

JSON (JavaScript Object Notation)

```
{
  "Vorname": "Lea",
  "Matrikel": 123
  "Noten": [1.3, 1.0, 2],
  "Auto": {
    "Typ": "Audi",
    "PS": "100"
  }
}
```

```
<student>
  <Vorname>Lea</Vorname>
  <Matrikel>123</Matrikel>
</student>
```

Erlaubte Datentypen

- String
- Zahlen
 - > Ganzzahlen
 - > Gleitkommazahlen 1.23
 - > Wiss. Notation: 1.2E3
- Bool: true, false
- Null
- Object
- Array

Gemeinsamkeiten mit XML

- Plain-Text, menschenlesbar
- Hierarchische Struktur

Unterschiede mit XML

- JSON ist "schlanker"
- JSON kennt nur vordefinierte Datentypen, XML erlaubt die Definition eigener
- JSON kennt Arrays
- JSON lässt sich direkt in JavaScript

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="ISO-8859-1">
5   <title>Insert title here</title>
6
7   <script type="text/javascript">
8     function freundeListe() {
9       var JSONObject =
10        {
11          "vorname": "Alex",
12          "matrikel": 123,
13          "freunde" : [
14            {
15              "vorname": "Leoni",
16              "haarfarbe": "braun"
17            },
18            {
19              "vorname": "Steffen",
20              "haarfarbe": "rot"
21            }
22          ]
23        };
24
25        //alert("Der Student heißt " + JSONObject.vorname);
26        //alert("Der erste Freund heißt " + JSONObject.freunde[0].vorname);
27
28        var freunde = JSONObject.freunde;
29        var ergebnis = "";
30        for(freund in freunde){
31          ergebnis += freunde[freund].vorname + " ";
32        }
33        alert("Die Liste der Freunde: " + ergebnis);
34        document.getElementById("f").innerHTML = ergebnis;
35      }
36
```



Werte aus einer JSON-
Datei auslesen

JQuery

JQuery ist eine JavaScript-Bibliothek, die über die `$`-Funktion zahlreiche Möglichkeiten bietet, um

- den DOM-Baum zu manipulieren: `html()`, `append()`, ...
- Effekte hinzuzufügen: `slideUp()`, `hide()`, `fadeIn()`, `animate()`, ...
- Daten asynchron vom Server zu laden (AJAX): `ajax()`, `load()`,

Syntax

`$(SELECTOR).action();` //wobei SELECTOR dem CSS-Selector entspricht (Elementen, Klassen, IDs, ...);

```
$(document).ready(){  
    function() {  
        //hier eigener Code  
    }  
}
```

Beispiel

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="ISO-8859-1">
5   <title>Insert title here</title>
6   <script type="text/javascript" src="./js/jquery-3.3.1.js"></script>
7
8   <style type="text/css">
9     .rot {color:red};
10  </style>
11
12 </head>
13 <body>
14   <h1>Arbeiten mit JQuery</h1>
15
16   <p id="JQ1">Alter Text</p>
17   <p>Das zweite "p"</p>
18   <div id="JQ2" style="background-color:#CFF">Noch ein alter Text</div>
19
20   <script type="text/javascript">
21     //alte Variante ohne JQuery
22     document.getElementById("JQ1").innerHTML = "Neuer Text";
23
24     //neue Variante mit $-Funktion von JQuery
25     $("p").html("Noch eine Text mit JQuery eingefügt");
26
27     //Element mit JQuery einfügen
28     $("#JQ2").append("Neuer angehängter Text");
29     $("#JQ2").append("<p>Neuer angehängter Text</p>");
30
31     //Animation
32     $("#JQ2").slideUp();
33
34     //CSS-Klassen zuweisen
35     $("p").addClass("rot");
36
37
38   </script>
39 </body>
40 </html>
```

Auswahl aller "p"-Tags über css-Selektor

Layout-Änderung durch Zuweisen einer CSS-Klasse

HTML 5

1. Video-Tag, Audio Tag

Kurze Variante

```
<video id="video" controls autoplay
  src="video.mp4" type="video/mp4"
  src="video.ogv" type="video/ogg"
  src="video.webm" type="video/webm">
  Ihr Browser ist zu doof
</video>
```

2. Neue Struktur-Element (Container): article, nav, header, footer

3. Input-Type : email, date, url,

```
<input type="..."
```

4. Place-Holder

5. Typisierte Link (a href="tel:07531206111"), SMS,

6. Canvas

Neue Input-Typen

- text
- text autocorrect
- text autocapitalize off
- tel
- url
- email
- password
- number
- text pattern
- Checkbox
- A
- B
-
-
-
-
-
-
-

Powered by [iWebKit](#)

```
<!-- HTML 5 Input-Typen / Eingabefelder
Markus Spiering / Sven Haiges
-->
<!DOCTYPE html>
<html>
<head>...</head>
<body>
  <div id="topbar">...</div>
  <div id="content">
    <ul class="pageitem">
      <li class="bigfield">...</li>
      <li class="bigfield">...</li>
      <li class="bigfield">...</li>
      <li class="bigfield">...</li>
      <li class="bigfield">...</li>
      <li class="bigfield">...</li>
      <li class="bigfield">
        <input placeholder="email" type="email">
        </li>
      <li class="bigfield">...</li>
      <li class="bigfield">...</li>
      <li class="bigfield">...</li>
      <li class="checkbox">...</li>
      <li class="radiobutton">...</li>
      <li class="radiobutton">...</li>
      <li class="bigfield">
        <input type="range" value="0" min="0" ma
        </li>
      <li class="bigfield">...</li>
      <li class="bigfield">...</li>
      <li class="bigfield">...</li>
      <li class="bigfield">...</li>
      <li class="bigfield">...</li>
      <li class="bigfield">...</li>
      <li class="bigfield">...</li>
    </ul>
  </div>
  <div id="footer">...</div>
</body>
</html>
```

HTTP: Hypertext-Transferprotokoll

- Unverschlüsselt, textbasiert
- Request-Response-Protokoll
- Zustandslos (stateless) (Ausweg: Cookies, ...)

```
Headers
GET /fileadmin/css/johner.css?1498219926 HTTP/1.1
Host: www.johner.org
Accept: text/css,*/*;q=0.1
Accept-Encoding: gzip, deflate
Accept-Language: de-DE,de;q=0.9,en-US;q=0.8,en;q=0.7
Cookie: __utmpk=0
Referer: http://www.johner.org/
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64) Appl

HTTP/1.1 200 OK
Accept-Ranges: bytes
Content-Encoding: gzip
Content-Length: 4488
Content-Type: text/css
Date: Mon, 09 Apr 2018 13:17:52 GMT
ETag: "4b24-5529f827d8413-gzip"
Last-Modified: Fri, 23 Jun 2017 12:12:06 GMT
Server: Apache/2.4.10 (Debian)
Vary: Accept-Encoding
```

Request

Response

	Request	Response
1. Zeile	Methode z.B. GET Ressource z.B. /images/bild1.png Protokoll: HTTP/1.1	Protokoll: z.B. HTTP/1.1 Status-Code: z.B. 200 (ok), 400er (Client Fehler, 500 (Server)
Header	Informationen über Client/Browser <ul style="list-style-type: none">• Encoding• Browser-Typ• Betriebssystem•	Informationen über Server <ul style="list-style-type: none">• Debian• Webserver, PHP-Version
Body		Nutzlast z.B. <ul style="list-style-type: none">• Bilder• CSS• HTML• ...

Der Server gibt dem Client ein Cookie (Text-Datei) mit, die der Client bei weiteren Anfragen an den Server schickt, damit dieser den Client wiedererkennt.

Suchen

Folgende Cookies sind auf Ihrem Computer gespeichert:

Website	Cookie-Name
chrome.google.com	
doubleclick.net	
google.com	
google.de	
google.de	CONSENT
google.de	SNID
google.de	NID
google.de	DV
google.de	1P_JAR

Name: DV
Inhalt: Y3ujN-7iTaoYMD5YAYY_xWvRhALqKxY
Host: www.google.de
Pfad: /
Senden für: Jeden Verbindungstyp
Gültig bis: Freitag, 13. April 2018, 12:59:58

Ausgewählten löschen Alle löschen Schließen

Backend-Programmierung und Dynamische Webseiten

Generieren von HTML-Seiten

- PHP
- Microsoft: .NET-Sprachen ASP.NET,
- Ruby on Rails
- Java: JSP, Servlets,
- Perl
-

