Seminar Work
SBWL BIS

HTML5

Martina Oppermann

Beurteiler: Univ. Prof. Dr. Rony G. Flatscher

Wien, am 15. Dezember 2022

WIRTSCHAFTS
UNIVERSITÄT
WIEN VIENNA
UNIVERSITY OF
ECONOMICS
AND BUSINESS

Eigenständigkeitserklärung

für Seminararbeiten, Aufgaben, Reflexionen, Prüfung

Lehrveranstaltungsnummer: 0082 **Semester:** 2022/23

Lehrveranstaltung: BIS Seminar

LehrveranstaltungsleiterIn:
ao.Univ.Prof. Dr. Rony G. Flatscher

VerfasserIn: Martina Oppermann

Matrikelnummer: h11904961

Ich versichere / stimme zu:

1. dass ich die hochgeladenen Arbeiten, Aufgaben, Reflexionen sowie die Prüfung selbstständig verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt und mich auch sonst keiner unerlaubten Hilfe bedient habe.
2. dass ich dieses Thema bisher weder im In- noch im Ausland (einer Beurteilerin/einem Beurteiler zur Begutachtung) in irgendeiner Form als Prüfungsarbeit vorgelegt habe.

Mit der Unterschrift nehme ich zur Kenntnis, dass falsche Angaben studien- und strafrechtliche Konsequenzen haben können.

____15.12.2022______ Martina Op____________

Datum Unterschrift

Table of Content

Table of Listing

List of Tables

Table of figures

1 Introduction

Everyone uses it every day but most of the people don’t know that HTML is integrated into their daily life. Searching for some information for the next trip or visiting a specific website – HTML is a constant companion. HTML allows an amount of possibilities to create a personal website with different features predominantly used in the world wide web. For that reason, the following paper deals with this important topic. The aim of this work is to get a better comprehension of this powerful markup language.

The paper is organized into four main sections. The first section gives a brief overview of the general information about HTML5 which separates into background information, history and the difference between HTML and HTML5. This paper will provide a deeper view into the development and the conflicts. Furthermore, it gives interesting information about the group WHATWG and the W3C (web wide web consortium). The next chapter deals with the different kinds of elements of HTML5 especially the news of HTML5 and their usage. In the third section, two examples present the possibilities of this markup language and will enable a better understanding. Thanks to these examples and the information of the chapter above, the reader will be able to create its own website with HTML5. The conclusion with an outlook are drawn in the final section.

2 General Information about HTML5

This chapter allows the reader to get basic information about the markup language and its development. Furthermore, it gives a brief overview of the HTML history and mentions the main differences between the beginning version HTML and the actual version HTML5.

2.1 Background Information

HTML5 is based on HTML which means “Hypertext Markup Language”. It is a markup language that has the task to describe the rough and soft structure of a website (Wesolowski & Scheel, 2013). Very important is that HTML is developed only for the reason to structure a document and not for formatting. For formatting exists the program CSS (Cascading Style Sheets) which enables a simplified implementation of the presentation of the website (selfhtml, 2022). HTML is using a clear words format that can be edited by different text editors and so it is not bound to a specific software product. Furthermore, this markup language is not binded to any license (Münz & Gull, 2011, p. 21f). HTML5 is using the DTD which is based on SGML but it enables the omission of some closing tags, the definition of exclusions and that the upper and lower case is not important anymore (Flatscher, 2022, p. 4). For using the actual versions of web browsers like Internet Explorer, Firefox, Safari, Google Chrome and Opera is applicable. For older web browser such as Internet Explorer 7 or 8 there is the possibility to use the JavaScript-library “modernizr” for a faultless presentation of the website (Wesolowski & Scheel, 2013).

The tasks of HTML5 consist of the structuration of texts with the binding of graphic and multimedia content in form of references. Furthermore, it is possible to create headlines, paragraphs, lists and tables. In this connection, HTML5 can separated a website into different parts beginning with the head, content, tail and navigation up to the article of the website. It is also possible to use different interfaces like CSS or JavaScript which allow designing HTML elements tailored to specific wishes (Münz & Gull, 2011, p. 21).

Another interesting detail of this markup language is the structure: it has a hierarchical structure that consists of head and content data. Headlines, lists, paragraphs, tables and graphics are part of the last one. For instance, the count list contains bullet points. The author of the book "HTML5 Handbuch" describes this structure as a tree-structure with fine forks. One of the important features is that it exists the possibility to define hyperlinks (Münz & Gull, 2011, p. 21 f).

2.2 History

The development of HTML5 was developed by WHATWEG and the W3C HTML5 Working Group. In 1993, the first HTML was published as an internet draft with elements like headings and paragraphs (Lubbers, Albers, & Salim, 2010, p. 1). With the version 2.0 a new official language standard started in the year 1995. The interesting detail is that the difference between version 1.0 and 2.0 is not significant. In the year 1997, HTML3.2 was published with the main focus on optic effects which was not the being of the language. Some years later, this feature was removed from the language. Some little corrections are made in version 4.1 which replaced the previous version 4.0 in the year 1999. The latest version which is based on HTML 4.01 and XHTML1.0 was published in 2012 (Münz & Gull, 2011, p. 21 f).

Tim Berners-Lee was the developer of HTML in 1989 with the aim that all researchers of the world are able to collaborate and share a high volume of data in an electronic way. To achieve this target, Berners-Lee used for his new language SGML (Standard Generalized Mark-up Language). His biggest success in HTML was the development of hypertext links. Because of the constant development and the increasing part of new web browsers some problems with the display arose. Websites which were created for a specific browser displayed partial or not at all in different browsers. In reference to solving this problem, Berners-Lee established the W3C (World Wide Web Consortium) in the year 1994. W3C decided to develop open standards for different web technologies including HTML in the hope to enable a free and available web for all people independently of their language, culture and hard- or software. In the year 1998, W3C published the first real standard for HTML – it was called HTML4 with a lot of new features like frames, complex forms and style sheets (Hoy, 2011, p. 50 f).

Subsequently, a brief excursus to an important milestone of the history of HTML namely the browser war is provided. With the development of Internet Explorer by Microsoft, this special time area in browser history started. The beginning of the fight about the market share was aroused. Both companies, Microsoft and Netscape, tried with different developing strategies to stand out against each other. Microsoft used the advantage that it owns an already widespread operating system and as a result, Microsoft connected Internet Explorer with its operating system. Because of this effect, Microsoft almost squeezed Netscape out of the market. Some years later, Netscape published its source code and made it open-source. At this point, the Navigator changed its name to Firefox. In the last years, the numbers of different browsers have grown and the browser war is not as aggressive anymore as in the years before. Nevertheless, it exists some hints of a new browser war (Handelsblatt, 2008).

The previous web was suitable for static documents and the access was only possible by computers. The time changed and also the desires of the users. They wanted access from different devices to the web, dynamic pages with the aim that these behave more like an application than a document and high-quality audio and video. To enable these wishes the W3C developed a new version of HTML named HTML5 (Hoy, 2011, p. 51). According to the WHATWG with participation of W3C, HTML5 should be the last version of HTML because future changes will be further developed in a living standard without any numbers of versions (selfhtml, 2022).

2.3 Difference between HTML4 and HTML5

Although HTML5 is the successor of HTML4, they have a big difference namely the kind of definition of the language. In the previous version SGML was used which uses DTD (Document Type Definitions) for describing its structure (Münz & Gull, 2011, p. 22). But HTML5 defines the language with DOM (Document-object-model) which has the advantage that it enables software engineer to navigate their structure, build documents and add, delete or modify elements and content (Robie, n.a.).

Furthermore, HTML5 enables a good presentation of the content in the web- and mobile browsers and, in contrast to HTML4, it does not offer following features: <font>, <strike>, <frame>, <table border> and <body background> (Wesolowski & Scheel, 2013). So, there are elements that are used to define the formatting task of HTML5. Nevertheless, HTML5 is backward compatible and so the HTML4 elements have to be supported. The last important difference is that HTML5 has a new Application Programming Interface that has the task to implement features like audio/video playback, drag&drop and offline use (von Kesteren & Pieters, 2011).

To sum up, HTML5 is fully supported by multimedia, it did some replacements of the elements of HTML4 and has more features than the previous version. The advantages of HTML5 are a simple and clear syntax, introduction of multimedia elements and improved access to the geographical location of the programmer through the use of JS GeoLocation. The next one is that the communication with the client-server, this is possible with web sockets that make the general process much simpler and last but not least, JavaScript runs in a separate thread. That means that JavaScript and the browser operate in two different threads (Everything, Position is, n.a.).

3 Elements of HTML5

This part of the paper will be concerned with the different elements of HTML5 from the basic element which are also important for other HTML versions to the specific and new elements of HTML5 like the text and structure elements. After every important element, the reader will find next to the description also an example to get a better comprehension of this element and its use.

3.1 Basic Elements

Before starting with the new Elements of HTML5 it is important to get an overview about the basic elements of HTML. Only with these basics it is possible to create an own website and it helps to get a better understanding for the examples in the following chapter.

A HTML element consists of an opening tag, the content of the HTML element and ended with a closing tag. The tag always opens with a clip (<) and closes with a second one (>) for instance <tag_name>. For the closing tag it is necessary to use a slash before the first clip (</) as </tag_name>. This concept makes it easier to parse the text and denote which kind of tag was used to mark it up (Flatscher, 2022, p. 2).

Comments are the next important element. Without them the programmer could lose the overview of their program codes. They are useful in order to keep an orientation about the different parts of the HTML codes, place intern information about the author and date of creation and give intern notes to specific text parts. The advantage of this element is that it will be ignored from the browser and so it will not be shown in the finished document. A comment starts with this symbol order " <!-- ". After this the programmer can write a text of any length and finish the comment with this end tag " --> ". Commands can be placed everywhere where normal elements can be standing and they cannot be nested (selfhtml, 2022). Listing 1 shows a normal use of the element "comment".

```
<!--this is a comment-->
```

Listing 1: Example comment

A website which complies with the standard set by W3C should include at least four elements. Before beginning with the fundamental element, it is necessary to mention the <doctype>-element. This one informs about the type of the document which has to be in the first row of the file without any space or rows before. This has the result that the standard modus is activated and the browser has to interpret the HTML5 (Abel, 2016, p. 23). The first one is the “html”-element which encloses the whole document and is the starting point of the tree structure. This is the reason why it is called root element (Selfhtml, 2022).

Here is a brief excursion to the tree structure and their elements. The root element is the ancestor of all other elements which would be used on a page. The other elements can be distinguished in “sibilings”, “children”, “parents” and “leaf” nodes. The “siblings” one of is the element which shares the same tree trunk such as two branches. In comparison, the “children” are elements of other elements and the return way of it called the “parent” node. So it can be imagined like a smaller branch that starts from a bigger branch. “Leaf” nodes are these elements that do not have any children (Pilgrim, HTML5: up and running, 2011, p. 33).

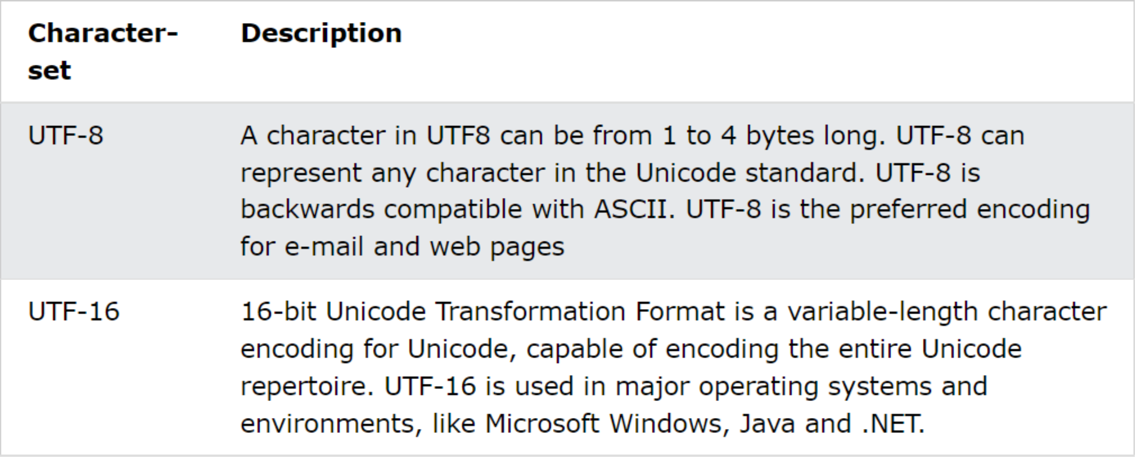
In the tags of the “html”-element exist the possibility to fix the language. The use of the attribute “lang” enables the fixation and the corresponding value is a language code by the IANA. These language codes have different groups of languages for instance “de” for German and subclasses of the language such as “de-CH” for the Swiss German language. Listing 2 shows the using of the attribute “lang”.

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

Listing 2: <html>-element with attribute "lang"

The next one is the <head> - element which is a necessary part of every HTML-standard and contains information about the text in the body part. It is the first child of the root element and contains metadata information of the page and not the content of this page (Pilgrim, Durschstarten mit HTML5, 2011, p. 36). In this element the <title> of a website, declare the character code, integration of extern files and general information about the document is defined.

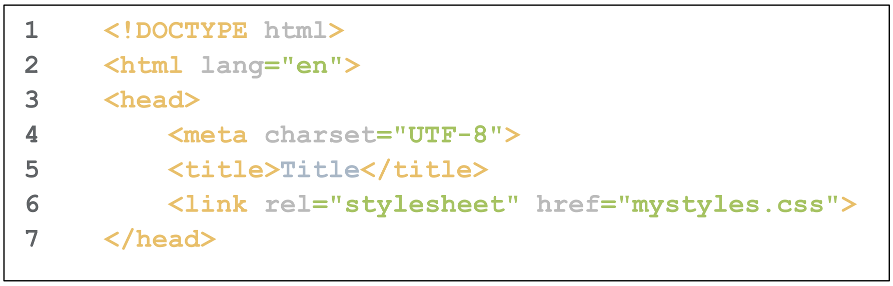
Apart from the title and some icons, the rest of the information stays invisible (Abel, 2016, p. 24). Regarding the character code in the <head>-element, it is important to understand what this element is working for. Character code is the assignment of linguistic signs to a computer-internal representation. This means that it defines how the text will be present on the display. It exists a “code-table” that defines which sign will be interpreted through which bit sequence. For integration this character code in the file, the programmer has to use <meta charset=”utf-8”> (Münz & Gull, 2011, p. 56f). In Figure 1, a shortcut of the most commonly used encoding character sets is provided.



Character-set	Description
UTF-8	A character in UTF8 can be from 1 to 4 bytes long. UTF-8 can represent any character in the Unicode standard. UTF-8 is backwards compatible with ASCII. UTF-8 is the preferred encoding for e-mail and web pages
UTF-16	16-bit Unicode Transformation Format is a variable-length character encoding for Unicode, capable of encoding the entire Unicode repertoire. UTF-16 is used in major operating systems and environments, like Microsoft Windows, Java and .NET.

Figure 1: UTF-8 vs. UTF-16 (available under https://www.w3schools.com/charsets/ref_html_utf8.asp)

The next subitem of the <head>-element is the <title> that specifies the title of the document. This text of the title is the title of the whole document and will be shown in the title bar of the browser. Furthermore, the <link>-element allows to integrate extern files in this document. Extern files could be scripts, pictures, symbols but the most common usage is to paste an extern CSS-file style. With <link rel=”stylesheet” href=”mystyles.css”> it is possible to integrate the CSS-file in the document. The attribute “rel” is standing for relation and means the relation between the document of the file that will be integrated and this website (Abel, 2016, p. 27f).



```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
    <link rel="stylesheet" href="mystyles.css">
</head>
```

Listing 3: Using the <head> with its attributes

The following part is dedicated to the <body> that is displayed in the browser. The attribute <p> starts a paragraph. The <div>-element is used to involve several elements in common areas like table, text and graphic. When using this element, the text starts in a new row (Selfhtml, 2022). More information about the structure of the body will be presented in the following point. Listing 4 shows the finished structure of a HTML document.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
    <link rel="stylesheet" href="mystyles.css">
</head>
<body>

</body>
</html>
```

Listing 4: a whole HTML document

3.2 Structure Elements

The usual layout of a website is separated in five different sections. Starting with the header on the top of the website, logos, the name, the subtitle and a brief description of the website are located there. It is followed by the navigation bar which is used to place a menu or a list with links for the navigation. This enables to offer the user an easy and fast guide to a different page or document. The next part shows the important contents and subdivides in some rows and columns. In Figure 2, this part is divided into the main information field and sidebar. As the name is already saying, the main information field is for the important information and links. Last but not least, in the bottom the institutional bar where the general information about the website, the author or the company with the rules is located, AGBs and so on is placed (Abel, 2016, p. 30 fff). Figure 2 shows this layout with the typical HTML5 elements for each section.

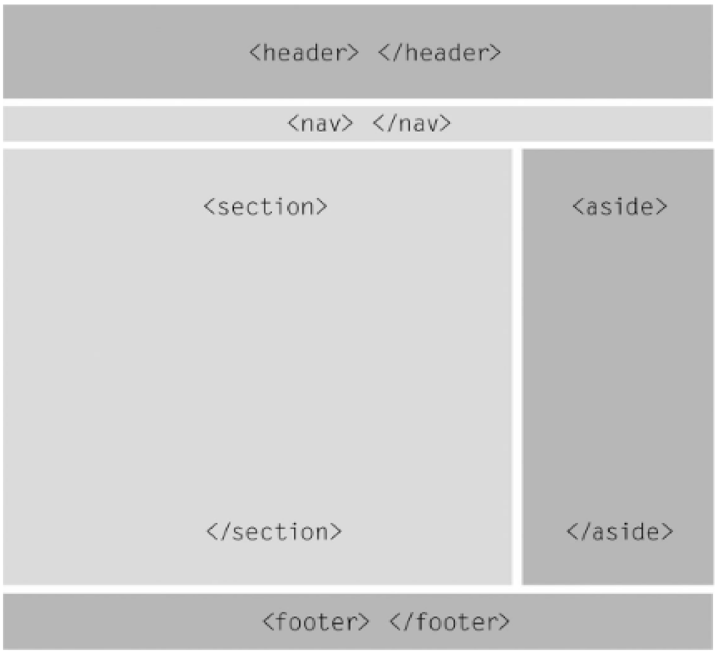


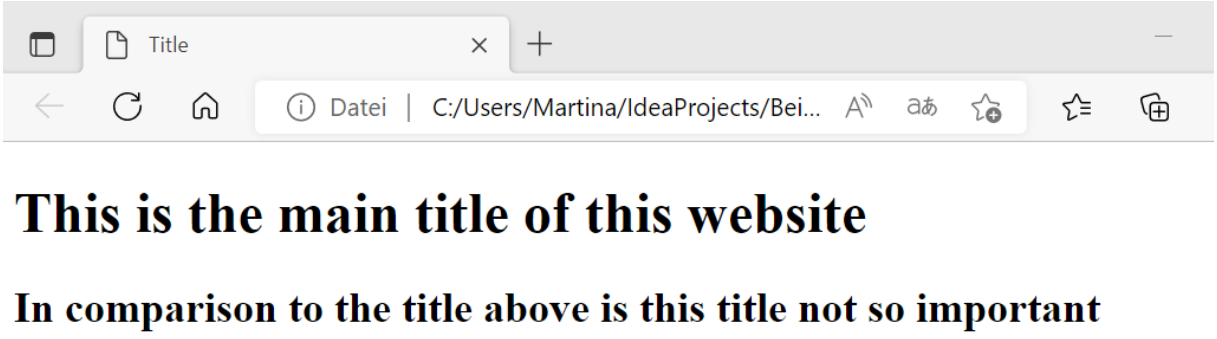
Figure 2: Presentation of the order of each part with HTML5-tags

The first new element of HTML5 is the <header> which represents the introducing information like navigational aids, title and logos. It is important to understand the difference to <head> that delivered information about the document as a whole. In contrast, the <header>-element is only meant for the body or its individual sections (Abel, 2016, p. 30 fff). Listing 5 presents a sample code for the <header>-element.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>
<body>
<header>
  <h1> This is the main title of this website</h1>
  <h2> In comparison to the title above is this title not so
       important</h2>
</header>
</body>
</html>
```

Listing 5: Example of the <header>-element

In this example, the difference between the <head> and the <header>-element is well recognisable. In Listing 6, the reader can see that the title of the document is the one which is standing between the <head>-tags. In contrast to this, the <header>-element shows the beginning of the body and the part of the visible document. That means everything after this tag will be shown in the browser window. Furthermore, <h1>-tags are placed in the example which are an old element of HTML and define the headline. The number behind the "h" shows the importance of the headline and has different formatting (Abel, 2016, p. 33ff).



This is the main title of this website

In comparison to the title above is this title not so important

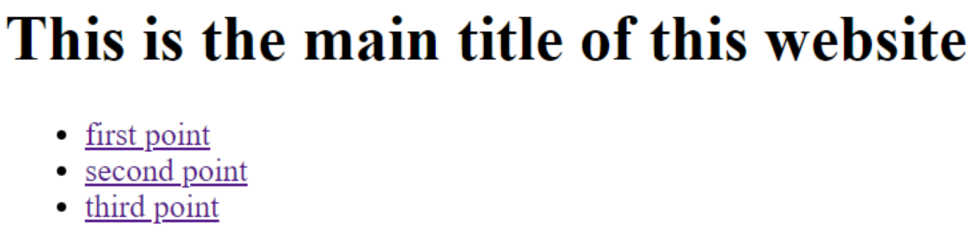
Listing 6: Result of the Code from Listing 5

The next element is the <nav> which presents a part of the document that links to different parts or pages. It is not necessary that all groups of links are placed in this <nav> element, such as terms of service, copyright page or home pages. These types of links will be found in the <footer>-element that shows a short list of links. More information about the <footer>-element can be found below (Pilgrim, HTML5: up and running, 2011, p. 41). Listing 7 shows that the <nav>-element is between the <body>-tags but behind the closing tag of the header.

```
1     <!DOCTYPE html>
2     <html lang="en">
3     <head>
4         <meta charset="UTF-8">
5         <title>Title</title>
6     </head>
7     <body>
8     <header>
9         <h1> This is the main title of this website</h1>
10    </header>
11    <nav>
12       <ul>
13          <li> <a href="https://de.wikipedia.org/wiki/First
                    _Point">first point</a></li>
14          <li> <a href="https://www.powerthesaurus.org/
                    second_point/synonyms">second point</a></li>
15          <li> <a href="https://www.handelszeitung.ch/stichworte
                    /t/third-point">third point</a></li>
16       </ul>
17    </nav>
18    </body>
19    </html>
```

Listing 7: Example for <nav>-element

In this example the <nav>-element will be used to create a menu option. Between the two <nav>-tags there are two elements which are necessary for the creation of a list. The first one is the <ul>-element in row 12 that defined a list. The next element is the <li>-tag in row 13 which is placed between the <ul>-elements and contains different texts. This represents the menu option. The element <a> which is also called anchor element, includes the href attribute and defines a hyperlink to switch to a different page. This href attribute specifies the destination of the selected link (Anghel, 2019). Listing 8 presents the result of the <nav> example.



This is the main title of this website

- first point
- second point
- third point

Listing 8: Result of <nav>-element

The <section>-element represents the section of information in a document. As in Figure 2 presented, it forms the part with the main information. It could be understood as a grouping of content with a heading like chapters, numbered sections of a thesis. In this case it could be a website which divide into different sections for instance introduction, history, products and contact information (Pilgrim, HTML5: up and running, 2011, p. 41). It should be noted that it exists three elements for award of the content area namely <main>, <article> and <section>. The main difference between these three is that the <main>-element is used for the main content of the page. In contrast to the <main>-element, the <article>-element is used for a self-contained article including heading (Selfhtml, 2022). It is placed between the <section>-elements and can be seen as the children of the section such as the <body>-tag has its children, which are the elements between its tags. This element can be used for newspaper article, user comment, forum post or blog post. The special part about this element is that it is an independent part of the document and because of this it has its own structure. Its advantage is that it enables to use the elements <header> and <footer> to structure its content (Abel, 2016, p. 41 ff). The last one of these three is the <section> one, as already mentioned above. This one is used for the sections as a chapter (Selfhtml, 2022). Listing 9 shows an example how a code with the <section>-element could look like.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Title</title>
</head>
<body>
 <header>
  <h1> This is the main title of this website</h1>
     <h2> This title is not so important like the one above</h2>
 </header>
<section>
    <h1> HTML </h1>
    <h2> CSS</h2>
</section>
<section>
    <h1> WEB </h1>
    <h2> MAIL </h2>
</section>
</body>
</html>
```

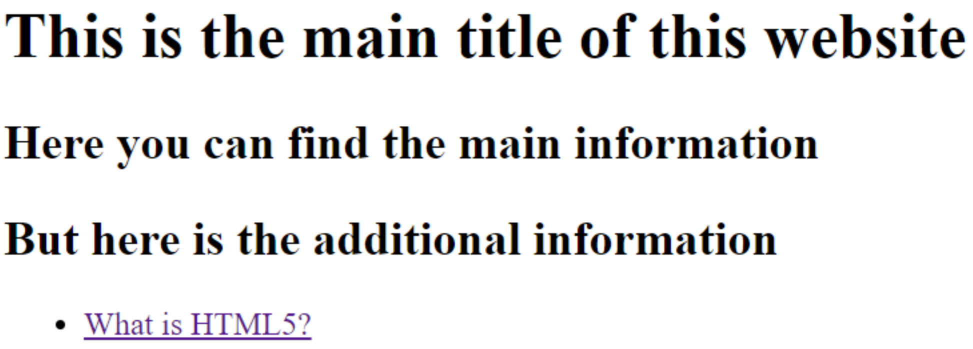
Listing 9: section element example

In combination to the <section>-element, the <aside>-element is also often mentioned. In Figure 2, it is placed next to the <section> and has a smaller part of this area. Normally it contains data which belong to the main information but are not that essential. It is possible to link some links in this area or offer information about the author. The <aside>-element can be placed for example right or left next to the section or below it, It doesn't have a predefined place. Also, a possibility is that it has a place in the section or within the information such as a quote in a text (Abel, 2016, p. 38 f). Listing 10 presents an example which gives a better understanding how it could look like.

```
1     <!DOCTYPE html>
2     <html lang="en">
3     <head>
4       <meta charset="UTF-8">
5       <title>Title</title>
6     </head>
7     <body>
8     <header>
9       <h1> This is the main title of this website</h1>
10    </header>
11    <section>
12      <h1>Here you can find the main information</h1>
13    </section>
14    <aside>
15       <h2> But here is the additional information </h2>
16        <ul>
17          <li> <a href="https://de.wikipedia.org/wiki/HTML5">What is HTML5?
                 </a></li>
18        </ul>
19    </aside>
20    </body>
21    </html>
```

Listing 10: example for <aside>

The code shows in row 11 the <section>-element and in row 14 the <aside>-tag, so it will give a good overview how it could look. In the <aside>-element, a link to a different website to give more information was embedded. Listing 11 depicts the result of this code and the reader will see that in this case the <aside>-area will be below the <section> one.



This is the main title of this website

Here you can find the main information

But here is the additional information

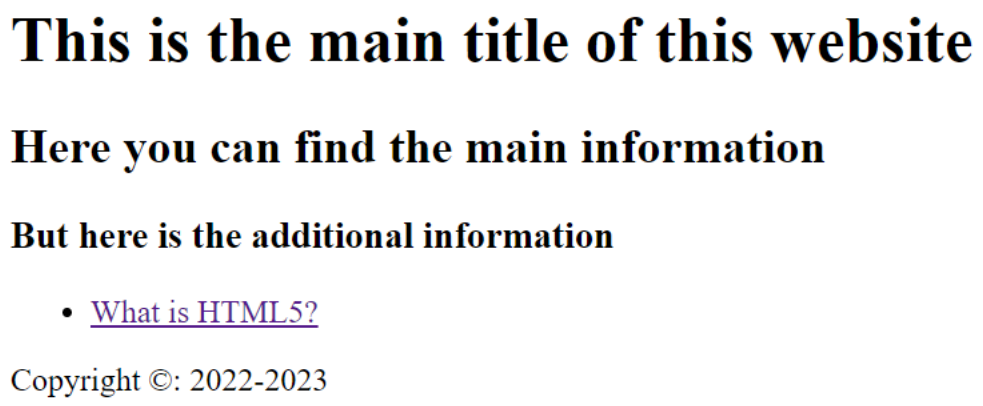
- What is HTML5?

Listing 11: result of the code

The last element of Figure 2 is the <footer>-element that represents the closing part of a website. It typically includes information about the author, links to related documents and copyright data (Pilgrim, HTML5: up and running, 2011, p. 41). It also exists the possibility that the <footer> can be used multiple times in the body part in order to mark the end of every section (Abel, 2016, p. 40). Listing 12 shows an example of a code and Listing 13 the result of it.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>
<body>
<header>
  <h1> This is the main title of this website</h1>
</header>
<section>
  <h1>Here you can find the main information</h1>
</section>
<aside>
  <h3> But here is the additional information </h3>
  <ul>
    <li> <a href="https://de.wikipedia.org/wiki/HTML5">What is HTML5?</a></li>
  </ul>
</aside>
<footer> Copyright &copy: 2022-2023 </footer>
</body>
</html>
```

Listing 12: <footer> example



This is the main title of this website

Here you can find the main information

But here is the additional information

- What is HTML5?

Copyright ©: 2022-2023

Listing 13: result of <footer> code

3.3 Text Elements

HTML is developed to simplify and specify the creation of codes and this happens not only for the structure elements but also for the text elements. The first new one is the <mark>-tag that should highlight a text part because of an interaction of the user. To sum up, this tag will be used to refer to text content, highlight search terms in search results and to mark highlights in quotations made by the page author. Furthermore, the browser shows the <mark>-element in different colours (Selfhtml, 2022). This is the main difference to other text highlighting tags like <strong> or <em>. The <stong>- tag will be used only for the importance and <em>-tag should be used for the emphasis (Abel, 2016, p. 49 f). Listing 14 gives an example how it should look like.

```
<p> I understand the <mark> HTML </mark> language </p>
```

Listing 14: Using <mark>-element

A different meaning gets the <small>-tag which is not used anymore to show the words in small letters but has the task to create the text in small print such as in the copyright-texts. In comparison to this one, the element <cite> has become more specific and encloses the title of papers like books, movies or songs. The next one is an older one which has become changed into a structure element. It enables to implement an URL with this <address>-tag. Normally this element will be positioned in the <footer> (Abel, 2016, p. 50). Listing 15 shows an example.

```
<header>
  <h1> This is an example</h1>
</header>
<footer>
  <address> <a href="http://www.sybex.de">Sybex</a></address>
</footer>
```

Listing 15: Example for <address>-element

The last element which will be presented, is the <time>-element. Its task is to declare a machine-readable timestamp and a human-readable date and time. Important is to remember that this tag will give an exact date and time information. This means that the date is following the Gregorian calendar. Furthermore, there are different kinds to present the date. Table 1 shows a brief overview of the possibilities (Selfhtml, 2022).

Table 1:different formats of data and time

format	description
yyyy	Only a year with 4 figures
yyyy-mm	The year with the month, it is separated through a hyphen. The month has to be double-digit
yyyy-mm-dd	Year, month and day; also the day has to be double-digit
hh:mm	Hour and minutes
hh:mm:ss	Hour, minutes and seconds
hh:mm:ss:fff	Hour, minutes, seconds and fractions of a second

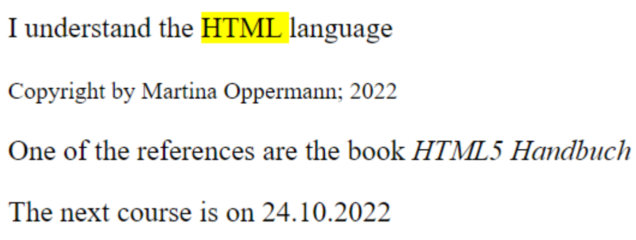
To conclude this chapter the reader will find an example with the almost text elements. Listing 16 shows the code of the document and Listing 8 the result in the web browser.

```
<!-- this is an example -->

<p> I understand the <mark> HTML </mark> language </p>
<small>Copyright by Martina Oppermann; 2022 </small>
<p> One of the references are the book <cite> HTML5 Handbuch </cite></p>
<p> The next course is on <time datatime = "2022-10-24 10:00">
    24.10.2022 </time> </p>
```

Listing 16: the code

In the code, the reader can see the usage of the <mark>-element, which highlights the word “language”. The next one is the <small>-tag that creates the text as a copyright text followed by the <cite>-element which encloses the name of a book. Last but not least, the <time>-element finds its practice. In Listing 17 it presents the output of the code above and shows the difference between the codes with its special features.



I understand the HTML language

Copyright by Martina Oppermann; 2022

One of the references are the book *HTML5 Handbuch*

The next course is on 24.10.2022

Listing 17: Result of the code

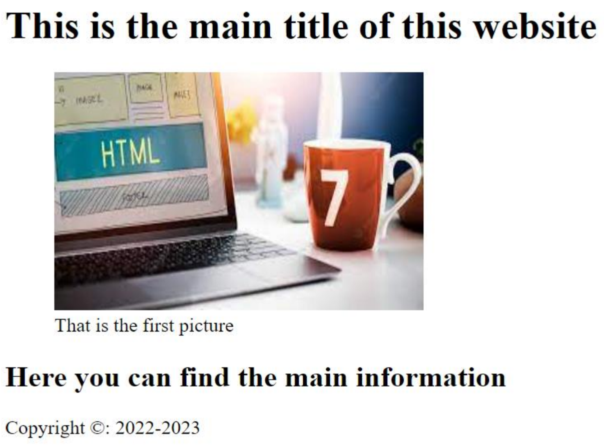
3.4 Embedded Elements

HTML5 had also a development in the area of embedded elements. It is not new that the programmer can add pictures to the code but the disadvantage of the earlier version was that the integration and control was hardly possible. That is the reason why HTML5 created a new element namely <figure> (Abel, 2016, p. 46 f). This element serves as a semantic parent element for graphics like pictures, diagrams and other media content such as videos and quotes. The <figure>-element offered the possibility to become nested in order to present more illustrations. In combination to this element, it is necessary to mention also the <figcaption>-tag that enables to create a labelling of the content. In a <figure>-element it is only possible to have one <figcaption>-element (Selfhtml, 2022). Listing 18 presents an example for an embedded picture.

```
1     <!DOCTYPE html>
2     <html lang="en">
3     <head>
4       <meta charset="UTF-8">
5       <title>Title</title>
6     </head>
7     <body>
8     <header>
9       <h1> This is the main title of this website</h1>
10      <figure>
11       <img src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcR4Ekw8v9LT
                   ATdtHY36RA7dhuGsRkRVyKDZE4dbBPgPyPE1YD4vR09Bv-ufLX5Ysub
12                _o-U&usqp=CAU"
13            alt= "Pic"
14            width="300" height ="200">
15        <figcaption> That is the first picture</figcaption>
16      </figure>
17    </header>
18    <section>
19      <h2>Here you can find the main information</h2>
20    </section>
21    <footer> Copyright &copy: 2022-2023 </footer>
22    </body>
23    </html>
```

Listing 18: the <figure>-element

In row 10 the <figure>-element is placed and is followed by the <img>-tag which includes the “src”, “alt”, “width” and “height”. The “src” indicates the source of the picture in this case the Url of the picture. The usage of a picture needs also the alt-attribute that serves as a substitute when the graphic cannot be loaded (Selfhtml, 2022). “Width” and “height” attributes informs about the size of the picture and allows to tailor it to the specific wishes of the programmer. In row 15 the <figcaption> which can be include a link too is located. Listing 19 depicts the result of the code above.



This is the main title of this website

That is the first picture

Here you can find the main information

Copyright ©: 2022-2023

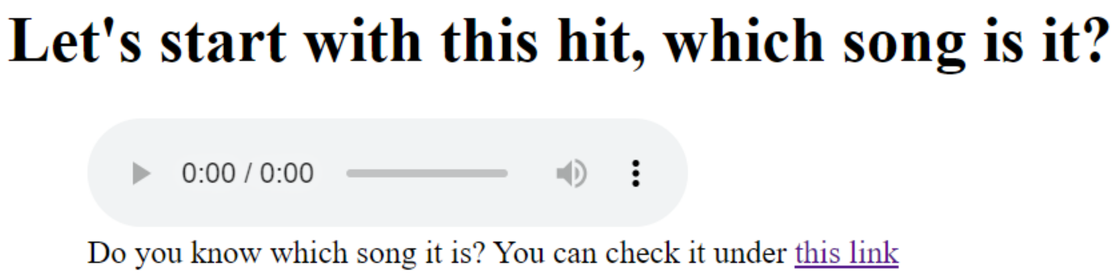
Listing 19: result of the <figure> code

The second new element is the <audio> and the <video> element that allows to play music or movies directly in the browser. In the previous version this feature was only possible when the programmer used Plug-ins such as the Flash-Player. First of all, the <audio>-element gets a closer look. It allows to embed music content or tones without any Plug-ins (Selfhtml, 2021). Listing 20 presents the code of an example with the <audio>-element.

```
1     <!DOCTYPE html>
2     <html lang="en">
3     <head>
4         <meta charset="UTF-8">
5         <title>It's music time</title>
6     </head>
7     <body>
8      <h1> Let's start with this hit, which song is it?</h1>
9      <figure>
10     <audio controls src="https://www.shockwave-sound.com/sound-
11     effects/christmas-sounds/jinglebells1.wav"
12            type="audio/wav"> Your browser can't load this file </audio>
13         <figcaption>
14             Do you know which song it is?
15             You can check it under
16                 <a href="https://www.youtube.com/watch?v=3CWJNqyub3o">this link </a>
17         </figcaption>
18     </figure>
19    </body>
20    </html>
```

Listing 20: example of the <audio>-element

Row 10 shows the <audio>-element which informs the browser about the audio file. Next to the <audio>-tag is the attribute “controls” that has the task to present the player of the audio with some basic functions like play, stop and pause. The “type” attribute informs the browser about the audio format but it is not necessary (Cayzland, n.a.). The <video>-element is used the same way like the <audio>-element. Listing 21 presents the result of this code.



Let's start with this hit, which song is it?

Do you know which song it is? You can check it under this link

Listing 21: result of the <audio>-code

3.5 APIs

Application Programming Interface or in short form API is used to implement codes which are already written to get some new features. All in all, it is a set of pre-built programs. With HTML5 also arose some useful APIs. Here is a brief overview about the most popular one. The first one would be the HTML Geolocation API that has the task to locate the current site of the users. Only with the allowance of the user it will show the location. The next one is the HTML Drag and Drop API that enables to drag an item from one place to another one and drop it there. HTML Web Storage API, on the other hand, has the function to store the data on the web browser. In comparison to the cookies, this API can store a larger data and transfers it to the server. Furthermore, the usage is more secure with the API than using cookies. The next API is the HTML Web Worker API that is used to help to upload the JavaScript with the difference that it enables no effect of the performance of the page. In summary, the JavaScript which runs in the background is not depending on other scripts (geeksforgeeks, 2022). Web Speech API is separated into two parts, on the one hand, the text to speech (TTS) and, on the other hand, the speech recognition. This API allows to interact with the browser through voice commands like speech search, speech navigation or dictating texts. It is restricted to the browsers “chrome” and “firefox”. A disadvantage is that it needs only an one-time permission and no authorisation for this tool (Ivanov, 2020). The last API that is mentioned in this paper is the Canvas API. The objective of it represents the display of bitmap-graphics such as diagrams or games. It is representing the two-dimensional area and is created by scripts.

4 Practical Examples of HTML5

In this chapter the reader is provided with two examples of HTML5 codes and their results. It enables to get an overview how examples could look like and what is possible when the programmer uses the different attributes from above together. For each example there is also an adequate description to get a better understanding for the world of HTML5.

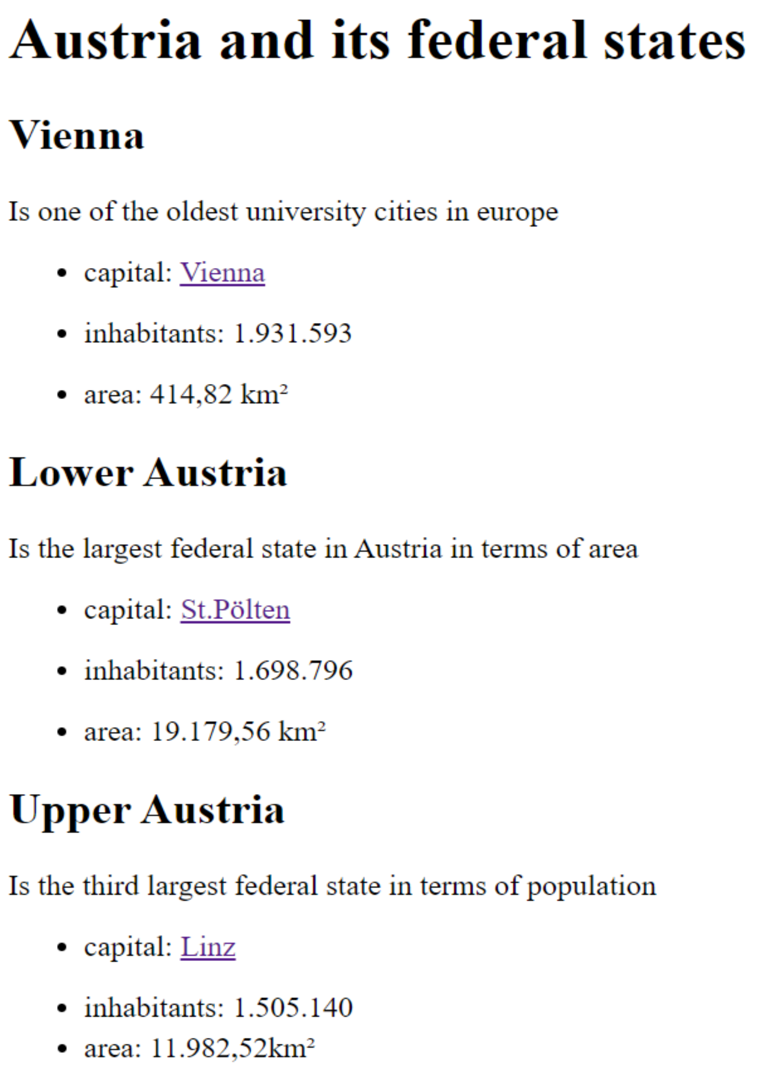
4.1 Nutshell Example – Simple

This example is about the federal state of Austria with some information about it. The code consists of the basic elements that were described above. Listing 22 presents a shortcut of the entire code.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Austria and its federal states</title>
</head>
<body>
 <header>
     <h1>Austria and its federal states</h1>
 </header>
  <article>
   <header>
    <h2>Vienna</h2>
      <p> Is one of the oldest university cities in europe </p>
        <ul>
            <li> <p>capital: <a href="https://de.wikipedia.org/wiki/Wien"
                             >Vienna</a></p></li>
            <li> <p>inhabitants: 1.931.593</p></li>
            <li> <p>area: 414,82 km²</p> </li>
        </ul>
   </header>
  </article>
 <article>
     <header>
         <h2>Lower Austria</h2>
         <p> Is the largest federal state in Austria in terms of area </p>
         <ul>
             <li> <p>capital: <a href="https://de.wikipedia.org/wiki/
                                 St._P%C3%B6lten" >St.Pölten</a></p></li>
             <li> <p>inhabitants: 1.698.796</p></li>
             <li> <p>area: 19.179,56 km²</p></li>
         </ul>
     </header>
 </article>
```

Listing 22: part of the nutshell example

In the first rows, basic information like the “doctype”, the language and the title of the program is placed. It is necessary so that the browser knows what it should create and which language to use. The body of the code begins with row 7. The <header>-element is used to present the heading of the website. After that the <article>-tag is placed for the purpose of presenting a self-contained article including heading. In this element the <h2>-tag to present a subheading, the <p>-tag in order to have a normal text and the <ul>-element are located. Furthermore, the code also shows the <a>-tag that enables to create a link for the selected word and the <li>-element for each subitem to create points before the words. This schema is used for each individual federal state and this is the reason why only a section of the code has been transacted. The whole code will be attached. Listing 23 depicts the result of the section of the code.



Austria and its federal states

Vienna

Is one of the oldest university cities in europe

- capital: Vienna
- inhabitants: 1.931.593
- area: 414,82 km²

Lower Austria

Is the largest federal state in Austria in terms of area

- capital: St.Pölten
- inhabitants: 1.698.796
- area: 19.179,56 km²

Upper Austria

Is the third largest federal state in terms of population

- capital: Linz
- inhabitants: 1.505.140
- area: 11.982,52km²

Listing 23: result of the nutshell example

4.2 Nutshell Example – Complex

This example presents an online registration for excursions. The idea of this website is to get a clearly arranged page where the user is able to fill out the form in order to take part of an excursion. In this program, a lot of different elements from already known to new ones like <style>-element, <form>-element and its attributes are used. The example is organised in four parts named head and body with the separation into section and aside. Listing 24 shows the first part of the code.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Registration for Excursions</title>
    <style>
   header, footer, nav, article, aside {display:block;}

   aside {
           float: right;
           width: 30%;
           height: 90%;

           margin-left: 5%;
                   }

    body{
         background-image: url("https://encrypted-tbn0.gstatic.com
                           /images?q=tbn:ANd9GcTwJJt_j1OAKZgpHrOz-
                           4BR_Sd_VD_36FfgNg&usqp=CAU");
         background-size: cover;
    }

   #mainContent {
           margin-left: 30%;
           padding: 8pt;
   }

   h1, h2, h3, h4, h5, h6 {
           font-family: Arial,sans-serif;
   }

   body footer {
           text-align: center;
           color: grey;
   }
   </style>
</head>
```

Listing 24: first part of the code of the example

The first part includes the <doctype>-element, the language and the head part. The special feature on the <head> is that it consists not only of <meta>- and <title>-attribute but also of the <style>-attribute. This attribute enables to style the website by making entries regarding the size, font and position of various parts of the structure. Most of the codes in Listing 24 are self-explanatory by the designation of the attribute. The next part of the code shows Listing 25.

```
37   <body>
38    <header>
39        <h1> Registration for Excursion </h1>
40        <p> Please fulfil the form below</p>
41    </header>
42    <aside>
43        <nav>
44           <h2>Excursions: </h2>
45            <ul>
46                <p> <b> <a href="https://www.getyourguide.at/wien-
                                   l7/stadtrundgang-durch-die-unterwelt-
47                      wiens-t543/"> Underground tour</a> </b> </p>
48                <p> <b> <a href="https://www.getyourguide.at/wien-
                                   l7/wien-spaziergang-zwischen-tod-und-
49                 geisterstunde-t41067/?"> Ghost tour</a> </b> </p>
50                <p> <b> <a href="https://www.getyourguide.at/wien-
                                   l7/wien-zentralfriedhof-stadt-der-toten
51                    -t104195/?">Central Cemetery tour</a></b> </p>
52            </ul>
53            <h3> Advertisement</h3>
54            <figure>
55                <img src="https://encrypted-tbn0.gstatic.com/images?q=tb
                          n:ANd9GcR32qtnC_U5az-0PTocNGHinmkQtCbsE4Iu2Uco
56                        LA8I_tFvyVriKNJT4CkvmME7OlDjn20&usqp=CAU"
57                     alt= "Pic"
58                     width="300" height ="200">
59                <figcaption> Only now - The best tourist bus for 20€</figcaption>
60            </figure>
61        </nav>
62    </aside>
```

Listing 25: second part of the code

The next step in the code is the <body> that consists of the <header>, <aside>, <section> and the <footer>. In row 42 the start-tag of the <aside>-element which will be located on the right side of the website is placed. It contains embedded links to different websites which give a better overview about each excursion. The attributes of this code were already described in the chapter above. But the <b>-attribute is new and is used to create the text in bold. The next element is the image that is also already known from the chapter above. An interesting detail is that it is possible to put a part of the URL in the next row.

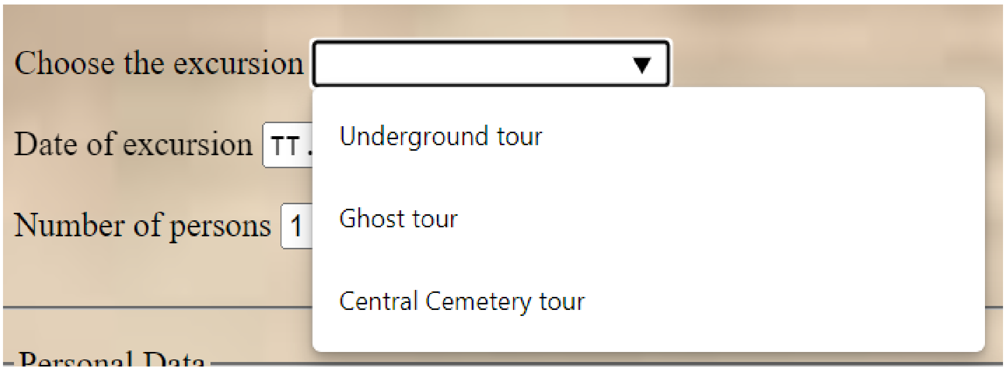
This possibility is really useful when the Url is too long for the programmer and the link still works when it is between the inverted commas. Listing 26 presents the following part with the topic <section>.

```
<section id="form">
    <form>
        <h1> Online Reservation for excursion</h1>
        <fieldset>
            <legend> Excursion</legend>
            <p>
             <label for="excursion"> Choose the excursion </label>
                 <input type="text" list="excursions"
                        id="excursion" name="excurison">
                 <datalist id="excursions">
                    <option label="Underground tour"
                            value="Underground tour">
                    <option label="Ghost tour" value="Ghost tour">
                    <option label="Central Cemetery tour"
                            value="Central Cemetery tour">
                </datalist>
            </p>
            <p>
             <label for="date"> Date of excursion </label>
                 <input type="date" id="date" name="date" required>
            </p>
            <p>
             <label for="person"> Number of persons </label>
                <input id="person" type="number" min="1" max="6"
                       step="1" value="1">
            </p>
        </fieldset>
        <p> </p>
        <fieldset>
            <legend>Personal Data</legend>
            <p>
                <label for="fullname">full name: <input
                        type="text" id="fullname" name="fullname" size="50"
                        placeholder="firstname, surname"
                        pattern="([a-zA-Z]+ *)+, ([a-zA-Z]+ *)+"
                        autofocus required></label>
            </p>
            <p>
                <label for="phone">phone: <input type="tel"
                       id="phone" name="phone"></label>
            </p>
            <p>
                <label for="email">e-mail: <input type="email"
                       id="email" name="email" required></label>
            </p>
        </fieldset>
        <p>
            <input type="submit" value="send">
        </p>
    </form>
</section>
```

Listing 26: third part of the code - <section>

Listing 26 starts with the <section>-element that includes the ID of it. The next step is to create the form of the website. To implement this, the <form>-element is required. Row 66 shows the attribute <fieldset> that enables a grouping below a common heading. This element presents a kind of a block and with the <legend>-element it is possible to create a description of this block (Selfhtml, 2022).

The <label>-element, on the other hand, is responsible for the description of the elements of the form and with “for” it determines the ID of the input-element (Selfhtml, 2022). The <input>-element allows to enter of the labels and with the addition “datalist” it is possible to create a list where the user can choose between some options such as in row 72. Listing 27 presents this feature.



Choose the excursion

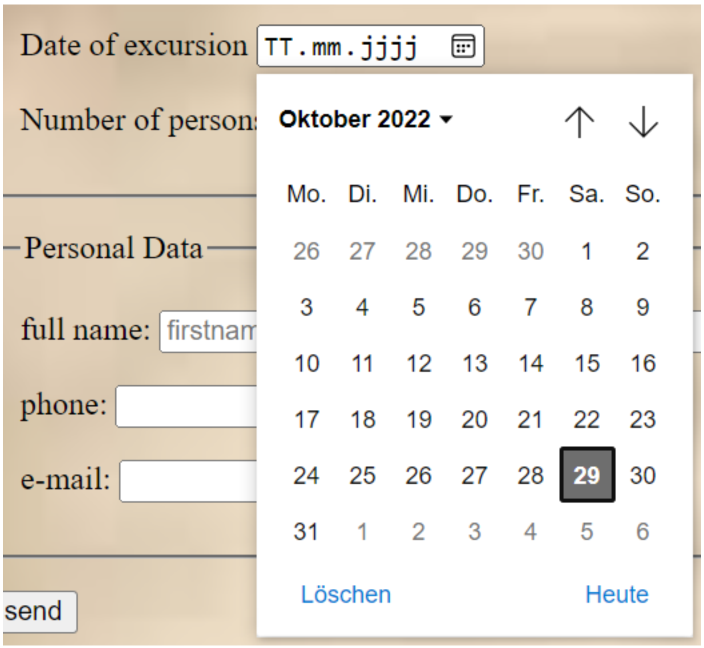
Date of excursion TT.

Number of persons 1

Personal Data

Listing 27: result of the datalist

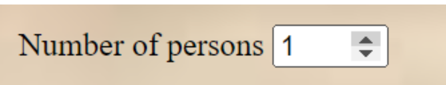
The next new attribute is the creation of a data field where the reader can select the preferred date. For the implementation the programmer has to type “date” in the field “type=”. In row 79 the word “required” which has the effect that the user has to choose a date is also placed. Listing 28 gives an example of this attribute.



Mo.	Di.	Mi.	Do.	Fr.	Sa.	So.
26	27	28	29	30	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
31	1	2	3	4	5	6

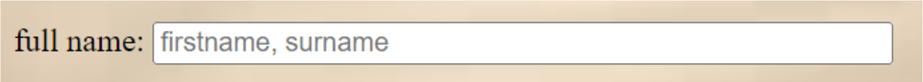
Listing 28: result of the "date" entry

Row 82 shows the creation of a field where the user can change the number of persons. To accomplish this, the use of number for the area “type=” is necessary. With the attributes “min” and “max” the creator can specify a range of the possible numbers and attribute “step” determines in which levels the values can be entered. The last attribute “value” allows pre-setting an input field with a content. Listing 29 depicts this feature.


Number of persons 1

Listing 29: result of the number element

The <fieldset> in row 87 starts a new box with the name “Personal Data”. In these three fields should be created where the user can enter its name, telephone number and e-mail address. All these fields work the same. So, here is a deeper insight into the first one – the name of the user. In row 90 the “type=” is placed with the entry “text” that allows the user to write something into this label. The attribute “size” enables to specify this field by limiting the numbers of letters. Furthermore, the attribute “placeholder” is used to make a specific placeholder for the input field. In comparison to this, the attribute “pattern” specifies the format of the input. Listing 30 presents the result of this special label.



full name: firstname, surname

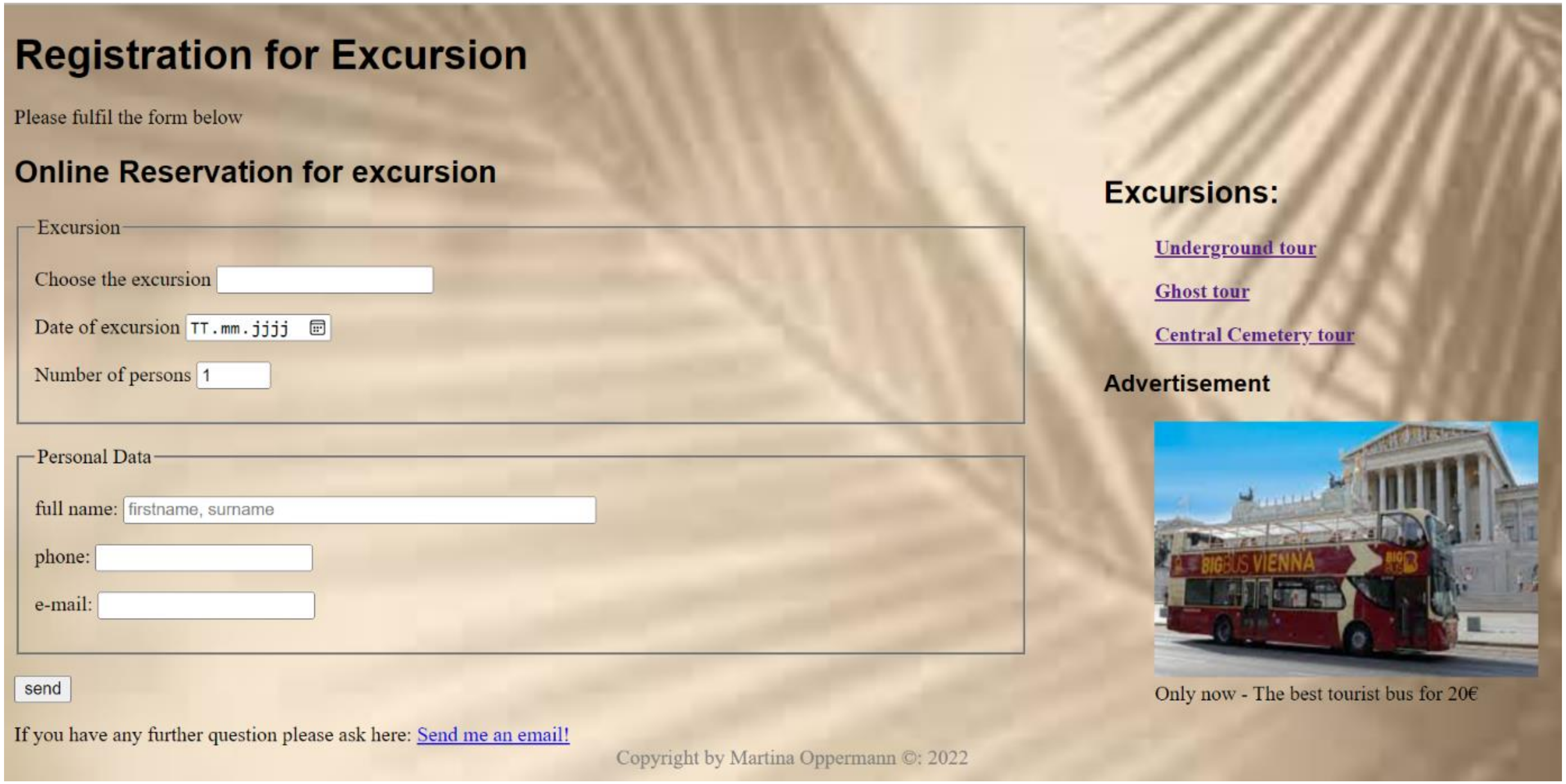
Listing 30: result of the code for full name

After closing the <fieldset>-element, a code for the button “send” is placed. It is again created with the attribute <input> again and has the word “submit” for the field “type=”. This enables that the form is sent and immediately afterwards the page is reloaded. With the entry of “send” in the attribute “value=” it is possible to name the button. Listing 31 shows the last part of the code.

```
107  <section>
108      <h5> If you have any further question please ask here:</h5>
109      <a href=mailto:octaviaanghel@gmail.com?subject=Important
          >Send me an email!</a>
110  </section>
111  <footer> Copyright by Martina Oppermann &copy: 2022</footer>
112  </body>
113  </html>
```

Listing 31: last part of the code

Started with the <section>-attribute and followed with the creation of a link to the mail of the user, this code is placed in row 107. The interesting part of this code is the possibility to get a direct access to the mail of the user. The “mailto” type of link enables this special feature and through the addition of “subject=Important” it is possible to set the subject of the mail marked as important. The last part of the code is the footer with the “copyright”-sentence which was already described in the chapter about the footer element. Listing 32 presents the whole result of the code.



Registration for Excursion

Please fulfil the form below

Online Reservation for excursion

Excursion

Choose the excursion

Date of excursion TT.mm.jjjj

Number of persons 1

Personal Data

full name: firstname, surname

phone:

e-mail:

send

If you have any further question please ask here: Send me an email!

Copyright by Martina Oppermann ©: 2022

Excursions:

Underground tour

Ghost tour

Central Cemetery tour

Advertisement

Only now - The best tourist bus for 20€

Listing 32: the result of the whole code

5 Conclusion and Outlook

The basic idea of Berners-Lee was to create a global uniform communication form which can be used by all people independent of their skills. And Berners-Lee was able to implement this idea. HTML5 allows an easy way to implement content also in terms of user-friendliness and customer loyalty. The new interfaces and functions of HTML5 enable to respond in a special way to the needs of the user and the search engine (Wesolowski & Scheel, 2013). This paper enables the reader to get an overview about the powerful programming features and how to use the new HTML5 APIs. It ranges from the different basic elements like <header>, <footer> and continue with the new features for instance <form>, <figure> and <audio>. With the help of the corresponding examples, it is not only possible for the reader to gain a better understanding of the application, but also to be able to implement it independently. Furthermore, it is incredible how many things are realizable just with the APIs such as showing the position on a map. There is also a progress regarding the browser support. More and more browsers can read and work with the HTML5 language that has the result that the development in this field is gaining speed (Lubbers, Albers, & Salim, 2010, p. 313). For the future it will be interesting how the joint development process by WHATWG and W3C will continue because of the different tensions between them. Also, the extent to which further development of HTML5 will take place and how quickly the adaptations will be made can only be guessed at. The future will soon show the answers to these questions.

6 References

Abel, C. (2016, 2 19). *silo.tips.* Retrieved 10 27, 2022, from https://silo.tips/download/kapitel-1-html5-dokumente

Anghel, O. (2019, 11 11). *HTMLGoodies*. Retrieved 10 25, 2022, from https://www.htmlgoodies.com/html5/html5-navigation-using-an-anchor-tag-for-hypertext/

Cayzland. (n.a.). *Cayzland*. Retrieved 11 10, 2022, from https://www.cayzland.de/info/mit-html5-musik-und-audio-auf-webseiten-einbinden

Everything, Position is. (n.a.). *Position is Everything*. Retrieved 11 10, 2022, from https://www.positioniseverything.net/html4-vs-html5/

Flatscher, R. (2022, 06 30). Windows-Automatisation 2 - Markup Languages, Document Object Model. Wien, Wien, Österreich. Retrieved 11 15, 2022

geeksforgeeks. (2022, 8 31). *geeksforgeeks*. Retrieved 11 10, 2022, from https://www.geeksforgeeks.org/explain-apis-available-in-html5/

Handelsblatt. (2008, Mai 02). *Computerwoche*. Retrieved 11 05, 2022, from https://www.computerwoche.de/a/damals-das-zeitalter-der-browserkriege,1862843

Hoy, M. (2011, 01 24). *Taylor Francis Online.* Retrieved 10 27, 2022, from https://www.tandfonline.com/doi/full/10.1080/02763869.2011.540212

Ivanov, I. (2020, 10 15). *codecoda*. Retrieved 10 26, 2022, from https://codecoda.com/de/blog/entry/a-guide-to-understanding-html-apis

Lubbers, P., Albers, B., & Salim, F. (2010). *Pro HTML5 Programming.* apress. Retrieved 10 30, 2022

Münz, S., & Gull, C. (2011). *HTML5 Handbuch.* Franzis. Retrieved 10 18, 2022

Pilgrim, M. (2011). *Durschstarten mit HTML5.* O'Reilly. Retrieved 10 17, 2022

Pilgrim, M. (2011). *HTML5: up and running.* O'Reilly Media. Retrieved 10 20, 2022

Robie, J. (n.a.). *W3*. Retrieved 10 22, 2022, from https://www.w3.org/TR/REC-DOM-Level-1/introduction.html

Selfhtml. (2021, 2 18). *wiki.selfhtml*. Retrieved 11 05, 2022, from https://wiki.selfhtml.org/wiki/HTML/Elemente/audio

selfhtml. (2022, 8 30). Retrieved 11 05, 2022, from https://wiki.selfhtml.org/wiki/HTML/Tutorials/Entstehung_und_Entwicklung

selfhtml. (2022, 03 28). *wiki.selfhtml*. Retrieved 10 29, 2022, from https://wiki.selfhtml.org/wiki/HTML/Tutorials/Element,_Tag_und_Attribut

Selfhtml. (2022, 8 18). *wiki.selfhtml*. Retrieved 11 05, 2022, from https://wiki.selfhtml.org/wiki/HTML/Tutorials/Textstrukturierung

Selfhtml. (2022, 7 23). *wiki.selfhtml*. Retrieved 11 03, 2022, from https://wiki.selfhtml.org/wiki/HTML/Tutorials/Seitenstrukturierung

Selfhtml. (2022, 8 18). *wiki.selfhtml*. Retrieved 11 03, 2022, from https://wiki.selfhtml.org/wiki/HTML/Tutorials/Textstrukturierung#figure

Selfhtml. (2022, 7 8). *wiki.selfhtml*. Retrieved 10 30, 2022, from https://wiki.selfhtml.org/wiki/HTML/Elemente/label

Selfhtml. (2022, 7 13). *wiki.selfhtml*. Retrieved 10 30, 2022, from https://wiki.selfhtml.org/wiki/HTML/Tutorials/Textauszeichnung#mark

Selfhtml. (2022, 10 6). *wiki.selfhtml*. Retrieved 10 28, 2022, from https://wiki.selfhtml.org/wiki/HTML/Elemente/time

Selfhtml. (2022, 1 29). *wiki.selfhtml*. Retrieved 10 28, 2022, from https://wiki.selfhtml.org/wiki/HTML/Elemente/fieldset

Selfhtml. (2022, 4 10). *Wiki.selfhtml*. Retrieved 10 29, 2022, from https://wiki.selfhtml.org/wiki/HTML/Tutorials/Grundger%C3%BCst von Kesteren, A., & Pieters, S. (2011, 01 13). *w3*. Retrieved 11 01, 2022, from https://www.w3.org/TR/2011/WD-html5-diff-20110113/

Wesolowski, R., & Scheel, S. (2013, März 11). *eStrategy*. Retrieved 11 03, 2022, from https://www.estrategy-magazin.de/html5-moeglichkeiten-fuer-ihre-website.html

Appendix

This appendix shows the full code from the nutshell example simple. Further information about the code is on page 22 in chapter 4.1.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Austria and its federal states</title>
</head>
<body>
 <header>
     <h1>Austria and its federal states</h1>
 </header>
  <article>
   <header>
    <h2>Vienna</h2>
      <p> Is one of the oldest university cities in europe </p>
        <ul>
            <li> <p>capital: <a href="https://de.wikipedia.org/wiki/Wien"
                              >Vienna</a></p></li>
            <li> <p>inhabitants: 1.931.593</p></li>
            <li> <p>area: 414,82 km²</p> </li>
        </ul>
   </header>
  </article>
 <article>
     <header>
         <h2>Lower Austria</h2>
         <p> Is the largest federal state in Austria in terms of area </p>
         <ul>
             <li> <p>capital: <a href="https://de.wikipedia.org/wiki/
                               St._P%C3%B6lten" >St.Pölten</a></p></li>
             <li> <p>inhabitants: 1.698.796</p></li>
             <li> <p>area: 19.179,56 km²</p></li>
         </ul>
     </header>
 </article>
 <article>
     <header>
         <h2>Upper Austria</h2>
         <p> Is the third largest federal state in terms of
             population</p>
         <ul>
             <li><p>capital: <a href="https://de.wikipedia.org/wiki/
                                Linz" >Linz</a></p></li>
             <li> <p>inhabitants: 1.505.140 </p></li>
             <li> <p>area: 11.982,52km²</p></li>
         </ul>
     </header>
 </article><article>
     <header>
         <h2>Salzburg</h2>
         <p> There are large salt deposits in the region</p>
         <ul>
        <ul>
            <li><p>capital: <a href="https://de.wikipedia.org/wiki/Salzburg" >Salzburg</a></p></li>
            <li> <p>inhabitants: 560.710</p></li>
            <li> <p>area: 7.154,56 km²</p></li>
        </ul>
    </header>
</article><article>
    <header>
        <h2>Styria</h2>
        <p> Is known for its forest and meadows</p>
        <ul>
            <li><p>capital: <a href=https://de.wikipedia.org/wiki/Graz >Graz</a></p></li>
            <li> <p>inhabitants: 1.252.922</p></li>
            <li> <p>area: 16.399,34 km²</p></li>
        </ul>
    </header>
</article><article>
    <header>
        <h2>Tyrol</h2>
        <p> Federal state which is most dominated by high mountains</p>
        <ul>
            <li><p>capital: <a href="https://de.wikipedia.org/wiki/Innsbruck" >Innsbruck</a></p></li>
            <li> <p>inhabitants: 764.102</p></li>
            <li> <p>area: 12.648,37 km²</p></li>
        </ul>
    </header>
</article><article>
    <header>
        <h2>Vorarlberg</h2>
        <p> is alsco called "Ländle"</p>
        <ul>
            <li><p>capital: <a href="https://de.wikipedia.org/wiki/Bregenz" >Bregenz</a></p></li>
            <li> <p>inhabitants: 401.674</p> </li>
            <li> <p>area: 2.601,67 km²</p></li>
        </ul>
    </header>
</article><article>
    <header>
        <h2>Carinthia</h2>
        <p> It has the highest mountain named Großglockner</p>
        <ul>
            <li><p>capital: <a href="https://de.wikipedia.org/wiki/Klagenfurt" >Klagenfurt</a></p></li>
            <li> <p>inhabitants: 564.513</p></li>
            <li> <p>area: 9.536,50 km²</p> </li>
        </ul>
    </header>
</article><article>
    <header>
        <h2>Burgenland</h2>
        <p>In this area the people use the UI-dialect</p>
```

```
         <p>In this area the people use the UI-dialect</p>
         <ul>
             <li> <p>capital: <a href="https://de.wikipedia.org/wiki/Eisenstadt"  >Eisenstadt</a></p></li>
             <li> <p>inhabitants: 297.583</p> </li>
             <li> <p>area: 3.965,20 km²</p> </li>
         </ul>
     </header>
 </article>
 <footer> Copyright &copy: 2022-2023 </footer>
</body>
</html>
```