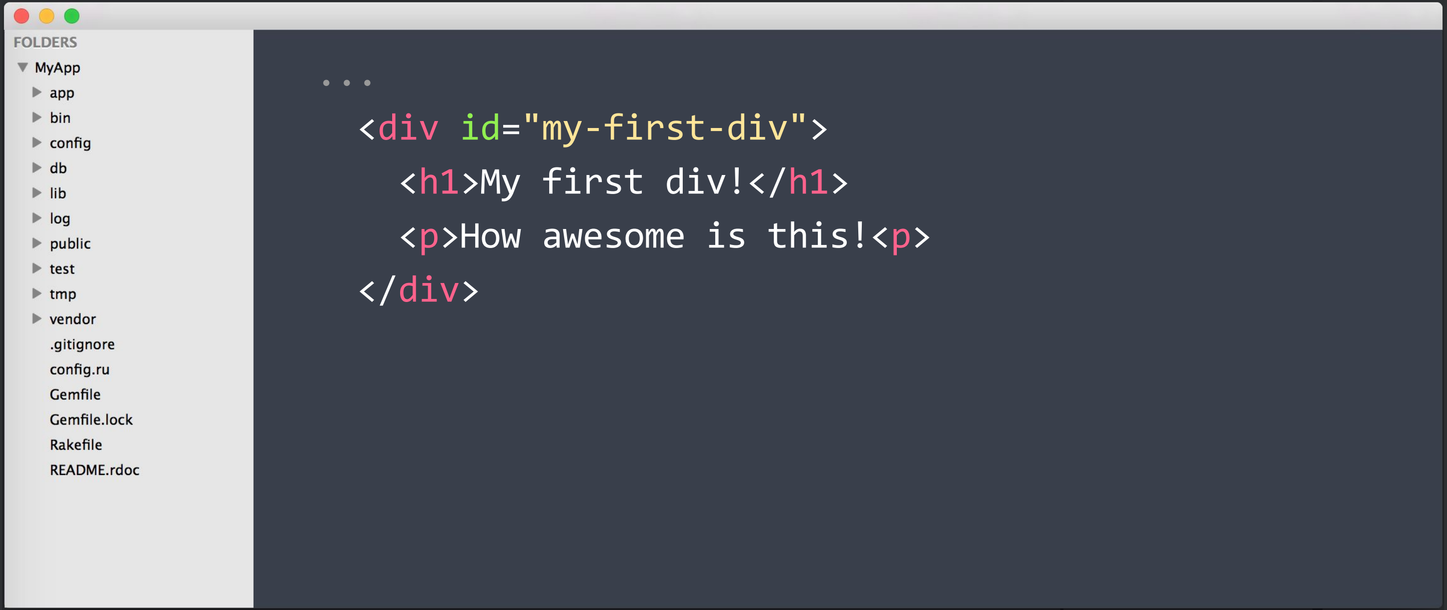


```
...
  <table>
    <tr>
      <th>Month</th>
      <th>Savings</th>
    </tr>
    <tr>
      <td>January</td>
      <td>$100</td>
    </tr>
  </table>
```

@xharekx33

Basic tags: Divs - `<div>`

Defines a division or a section in an HTML document. It's a meaningless container used to group block elements.



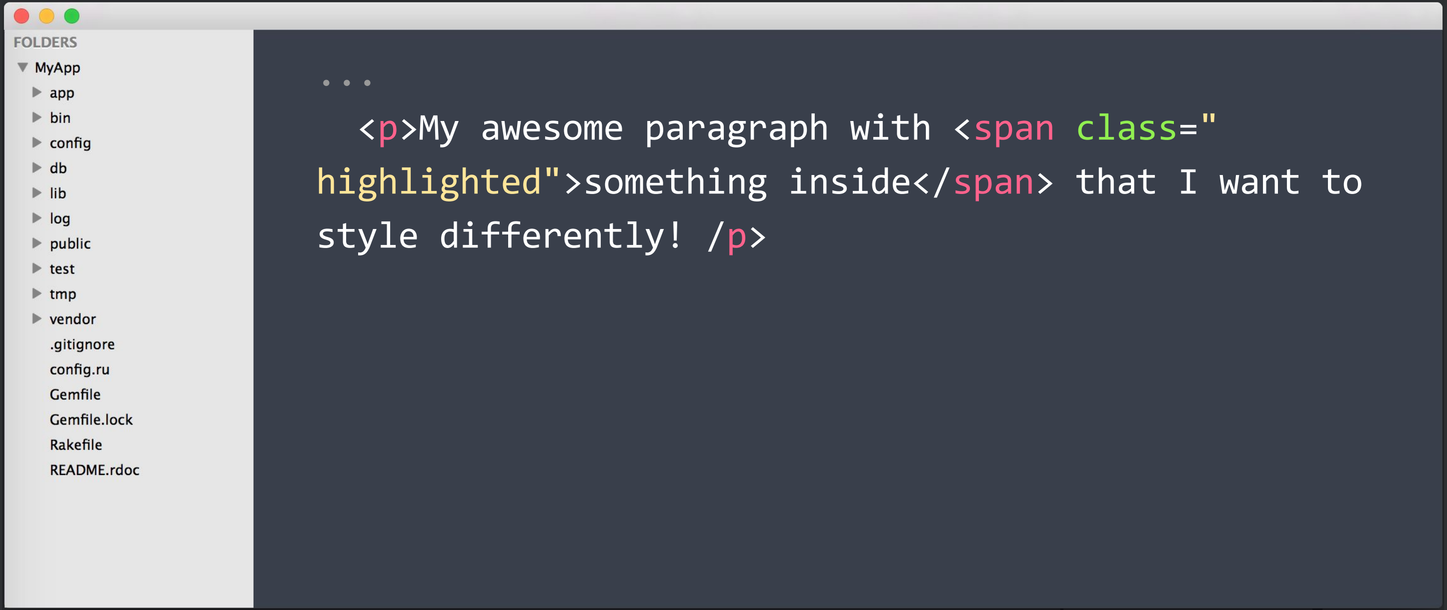
```
...
<div id="my-first-div">
  <h1>My first div!</h1>
  <p>How awesome is this!<p>
</div>
```

IRON HACK

@xharekx33

Basic tags: Spans - <span>

It is used to group inline-elements in a document for styling purposes or because they share common attributes.

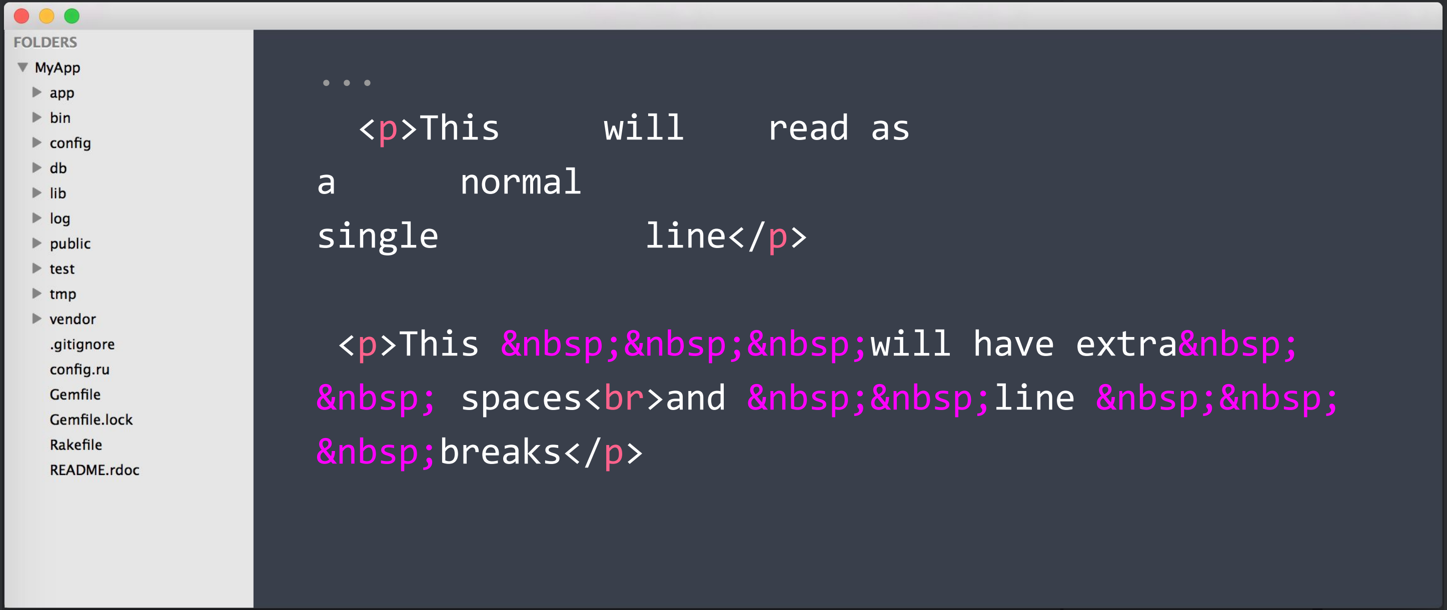


```
. . .
  <p>My awesome paragraph with <span class="
highlighted">something inside</span> that I want to
style differently! /p>
```

@xharekx33

HTML & whitespace: `&nbsp;`

HTML collapses consecutive spaces, tabs, and new lines into a single space. If you need to add whitespace use `&nbsp;` (non-breaking space).

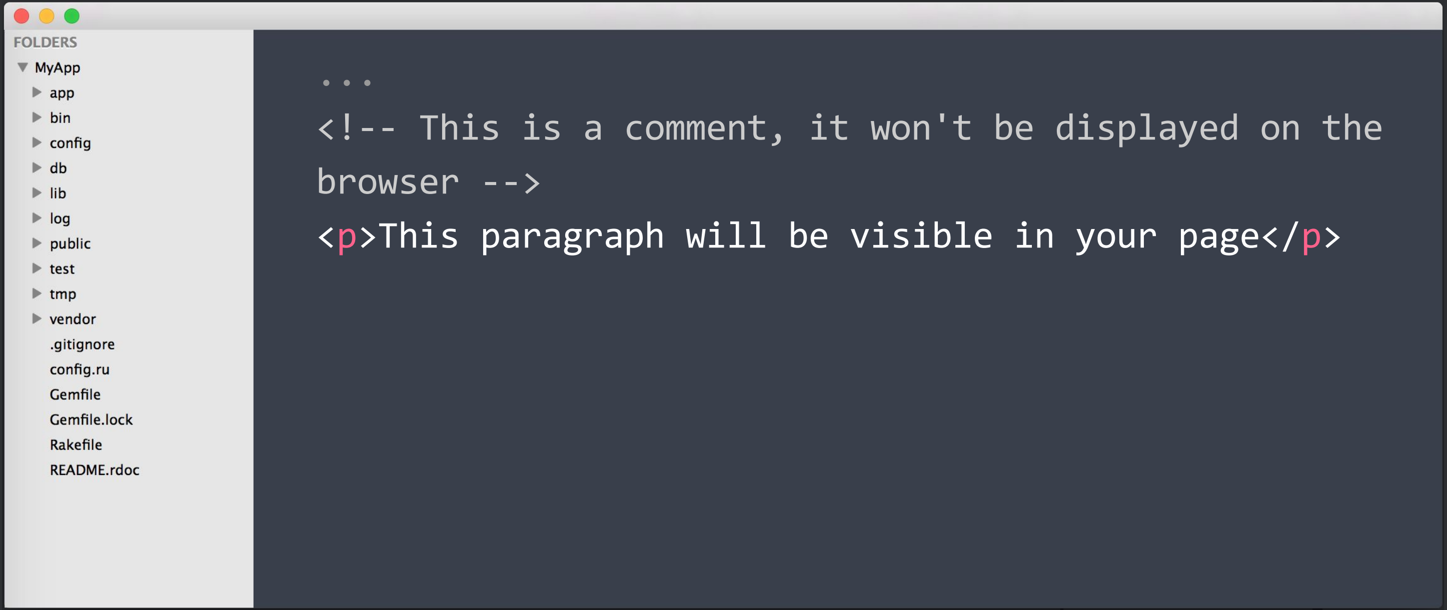


```
...
  <p>This     will    read as
a      normal
single          line</p>

 <p>This &nbsp;&nbsp;&nbsp;will have extra&nbsp;
&nbsp; spaces<br>and &nbsp;&nbsp;line &nbsp;&nbsp;
&nbsp;breaks</p>
```

@xharekx33

HTML comments

You can add comments to your code, to take notes, clarify stuff, but they will not be displayed on the page.



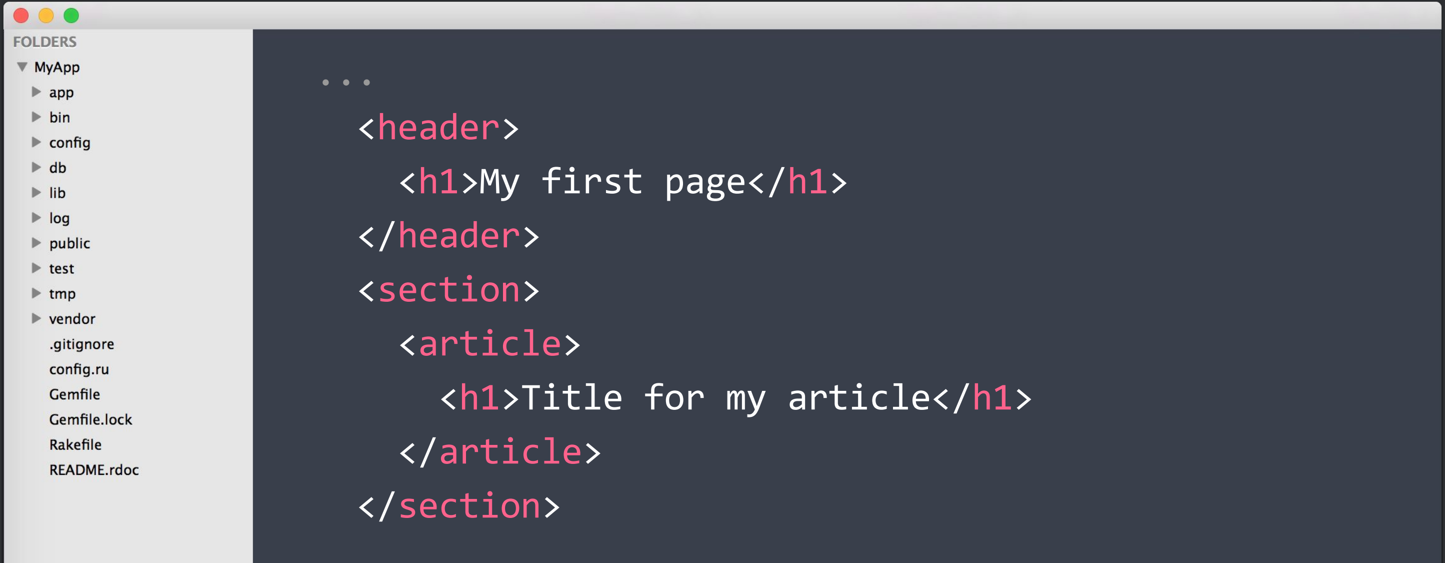
```html
...
<!-- This is a comment, it won't be displayed on the
browser -->
<p>This paragraph will be visible in your page</p>
```

IRON HACK

@xharekx33

HTML5 document structure

HTML5 provides a lot of tags to indicate structure and provide semantic meaning to your page: `<section>`, `<header>`, `<footer>`, `<aside>`. `<nav>` and `<article>`. Better than simple divs.



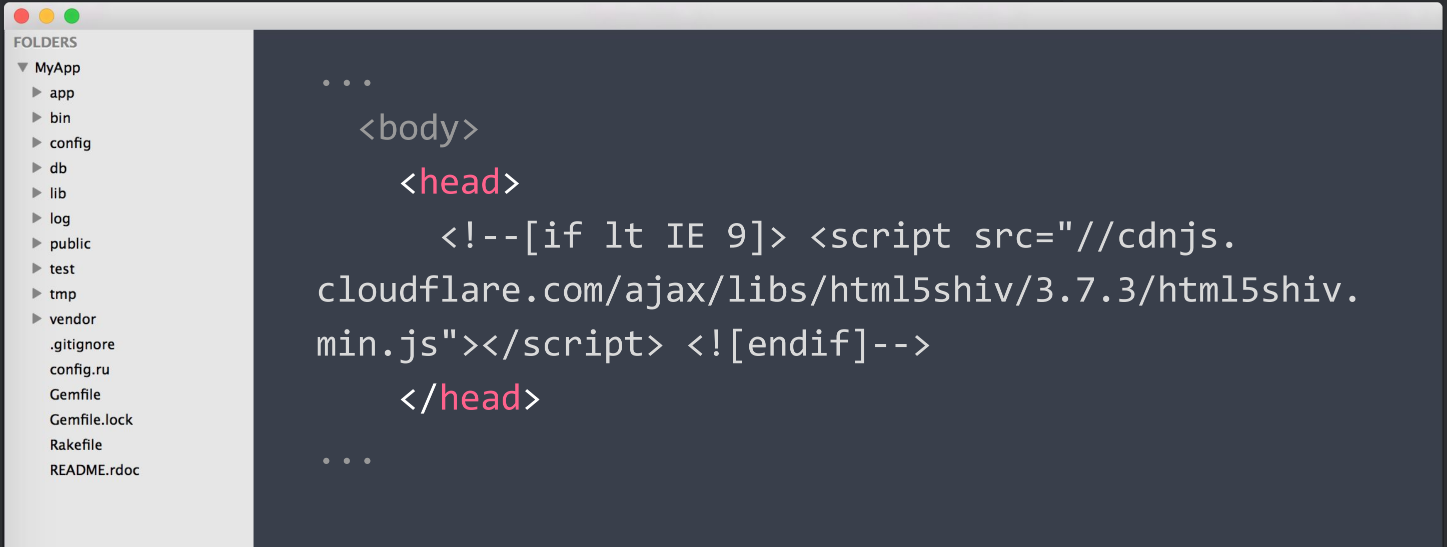
```
...
  <header>
    <h1>My first page</h1>
  </header>
  <section>
    <article>
      <h1>Title for my article</h1>
    </article>
  </section>
```

IRON HACK

@xharekx33

Browser support and the HTML5 shiv

Unfortunately old browsers do not support the use of these new HTML5 structural elements. In order to use and style them, we can include this code inside our `<head>` tag



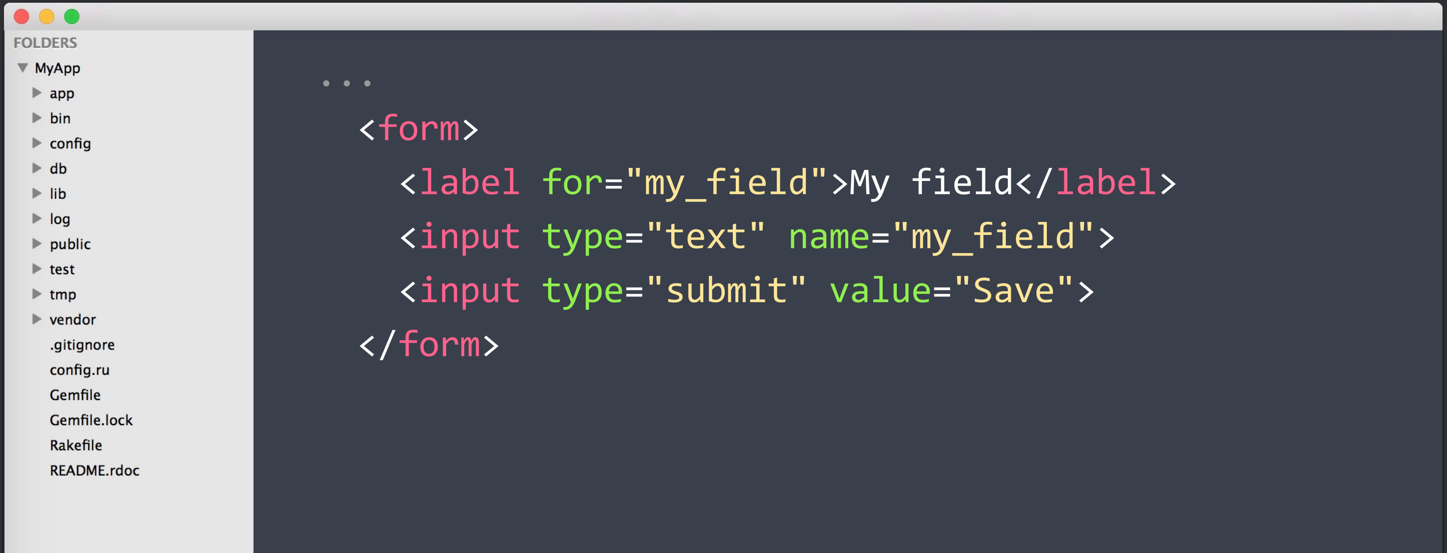
```
...
  <body>
    <head>
      <!--[if lt IE 9]> <script src="//cdnjs.
cloudflare.com/ajax/libs/html5shiv/3.7.3/html5shiv.
min.js"></script> <![endif]-->
    </head>
...
```

IRON HACK

@xharekx33

Forms

Forms allow users to send data to the website. Using forms, the website can take input from the site visitor and send it so it can be processed by a back-end application



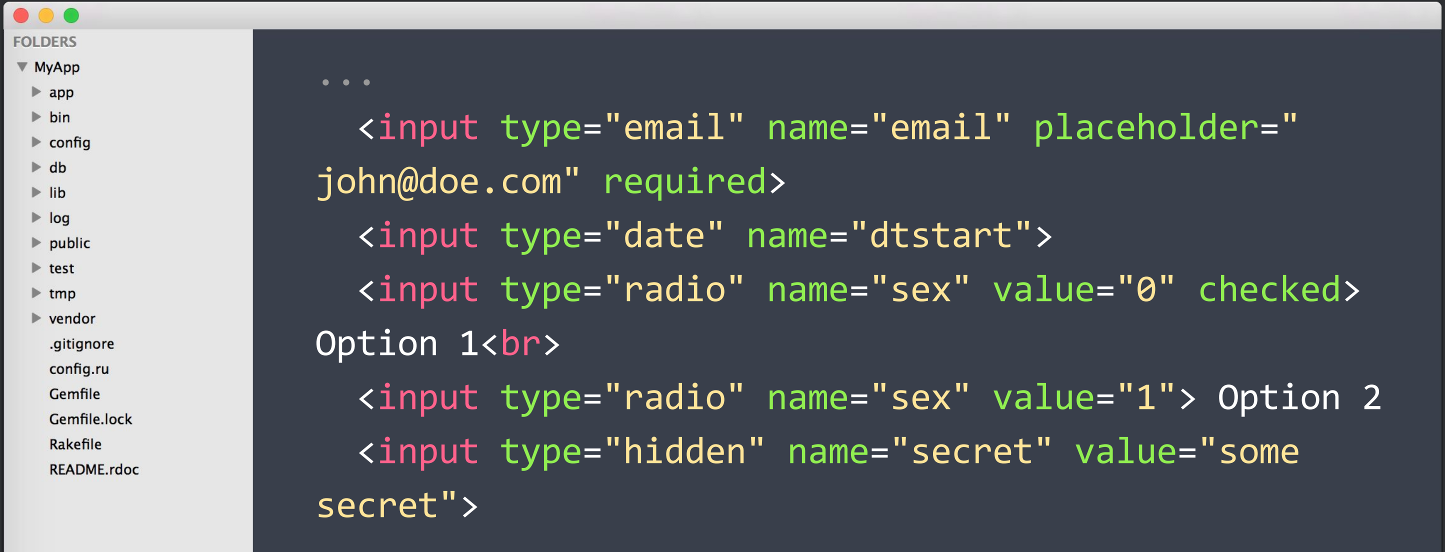
```
...
<form>
  <label for="my_field">My field</label>
  <input type="text" name="my_field">
  <input type="submit" value="Save">
</form>
```

IRON HACK

@xharekx33

Forms: field types

Forms support many kinds of different input types: text, button, checkbox, date, email, file, hidden, password, radio, search, submit...



```
...
  <input type="email" name="email" placeholder="
john@doe.com" required>
  <input type="date" name="dtstart">
  <input type="radio" name="sex" value="0" checked>
Option 1<br>
  <input type="radio" name="sex" value="1"> Option 2
  <input type="hidden" name="secret" value="some
secret">
```

IRON HACK

@xharekx33

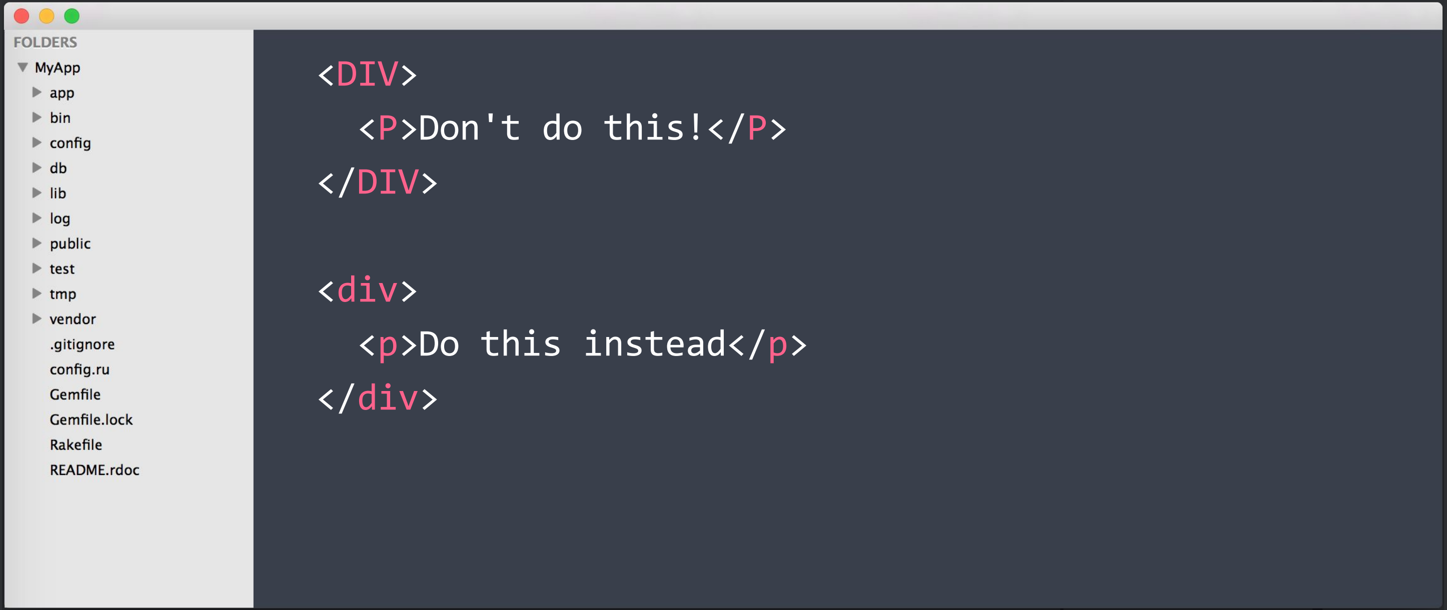
HTML and backwards compatibility

The good and bad thing about HTML is how flexible and backwards compatible it is. The browser will make its best attempt to render ANYTHING, no matter how mangled or outdated it is.

Part of this clusterfuck of spotty implementations, different coding styles and inconsistent rendering between browsers is a product of the history of HTML and its evolution: HTML 2.0 to HTML 4.01, XHTML 1 & 2, HTML5, the IETF, the W3C, the WHATWG...

Best practices

Use lowercase for all your elements



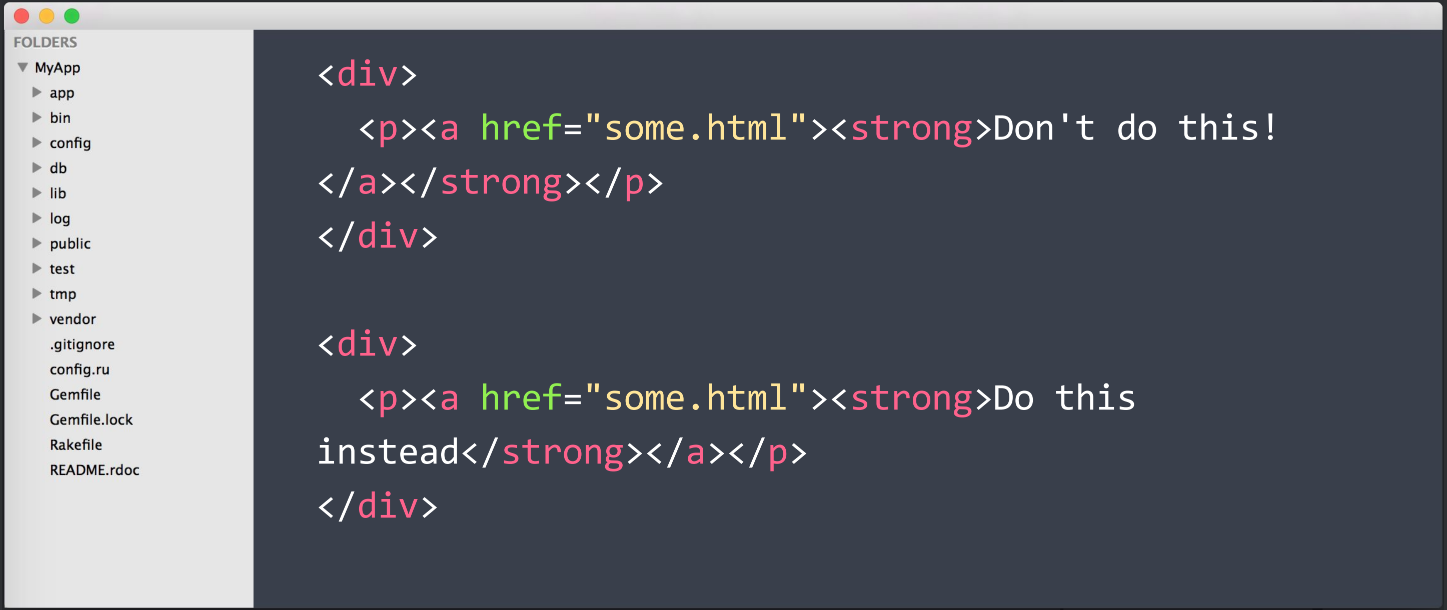
```
<DIV>
  <P>Don't do this!</P>
</DIV>

<div>
  <p>Do this instead</p>
</div>
```

IRON HACK

@xharekx33

Best practices

Use proper nesting!



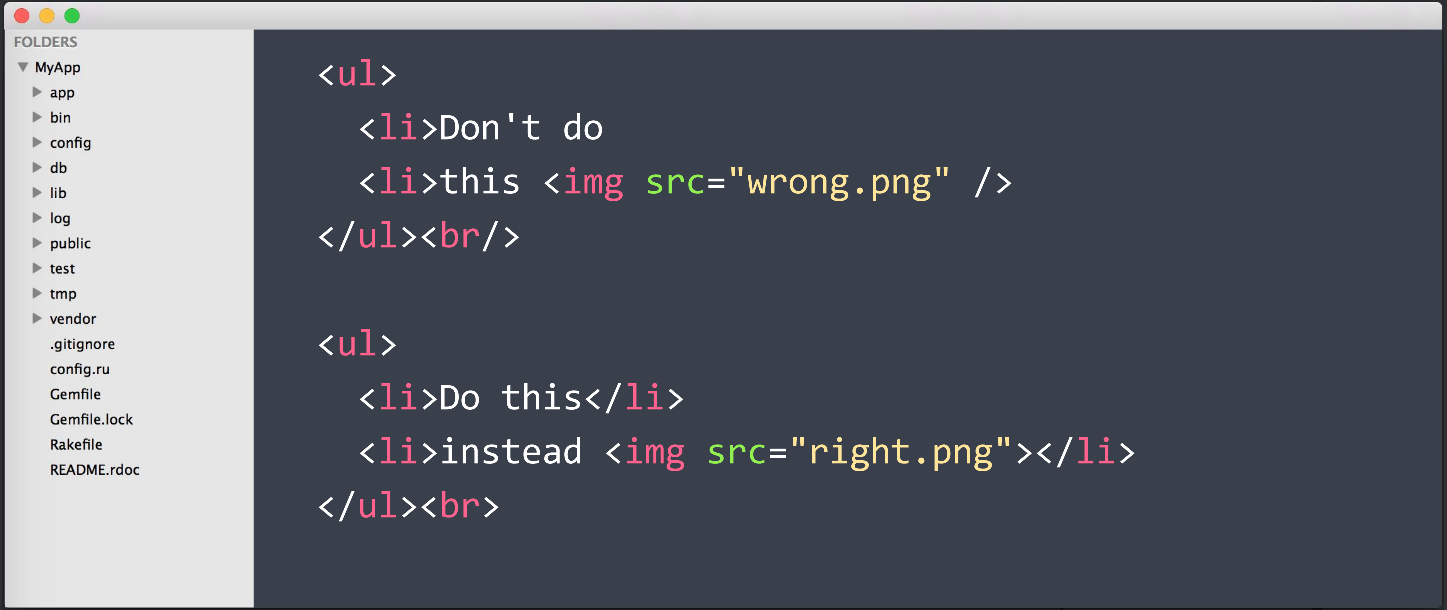
```
<div>
  <p><a href="some.html"><strong>Don't do this!
</a></strong></p>
</div>

<div>
  <p><a href="some.html"><strong>Do this
instead</strong></a></p>
</div>
```

IRON HACK

@xharekx33

Best practices

Always close your tags, but don't use trailing slashes on self-closing elements

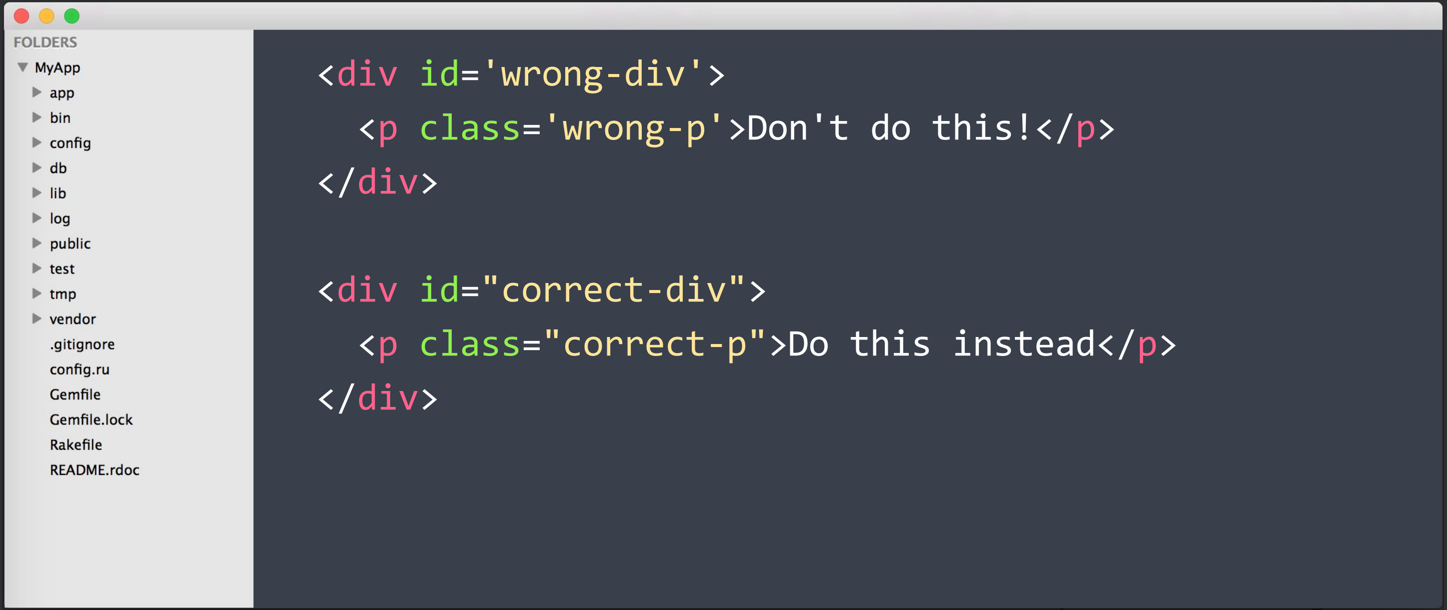


```
<ul>
  <li>Don't do
  <li>this <img src="wrong.png" />
</ul><br/>

<ul>
  <li>Do this</li>
  <li>instead <img src="right.png"></li>
</ul><br>
```

@xharekx33

Best practices

Use double quotes for all your attributes.

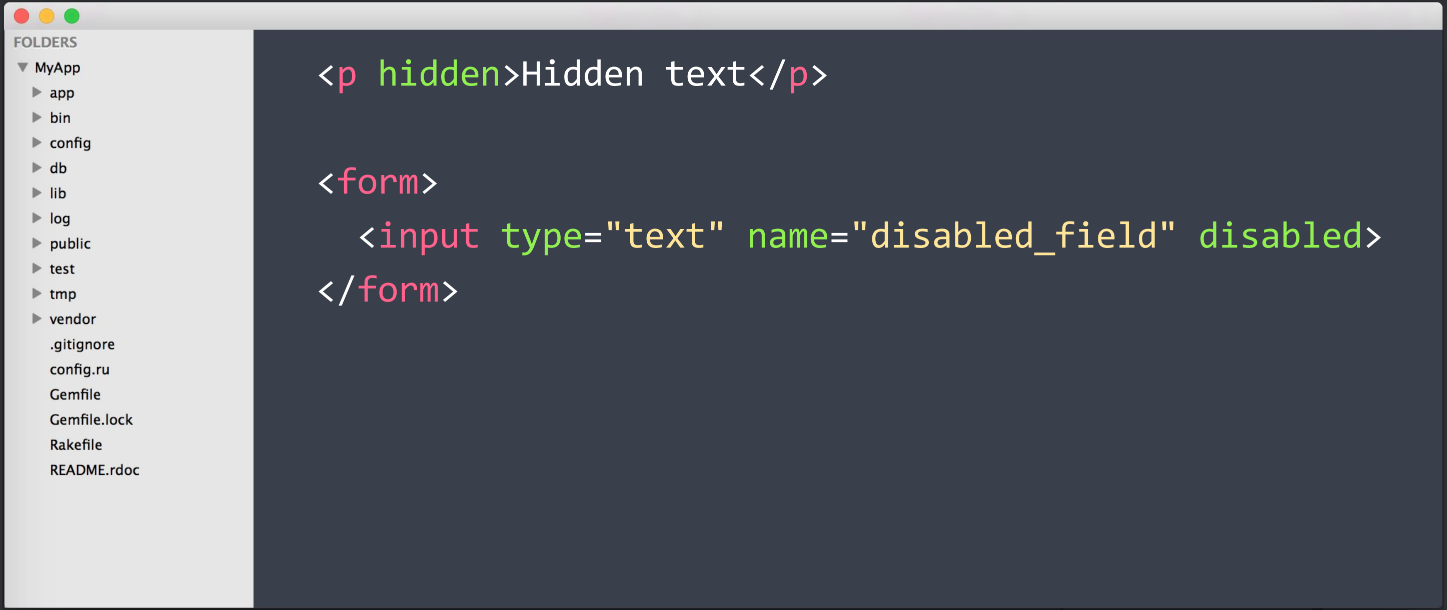


```html
<div id='wrong-div'>
  <p class='wrong-p'>Don't do this!</p>
</div>

<div id="correct-div">
  <p class="correct-p">Do this instead</p>
</div>
```

@xharekx33

Best practices

Boolean attributes (hidden, disabled, selected, checked...) don't need a value.
Their presence is enough.



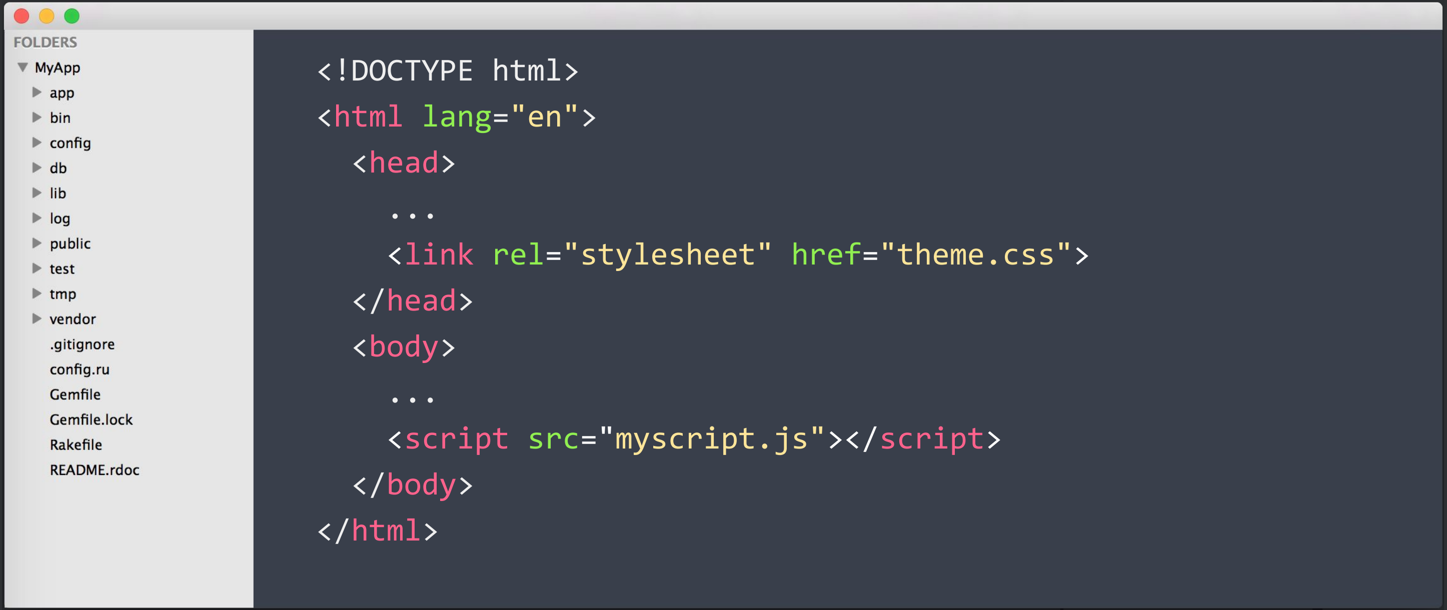
```html
<p hidden>Hidden text</p>

<form>
  <input type="text" name="disabled_field" disabled>
</form>
```

IRON HACK

@xharekx33

Best practices

Don't use inline css and load your javascript at the end.



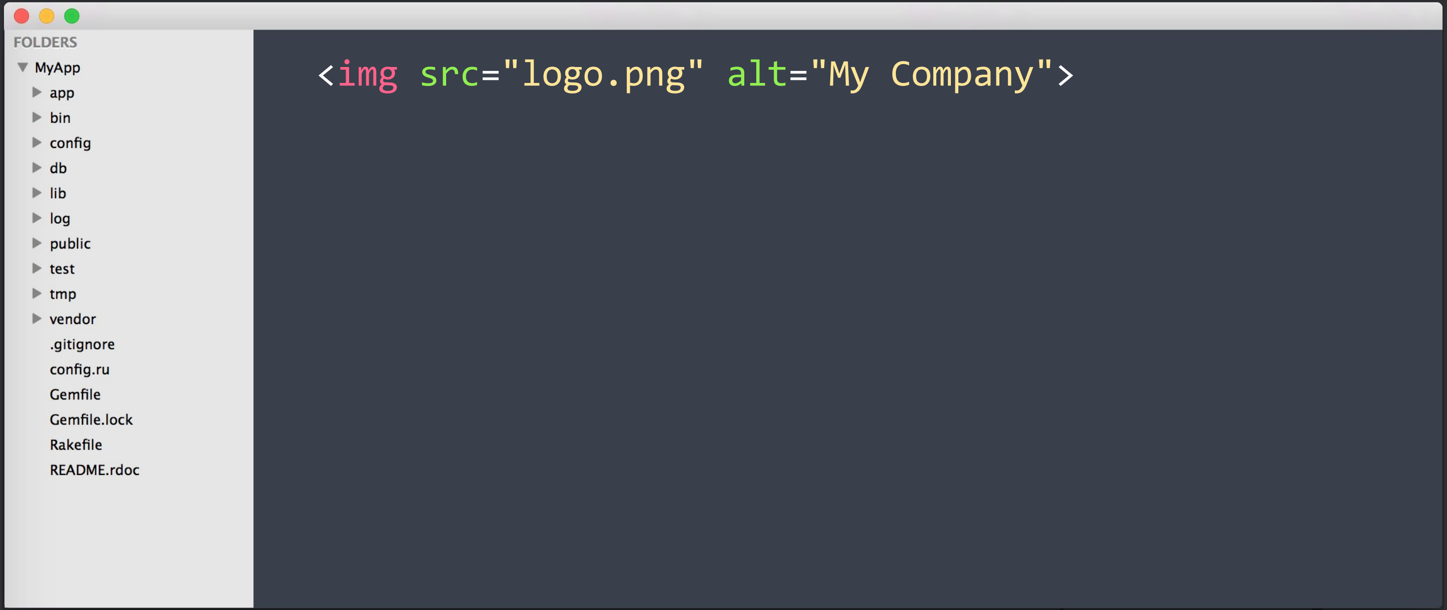
```
<!DOCTYPE html>
<html lang="en">
  <head>
    ...
    <link rel="stylesheet" href="theme.css">
  </head>
  <body>
    ...
    <script src="myscript.js"></script>
  </body>
</html>
```

IRON HACK

@xharekx33

Best practices

Use alt texts with images for screen readers and proper indexing.

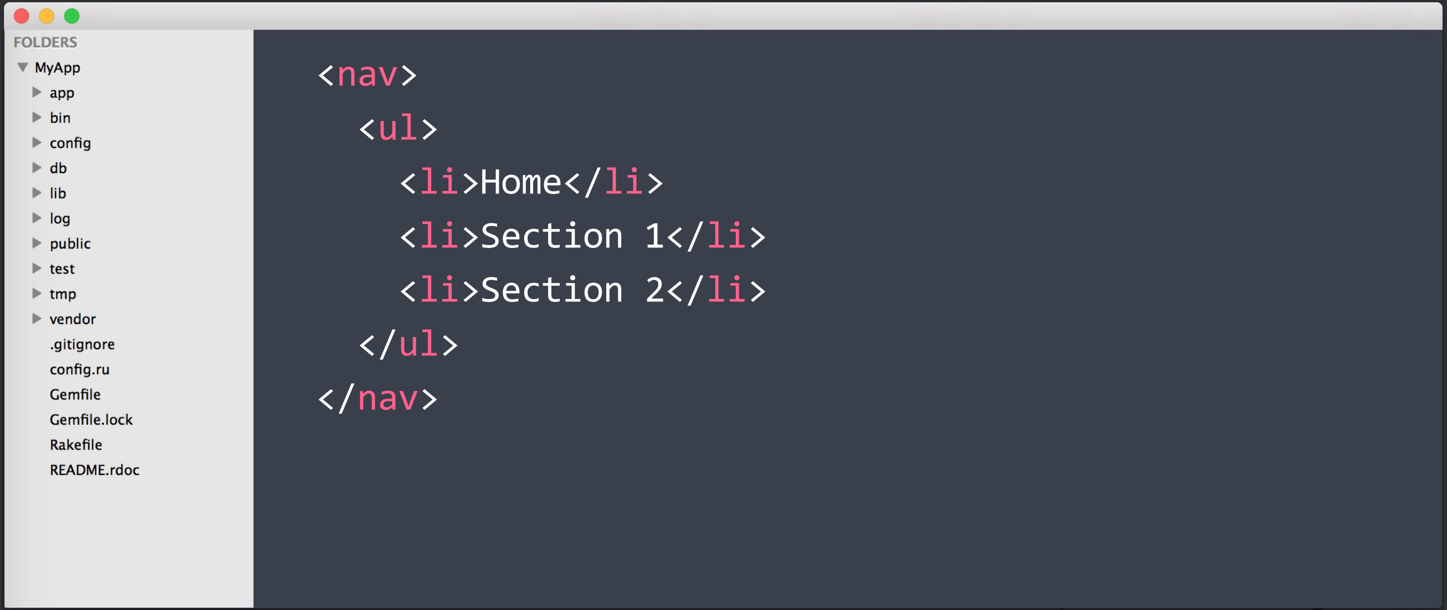


```
<img src="logo.png" alt="My Company">
```

@xharekx33

Best practices

Use ul for navigation.



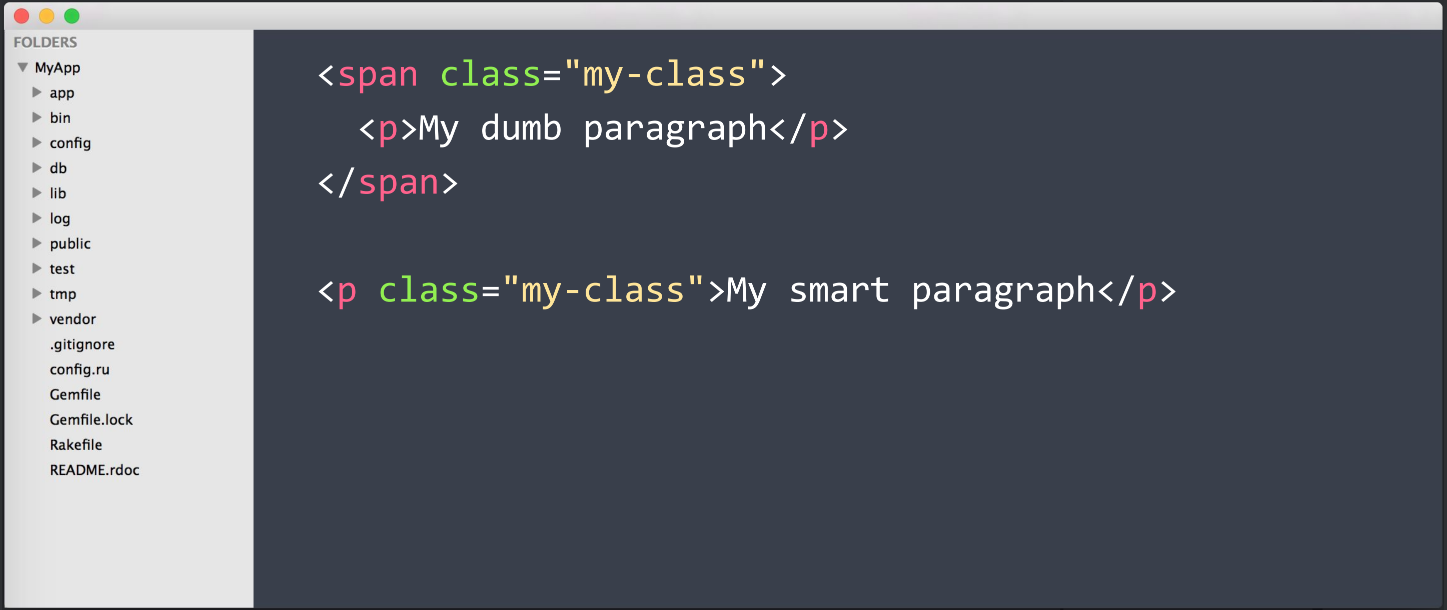
```html
<nav>
  <ul>
    <li>Home</li>
    <li>Section 1</li>
    <li>Section 2</li>
  </ul>
</nav>
```

IRON HACK

@xharekx33

Best practices

Don't use superfluous parent elements.

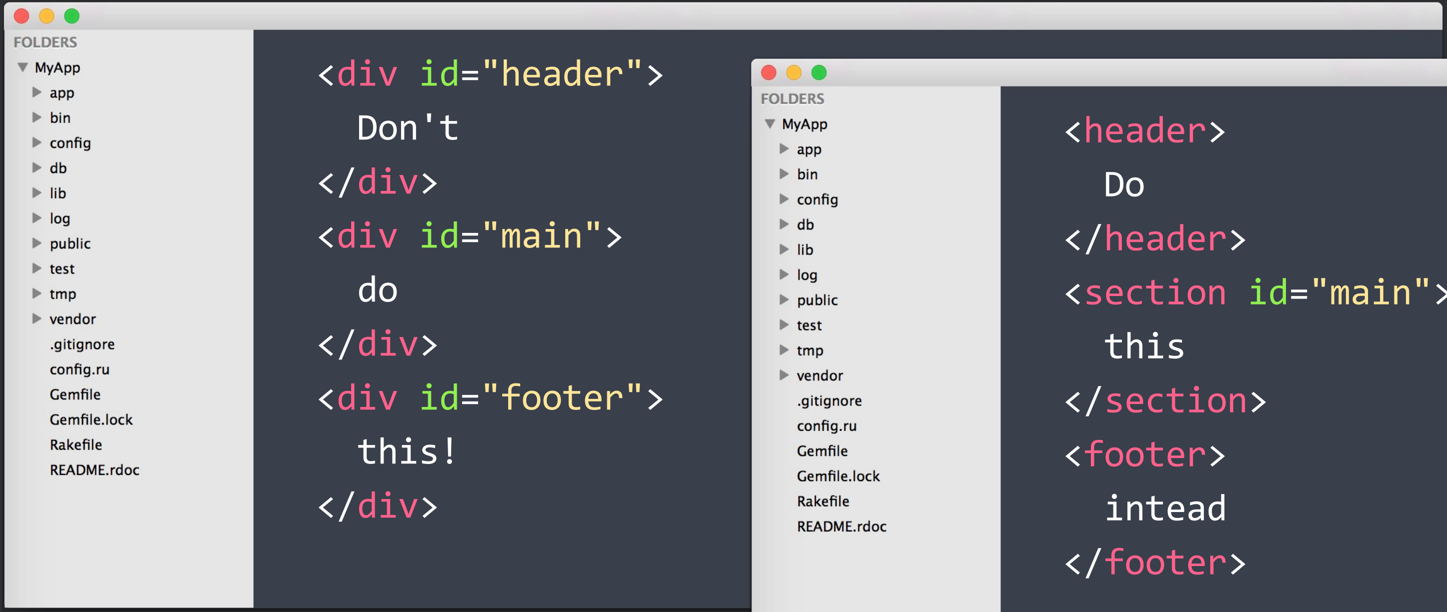


```
<span class="my-class">
  <p>My dumb paragraph</p>
</span>

<p class="my-class">My smart paragraph</p>
```

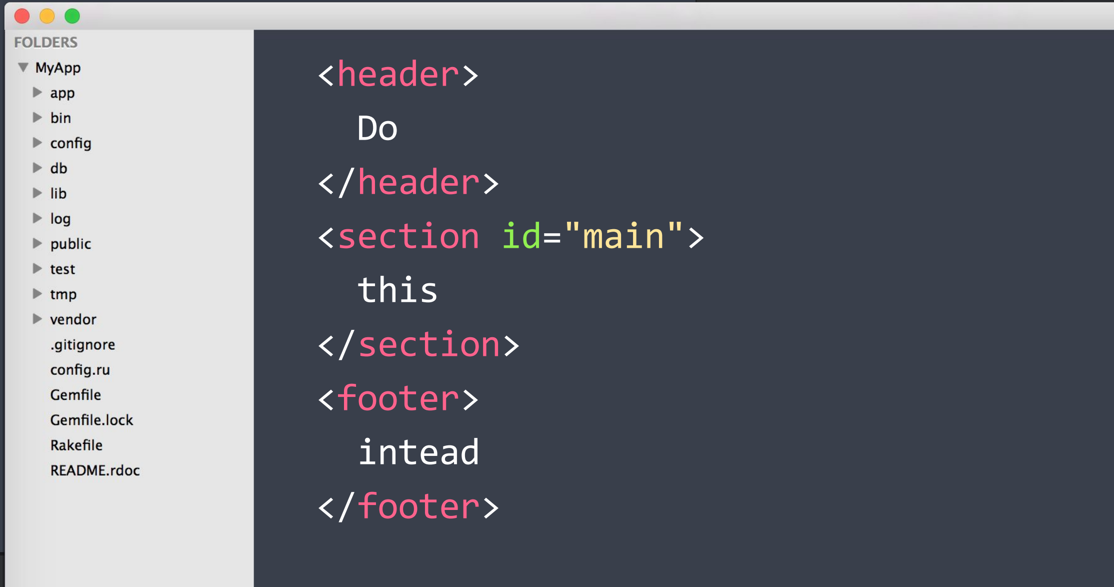
@xharekx33

Best practices

Prevent div-itis! Use the correct structural element for each case.



```
<div id="header">
  Don't
</div>
<div id="main">
  do
</div>
<div id="footer">
  this!
</div>
```



```
<header>
  Do
</header>
<section id="main">
  this
</section>
<footer>
  intead
</footer>
```

Exercise

Create markup for a blog post with a comment form and a couple comments:

- Use different and appropriate containers for the blog post, the comment form and the comments.
- The blog post must have: title, subtitle, date, author (with a link to their website), text, and image, a list, and some pull quotes.
- The comment form must have: name field, email field, space to write, a "remember my data" option and a submit button.

IRON HACK