



Formulare - Übung

- Legen Sie ein neues HTML-Dokument an, mit einem `<form>`-Tag und drei `<input>`-Tags als Listen-Elemente an
- Vorname, Nachname, E-Mail sollen als Text übertragen werden (`action="register.php" method="get"`)
- Mit `<button>absenden</button>` fügen Sie noch einen Button vor das schließende `<form>`-Tag ein.
- Ergänzen Sie das Formular um ein Passwort-Eingabefeld!

Requests

- GET als URL-Parameter in der Browserleiste
- method="post" ändert den Request-Typ
- Auf dem Server ist die Umstellung u.U. schwieriger!
- GET-Requests um Seiten zu holen (URL-Parameter sind nützlich für Lesezeichen!)
- POST-Request um Daten an den Server zu senden, die dann ggf. gespeichert werden sollen

Weitere Typen

- Checkbox / Radio-Buttons:

- `<input type="radio" name="geschlecht" value="w" id="geschlecht_w">`

- `<input type="radio" name="geschlecht" value="m" id="geschlecht_m">`

- `<input type="checkbox">`

- Select-Boxen:

- `<select name="studium">`

- `<option value="info">Informatik</option>`

- `<option value="contr">Controlling</option>`

- ...

- `</select>`

Formulare 2.0 – HTML5

Typ	IE 11	Edge	Firefox	Chrome	Safari	Bemerkung
type="email"	Ja	Ja	Ja	Ja	Ja	Lässt nur gültige E-Mails zu
type="date"	Nein	Ja	Ja	Ja	Nein	Eingabe eines Datums.
type="number"	Ja	Ja	Ja	Ja	Ja	Lässt nur Zahlen zu
type="search"	Ja	Ja	Ja	Ja	Ja	Wird vom Browser wie ein Suchfeld dargestellt
Required	Ja	Ja	Ja	Ja	Ja	Markiert ein Feld als Pflichtfeld

Block- und Inline-Elemente

- Block-Elemente sind HTML-Elemente, die nicht innerhalb des Textes vorkommen, aber Text umfassen können.
 - Blockelemente sind immer rechteckig
 - z.Bsp.: <p>, <table>
-
- Inline-Elemente sind HTML-Elemente, die mitten im Text vorkommen können.
 - z.Bsp.: <a>,



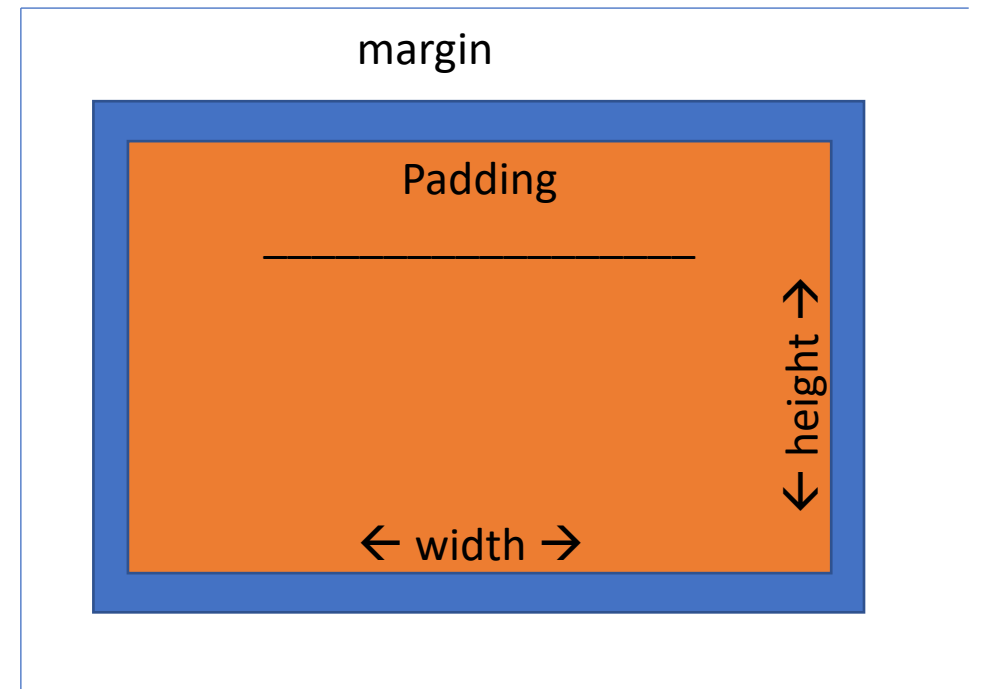
BOX-Model

- Das `<div>`-Tag
- Ist ein generischer Container und kann alle Elemente enthalten
- `<div>` funktioniert in allen Browsern.

Fügen Sie einen `<div>`-Container in Ihren HTML-Code ein und beschreiben Sie das Layout-Verhalten.

BOX-Model 2

- width und height beziehen sich auf den Platz, der dem Inhalt einer Box zur Verfügung steht. Sie enthalten weder margin noch padding.
- Padding: Abstand zw. dem Inhalt und dem Rahmen
- Margin: Mindestabstand zw. dieser Box und der nächsten



Übung <div>

- Platzieren Sie auf einer neuen Seite fünf <div>s mit Text darin und den Ids eins, zwei, drei, vier, fuenf. Geben Sie verschiedene Hintergrundfarben an.
- Div 1 ohne Größenangaben
- Div 2 eine Höhe in Pixeln
- Div 3 Breite und Höhe in Pixeln
- Div 4 Breite und Höhe in Pixeln
- Div 5 Breite und Höhe in em

Diskussion zur Übung

- Zwischen den <div>s gibt es keinen Abstand
- Wenn der Text zu lang ist, geht er einfach über den Rand
- Bei div Nr. 4 wird die Box so hoch, bis der Text passt. Höhenangaben in % funktionieren nur zuverlässig, wenn für das umgebende Element eine Größe gesetzt ist.
- Ist keine Größe angegeben, wird das Element gerade so groß, dass sein Inhalt komplett hineinpasst.

Abstände

- padding: 10px; – in allen Himmelsrichtungen 10px Abstand
- padding: 10px 20px; Nord + Süd: 10px, West + Ost: 20px;
- padding: 10px 20px 5px: Nord: 10px; West+Ost: 20px; Süd: 5px;
- padding: 10px 20px 5px 15px: ...?



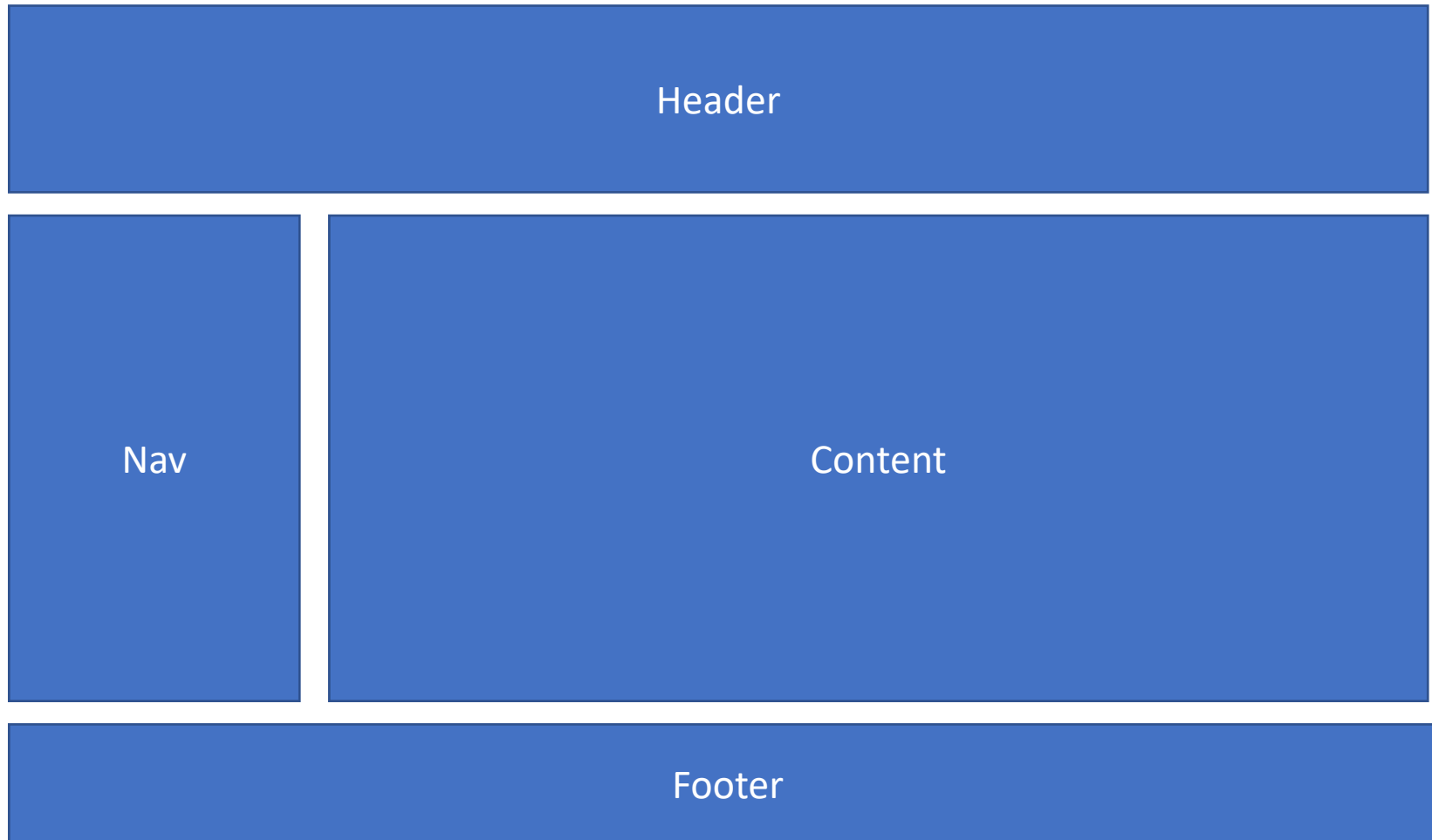
Das Fenster im Fenster

- `<iframe>` lädt eine HTML-Seite, deren URL im `src`-Attribut angegeben wird
- `<iframe>`s sind komplett unabhängig

```
<iframe src="http://google.de">
```

Fügen Sie einen `iframe` ein und färben Sie allen Text grün!

Komplette Webseite



HTML (grob)

```
<body>
  <header><div id="header">
    <h1>Lego, Lego, Lego</h1>
  </div></header>

  <nav><div id="nav">
    <ul>
      <li class="ausgewaehlt"><a href="#">Startseite</a></li>
      <li><a href="newscenter.html">News</a></li>
      <li><a href="galerie.html">Lego-Galerie</a>
        <ul>
          <li><a href="galerie2021.html">2021</a></li>
          <li><a href="galerie2020.html">2020</a></li>
          <li><a href="galerie2019.html">2019</a></li>
        </ul>
      </li>
      <li><a href="impressum.html">Impressum</a></li>
    </ul>
  </div></nav>
  <div id="inhalt"></div>
  <footer><div id="footer">Sohn von Florian Thürkow</div></footer>
</body>
```

CSS (grob)

```
header, footer, nav {  
    margin: 0;  
    padding: 0;  
    border: none;  
}
```

```
#header {  
    position: absolute;  
    left: 10px;  
    top: 10px;  
    right: 10px;  
    height: 100px;  
    background: transparent url("header_image.png") no-repeat;  
    text-align: right;  
}
```

```
#nav {  
    position: absolute;  
    left: 10px;  
    top: 110px;  
    width: 200px;  
    bottom: 30px;  
    background-color: #30804B;  
}
```

```
#inhalt {  
    position: absolute;  
    left: 210px;  
    right: 10px;  
    top: 110px;  
    bottom: 30px;  
    padding: 10px;  
    overflow: auto;  
}
```

CSS - Nav

```
#nav a, #nav a:visited, #nav a:hover, #nav a:focus {  
    display: block;  
    height: 2em;  
    text-decoration: none;  
    color: black;  
    text-align: right;  
    font-size: 1.6em;  
    padding: 2px 10px 0 0;  
    font-variant: small-caps;  
    font-weight: bold;  
}
```

Galerie CSS

```
#inhalt.galerie div{  
    border: 1px solid white;  
    width: 19%;  
    min-width: 200px;  
    height: 30em;  
    margin: 1em 0 0 0.5%;  
    float: left;  
    text-align: center;  
    padding: 0;  
    color: white;  
}
```

min-width: Dieses Element soll nie schmaler als 200 Pixel werden

height in em, damit sich die Höhe am Text orientiert und NICHT am umgebenden Element

float left ordnet die Bilder nebeneinander an

Lightbox (JavaScript)

<head>

```
<title>Lego Bauer</title>  
<link rel="stylesheet" href="lego.css"/>  
<link href="css/lightbox.css" rel="stylesheet" />  
<script src="js/jquery-1.7.2.min.js"></script>  
<script src="js/lightbox.js"></script>
```

Lightbox aktivieren:

```
<a href="galerie/lego1.jpg" rel="lightbox"></a>
```

JavaScript

- Vollwertige Programmiersprache
- Komplexe Programme (Bildbearbeitung, Office, Spiele, ...)
- Seit dem Netscape Navigator 2.0, 1995
- JavaScript != Java
- Dynamisch typisierte, prototypenbasierte, interpretierte Programmiersprache
- Seit 2009 node.js -> JavaScript auf der Serverseite



JavaScript – erste Übung

```
...  
<body>  
    <script>  
        alert("BA-Leipzig!");  
    </script>  
    <noscript>  
        JavaScript ist nicht aktiv!  
    </noscript>  
</body>  
...
```

alert() Funktionsaufruf

Funktionen können mit Parameter übergeben werden, z.B.:
"BA-Leipzig".

Jede JavaScript-Anweisung wird mit einem Semikolon
abgeschlossen.

Achtung: Anführungszeichen sind wichtig!

JavaScript – Datei einbinden

- `<script src="meinjavascript.js"></script>`
- Für HTML 4:
- `<script type="text/javascript" src=".."></script>`



Rechnen mit JavaScript

- Versuch:
- $7+3=<\text{script}>\text{alert}(7+3);</\text{script}>$. Rechnen mit JavaScript.
- Ersetzen Sie die alert Funktion durch document.write()

Variablen

```
var meineVariable;  
meineVariable = 3*3;  
var andereVariable = 5;
```

- Variablen sind benannte Speicherstellen
- Zwei Dinge:
 - Wert in eine Variable speichern
 - Wert wieder abrufen
- Dürfen mit Buchstaben, \$ oder _ beginnen
- Sollten immer mit einem Kleinbuchstaben beginnen
- Variablenname immer links vom Gleichheitszeichen

Variablen 2 + Kommentare

```
<script>  
    //Wir rechnen mit JavaScript  
    var regalBoeden = 5;  
    var steineJeRegalBoden = 140;  
    var steineImZimmerVerteilt = 300;  
  
    var steineGesamt = steineImZimmerVerteilt + regalBoeden * steineJeRegalBoden  
    /* Ausgabe im Browser.  
       Hoffentlich klappt es */  
    document.write(steineGesamt);  
</script>
```