

Schulung **Web-Entwicklung**

Tag 2:
Vertiefung HTML, Vertiefung CSS

Michael Brüggemann
Moritz Hipper



Übersicht

01 **HTML Vertiefung**

Formulare, Divs

02 **CSS Vertiefung**

Einbinden, Selektoren,
Pseudoelemente und -
klassen

HTML Vertiefung

01



Semantik

- „**Bedeutungslehre**“
- Der Tag-Typ bzw. der **Elementtyp**, gibt dem Inhalt eine **spezielle Bedeutung**

`<h1>Ich habe hunger </h1>` \leftrightarrow `<p>Ich habe hunger</p>`

- Gibt es für einen Anwendungsfall ein Entsprechendes HTML-Element, sollte dies **immer dem Div bevorzugt** werden

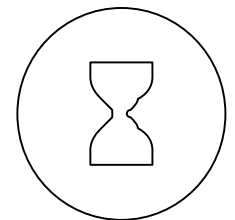


Aufgabe 3

Euer Chef will einen Pizzabestellservice über das Internet an die Kunden bringen. Dieser soll eine Bestellfunktion haben.

Legt vorerst nur eine Bestellseite an, auf der als Überschrift der Name eures Pizzaservices und euer Motto zu sehen ist.

! Legt die Datei `bestellung.html` an und beginnt.



10 Minuten

Formulare

`<form>` `</form>`

- Ermöglichen dem Nutzer die Dateneingabe
- Besitzt das besondere Attribut *action=zielresource*
 - ist dieses Attribut gesetzt, werden beim Klick auf das Submit-Element die Formulardaten an die angegebene Zielresource übergeben
 - Ist es nicht gesetzt, sollten die Daten per Script verarbeitet werden
- Beinhalten **diverse Input und Options – Elemente**
- **Anzahl** der Eingabeelemente ist **variabel**
- **Immer** ein **Submit-Element** vorhanden
- Das Element **<label>** kann genutzt werden, um **die Inputs zu beschreiben**



Aufgabe 4

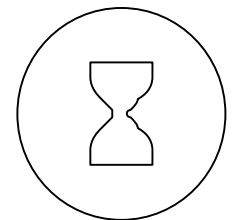
Damit der Kunde seine Wünsche äußern kann, fehlt der Bestellseite ein Formular.

Über dieses soll er folgende Aspekte Konfigurieren können:

- Variable Pizzagröße
- Beläge: Pilze, Ananas, Schinken, Brokkoli
- Nachricht auf dem Karton

Weil euer Chef bisher aber nur eine Website hat und keinen Pizzaofen, übergebt ihr die Eingabeparameter an den externen Dienstleister *localhost*.

! Erweitert die Datei `bestellung.html`.



30 Minuten



HTML – Objekt div

`<div> </div>`

- „**Division**“
- Wahrscheinlich **meist genutzt** Element
- Hat im Gegensatz zu anderen Elementen **keine semantische Bedeutung**
→ Diese **gibt ihm der Entwickler** durch CSS-Klassen oder JavaScript
- Beispielhafter Nutzen: **Cursor-Tracking** in bestimmtem Bereich
- Legen eines **Hintergrundes** für die Elemente **innerhalb des DIVs** mithilfe von CSS
- Spezielle Positionierung der Inhalte

CSS

Vertiefung

02



Wie man es einbindet

- **Inline**

→ wird in dem HTML-Attribut
STYLE platziert

```
<div style="background-color: #00557F; border-width: 5px;">  
  Content  
</div>
```

- **Intern**

→ wird in dem STYLE – Element
innerhalb des HEAD – Elements
platziert

```
<head>  
  <style type="text/css">  
    div {  
      background-color: #00557F;  
      border-width: 5px;  
    }  
  </style>  
</head>
```

- **Extern**

→ wird in externe Datei platziert, welche im HEAD
in das HTML importiert wird

- Als **Mischlösung**

→ CASCADING Style Sheet

```
<link rel="stylesheet" type="text/css" href="styles.css">
```

Selektoren

- Legen Fest, auf **welche HTML-Elemente** der **Style** angewandt wird
- **Universell**

```
* {  
    //styles  
}
```
- **Attribut: ID**

```
<div id=„idname“>  
#idname { ...
```
- **Attribut**

```
input[type=„password“] { ...
```
- **Element - Typ**

```
input { ...
```
- **Klasse** (kombinierbar)

```
<div class=„klassenname klassennamezwei“> ...  
.klassenname { ...
```



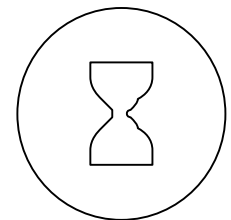
Aufgabe 5

Legt drei DIVs untereinander mit der Höhe und Breite 200px an.

Der Hintergrund des ersten ist Grün, der des zweiten Blau und der des dritten Schwarz.

Die Styles ersten DIVs ist inline deklariert, die des zweiten im dedizierten STYLE-Tag und die des dritten per css-Klasse in einer externen Datei.

Legt die Dateien uebersicht.html und styles.css an und beginnt.



30 Minuten

Aufgabe 6

Euer Chef meint, einfach so bunte DIVs Anzulegen sei unproduktiv. Außerdem sei die Hintergrundfarbe völlig unsinnig gesetzt und die Anzahl viel zu gering.

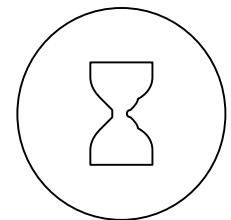
Lagert die Komplette Styledefinition in eine externe Datei aus.

Verdoppelt die Anzahl der DIVs und macht alle DIVs grau und gleich groß (200px x 200px).

Die Divs sollen je nach Platz in mehreren Reihen sein (flex-Layout).

Erweitert die Dateien uebersicht.html und styles.css.

Flex-Layout: <https://css-tricks.com/snippets/css/a-guide-to-flexbox/>



20 Minuten

Aufgabe 7

Euer Chef kann immer noch nichts mit leeren DIVs anfangen.

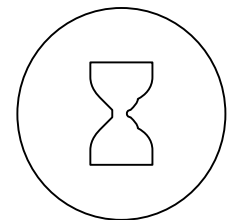
Er will, dass ihr die DIVs nutzt, um euren Kunden Pizzamodelle als Vorschau zu Präsentieren.

Deshalb soll jedes DIV nun folgenden Inhalt haben:

- Den Namen einer Pizza, darunter...
- ...eine Bildvorschau einer Pizza

Erweitert die Dateien uebersicht.html und styles.css.

Legt den Ordner Assets in eurem Aufgabenordner an, speichert in diesem Bilder von Pizza und nutzt diese in eurem HTML.



20 Minuten



Pseudoklassen und -selektoren

- Pseudoklassen **referenzieren** einen bestimmten **Element** – **Zustand** oder **Position**

→ Zustand:

```
div:hover { ...
```

```
a:active { ...
```

→ Position

```
.klassenname:first-child { ...
```

```
.klassenname:nth-child(2) { ...
```

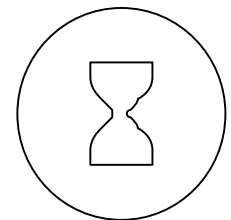
Aufgabe 8

Euer Chef hat gelesen, dass man den Nutzer unterschwellig gut mit suggestiven Farben zum Kauf überreden kann.

Die erste Pizzavorschau soll nun immer einen goldenen und die zweite einen orangenen Hintergrund haben.

Außerdem gefällt ihm nicht, dass alles so gequetscht aussieht. Sorgt auch dafür, dass um jede Vorschau ein Rand von 10px frei ist, also jeweils 20px zwischen jeder Vorschau Abstand herrscht.

Erweitert nur die Datei styles.css, nicht die html Datei.

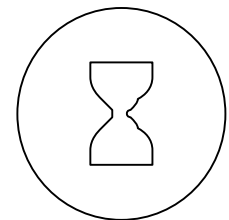


15 Minuten

Aufgabe 9

Irgendwie findet euer Chef, dass die Pizzaübersichtsseite nicht zeitgemäß wirkt. Er will, dass der Mauszeiger zu einem Zeigefinger wird, wenn man mit der Maus über einer Vorschau schwebt. Zusätzlich soll die Vorschau in diesem Fall einen dezenten Schatten werfen.

Erweitert nur die Datei `styles.css`, nicht die `html` Datei.



10 Minuten