



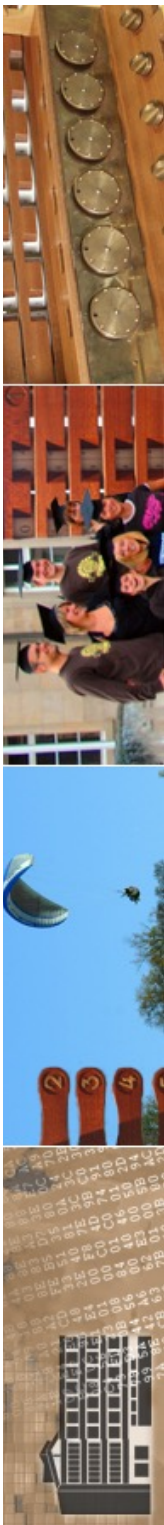
# Einführung Internet-Technologien

## INF3171

Clientseitige Web-Programmierung  
mit JavaScript

Version 1.0

02.05.2024





## Für Einsparungen: Google kündigt komplettem Python-Team, Ersatz wohl in München

Wegen Umstrukturierungen entlässt Google weitere Mitarbeiter. Einem ganzen Python-Team wurde offenbar gekündigt; ersetzt wird das Team durch eines in München.

🇬🇧 📄 🔊 🖨️ 💬 264



(Bild: Alberto Garcia Guillen/Shutterstock.com)

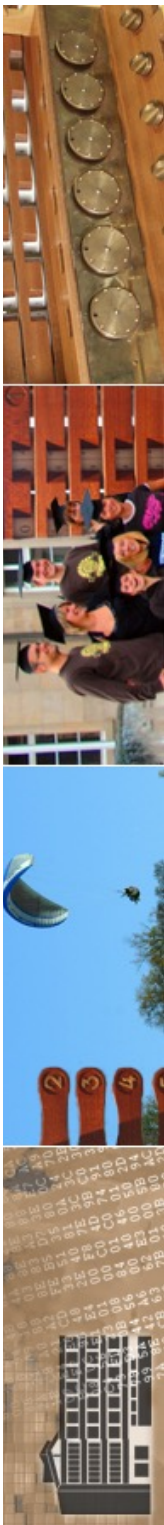
**UPDATE** 30.04.2024, 04:37 Uhr Lesezeit: 2 Min.

Von [Marie-Claire Koch](#)

Google hat in den letzten Wochen weiteren Mitarbeitern gekündigt. Speziell betroffen ist ein Python-Team des Unternehmens, das ersetzt wird, aber auch die Teams von Dart und Flutter, wie unter anderem TechCrunch berichtet. Das sei Teil der geplanten Umstrukturierungen; betroffene Mitarbeiter können sich demnach auf andere offene Stellen bei Google bewerben. Google plane zudem, die Kosten zu senken, indem es weniger teure Arbeitskräfte außerhalb der USA einstellt. Für die Aufgaben des Python-Teams soll es jetzt ein Team mit "billigeren Arbeitskräften" in München geben.

# clientseitige Web-Programmierung

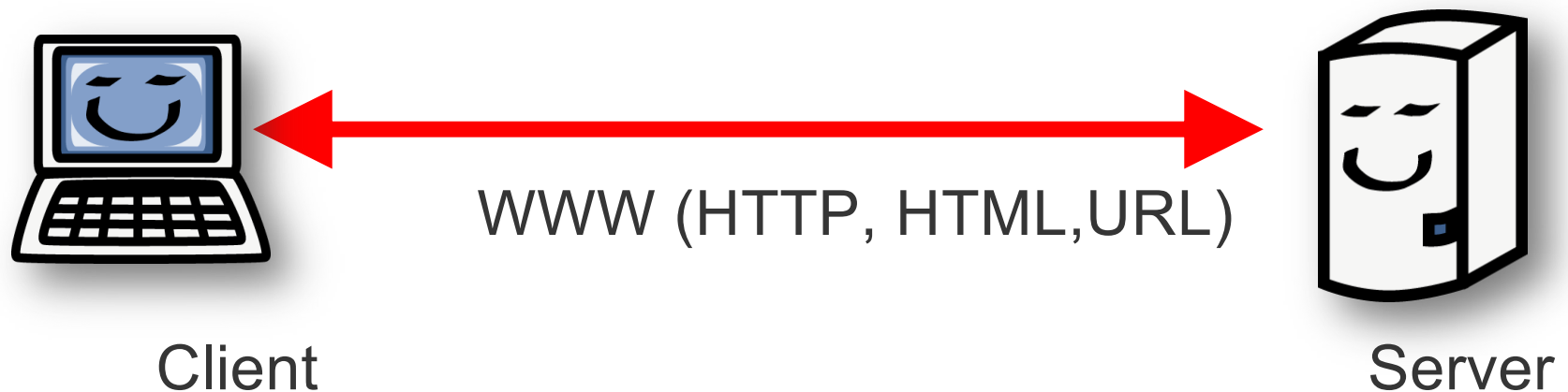
- es bestehen viele verschiedene Ansätze, u.a.
  - JavaScript
  - Java Applet
  - Adobe Flash
  - Microsoft Silverlight
  - Allgemeiner:  
RIA: rich internet application





# Architektur

- Situation:



- "Webprogramme" können auf dem Client oder/und dem Server ausgeführt werden
  - wichtige Mischformen: **AJAX** und **WebSockets**



# clientseitige Web-Programmierung

- Vorteil:
  - kein Netzwerk - schnelle Response
  - aktive Interaktion mit dem User
    - „Maus-Effekte“
  - Performance durch Nutzung der Client-CPU
    - jeder user hat eigene CPU im Gegensatz zur serverseitigen Web-Programmierung
- Nachteile:
  - Security auf dem Client
  - keine Kontrolle der Software auf dem Client
  - de facto keine Datenbank-Anbindung möglich



# Programmiersprachen

- Einschub: Was ist eine Programmiersprache?
  - "Notation für ein Computerprogramm"
    - maschinenauswertbare Notation eines Algorithmus
  - Unterscheidung
    - binäre Maschinsprache
    - Assembler
    - Hochsprachen
      - Algol, Fortran, Cobol, C, Pascal, Java, C#, ...
      - Basic, PHP, Perl, JavaScript, Python, Ruby, ...



Muhammed al-Chwarizmi  
783 - 850



# Programmiersprachen

- Hauptbestandteile von Programmiersprachen
  - **Variablen:** Name für Ort im Speicher
    - die „Subjekte und Objekte“ der Programmiersprache
    - ohne oder mit Typ für den Wert der Variablen
      - Typen: Zeichen (Char), ganze Zahl (Integer), Gleitkommazahl (float/double), Zeichenkette (String), ...
  - **Kontrollstrukturen:** Steuern den Programmfluss
    - Verzweigung (wenn - dann - sonst), Schleifen
  - **Anweisungen**
    - drucke, lese, beende, ...





# Programmiersprachen

- Unterscheidung: Compiler und Interpreter
  - aus dem Code der höheren Programmiersprache muss zum Ausführen Maschinencode werden
- **Compiler:** Übersetzt vor der Ausführung
  - keine Syntaxfehler bei der Ausführung
  - performant
- **Interpreter:** Übersetzen bei der Ausführung
  - flexibler







# Scriptsprachen

- in dieser Veranstaltung benutzen wir drei Scriptsprachen:
  - JavaScript
    - clientseitige Web-Programmierung
  - PHP
    - serverseitige Script-Programmierung
  - Python
    - serverseitige CGI-Programmierung
  - Ruby
    - serverseitige CGI-Programmierung





# JavaScript

- 1995 Brendan Eich: **LiveScript** (Netscape)
  - Netscape Navigator 2.0
- 1996 Umbenennung in **JavaScript**
- Klassiker: Version 1.5
- aktuell Version 1.8.5
  - Juli 2010
  - Sprachkern **ECMAScript**
- Literatur:
  - RRZN: JavaScript
  - Flanagan, David:





# Den Kinderschuhen entwachsen – 25 Jahre JavaScript

Was 1995 als einfache Skriptsprache begann, ist heute die Grundlage praktisch jeder modernen Webanwendung. Zeit für einen Blick zurück, aber auch nach vorn.

Lesezeit: 6 Min.  In Pocket speichern

   181

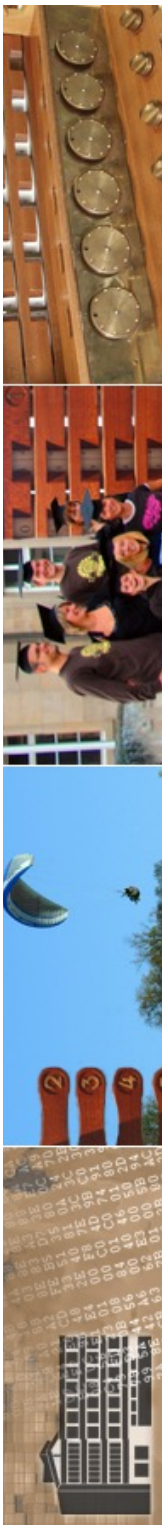


(Bild: ritchielee CC-BY-SA 2.0)

04.12.2020 18:35 Uhr | Developer

Von Golo Roden

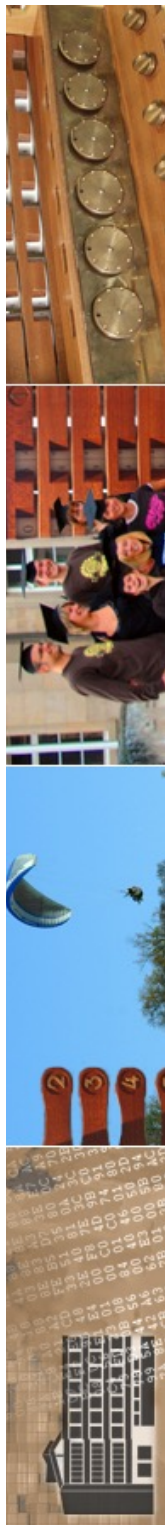
Mit Jubiläen bei Programmiersprachen ist das so eine Sache. Was als Geburtsstunde gilt, kann variieren. Fakt ist, dass die Netscape Communications Corporation in Verbindung mit Sun Microsystems am 4. Dezember 1995, also vor genau 25 Jahren, eine neue Programmiersprache namens JavaScript ankündigte, die als offene und plattformunabhängige Sprache speziell für den Einsatz im Internet gedacht war. Ein wesentliches Ziel war von Anfang an, eine Sprache zu erschaffen, die zwar einfach, zugleich aber auch mächtig sein sollte.



# Selfhtml



The screenshot shows a web browser window with the URL <https://wiki.selfhtml.org/wiki/JavaScript>. The browser's address bar and tabs are visible. The page content includes a navigation menu with links for 'SELFHTML', 'Wiki', 'Forum', and 'Blog'. A search bar is present with the text 'Suchbegriff'. The main heading is 'JavaScript', and the introductory text states: 'Für die Entwicklung dynamischer Webseiten ist JavaScript unerlässlich. Mit ihr lassen sich kleine Hilfsroutinen bis hin zu komplexen Frameworks schreiben.' Below this, there are three columns of links: 'Sprachelemente' (including Kommentare, Bezeichner, Reservierte Wörter, Steuerzeichen, Variablen, Datentypen, Operatoren, Funktionen, Arrays), 'Standardobjekte' (including navigator, screen, window, document, forms, elements, history, location), and 'Schnittstellen' (including DOM, Element, Event, EventTarget, Node, Canvas 2D, Drag & Drop, File Upload, Geolocation, Google Maps-API einbinden, IndexedDB). A sidebar on the left contains a table of contents with sections like 'ÜBERSICHT', 'SCHNELL-INDEX', 'MITMACHEN', 'WERKZEUGE', and 'SPENDEN'. The browser's address bar shows the user's name 'Thomas'.





# Alternativen

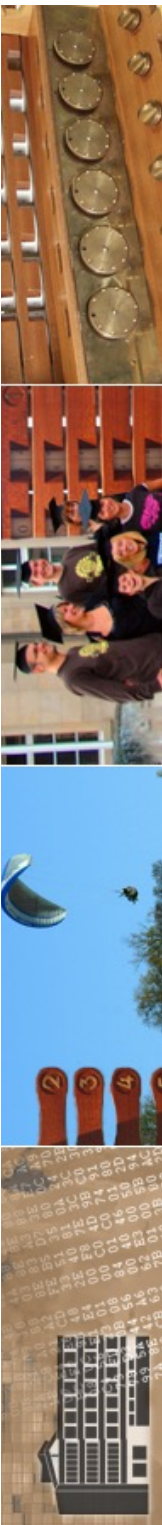
- allgemeiner Ansatz: **ECMA-Script** (s.u.)
  - löst als Standard faktisch JavaScript ab
  - jährlich neue Version
- proprietäre Alternativen (insbesondere von Microsoft)
  - JScript
  - TypeScript
  - ...und einige mehr...





# JavaScript

- zunächst ist JavaScript eine moderne, einfache Programmiersprache
  - nicht-typisiert („dynamisch typisiert“)
  - wird interpretiert
  - objektorientiert, aber *ohne Klassen* ☹️
  
- interessant wird JavaScript, weil diese Sprache *mit Webseiten "verbunden" werden kann*
  - Elementen in einer Web-Site werden JavaScript-Methoden zugeordnet
  - zentral: Verknüpfung von HTML und JavaScript: **DOM**





# Grundprinzip

- mit JavaScript wird aktiver Code geschrieben, *der in einem HTML-Dokument integriert wird*
- Webseite „enthält das Programm“
- JavaScript wird vom jeweiligen Browser lokal ausgeführt





# Editoren für JavaScript

- jeder Editor, der für HTML geeignet ist, ist auch für JavaScript geeignet
  - einfache Editoren wie notepad++ oder vi
  - HTML-Tools wie Dreamweaver
  - Programmierertools:  
Eclipse mit Webtools







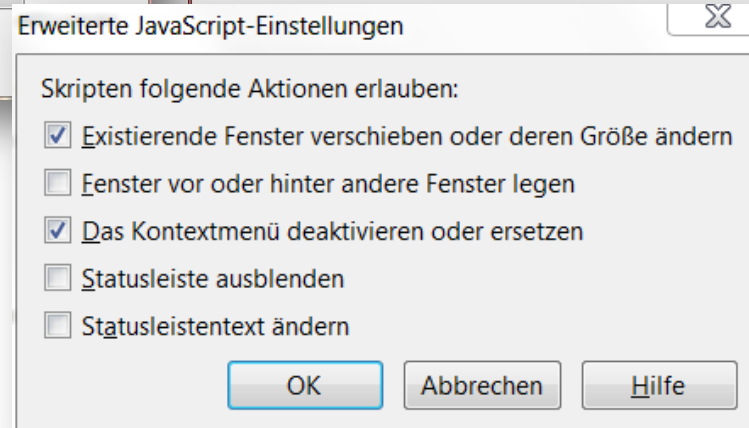
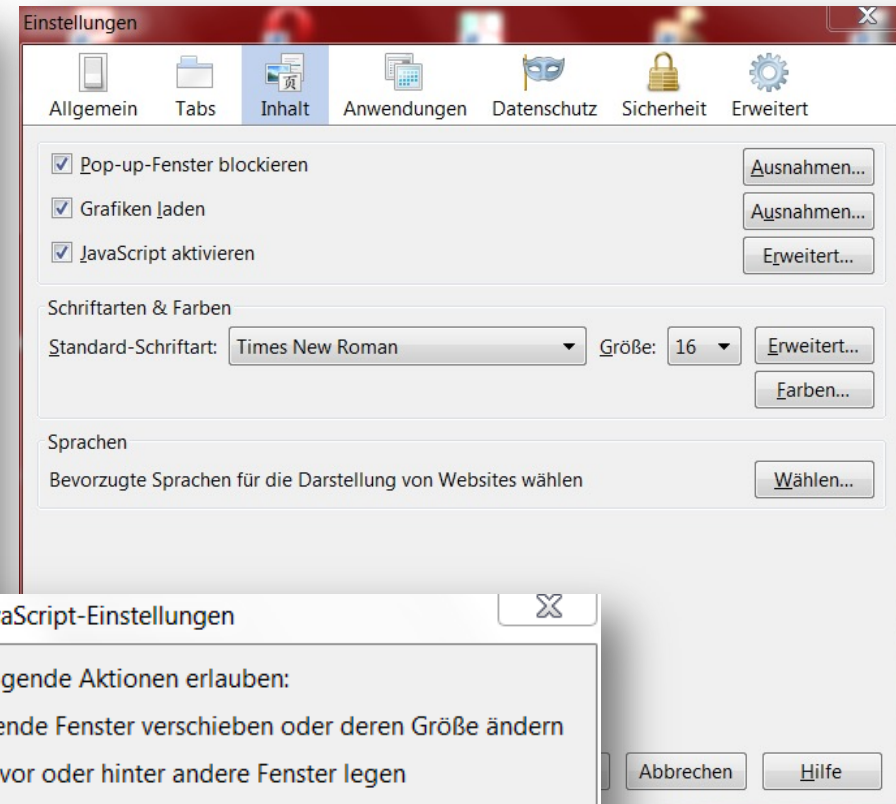
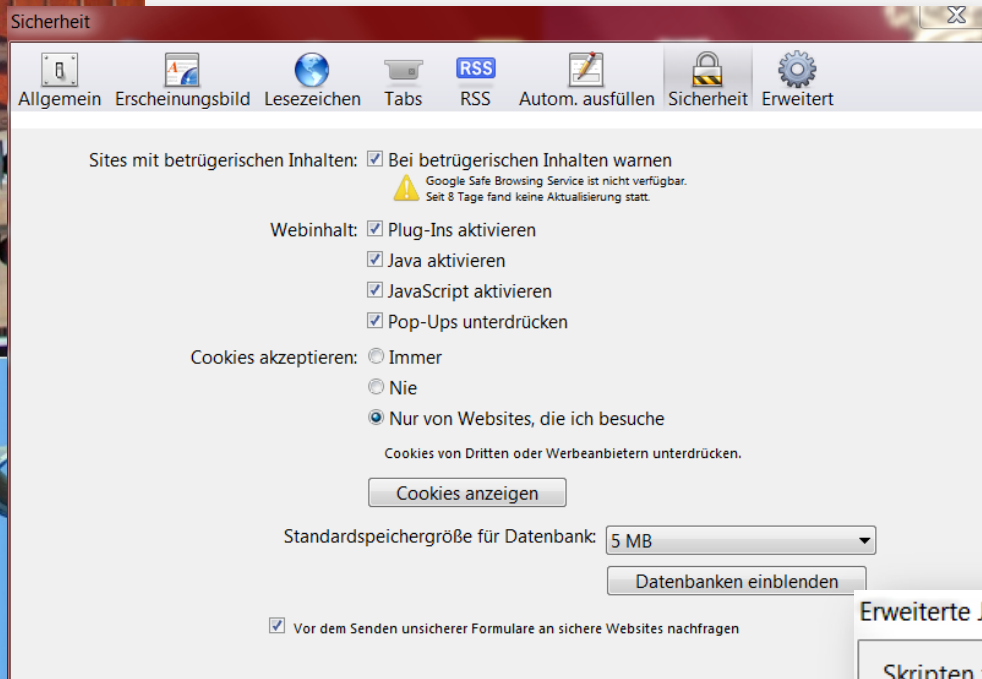
# JavaScript und der Browser

- JavaScript kann vom Benutzer erlaubt/verboten werden
- je nach Browser werden verschiedene Versionen von JavaScript unterstützt
  - Problem für den Diensteanbieter
- engere Verknüpfung von HTML und JavaScript in HTML5



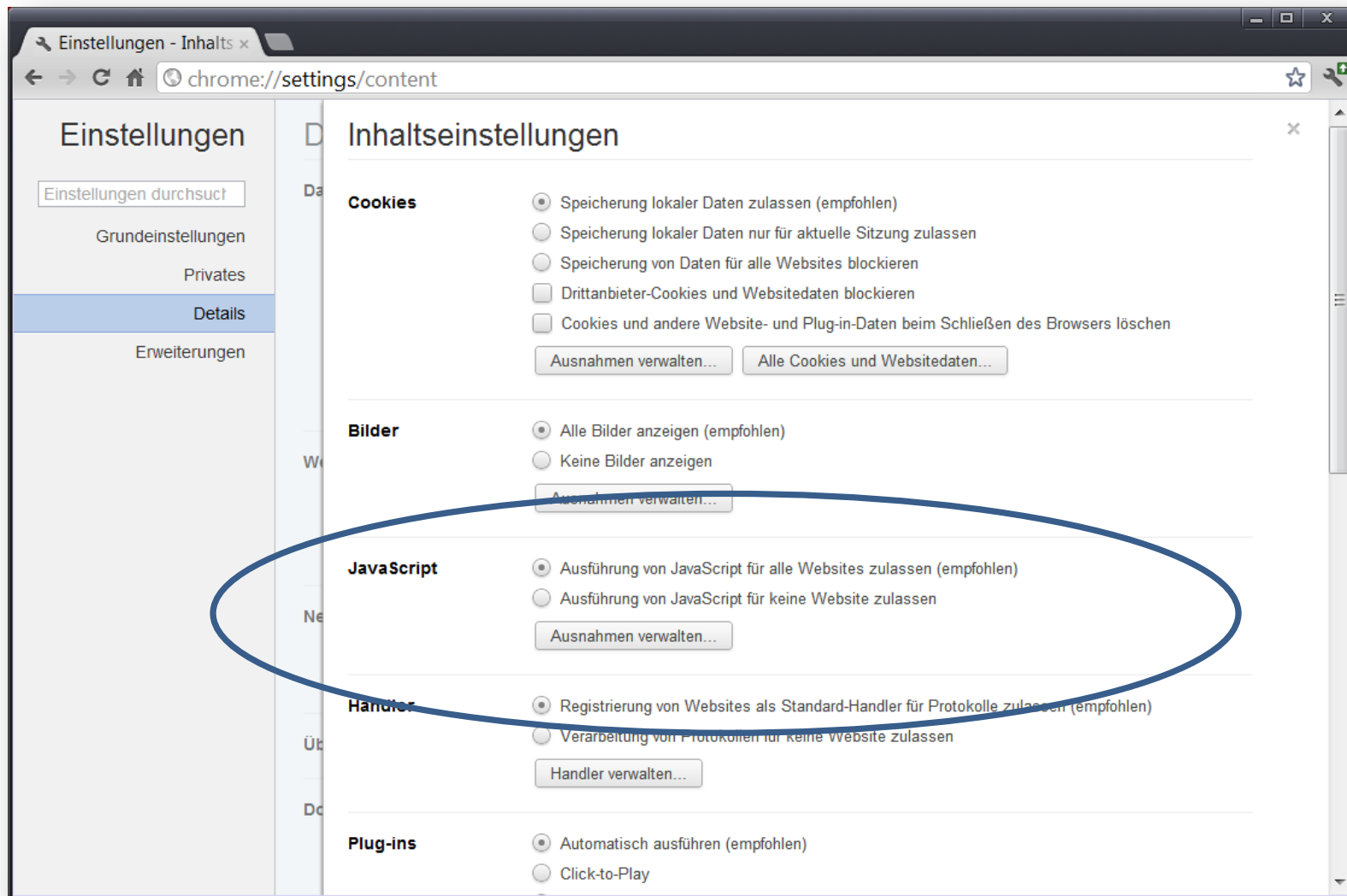


# Browsereinstellungen





# Browsereinstellungen





### Inhaltseinstellungen

**Cookies**

- Speicherung lokaler Daten zulassen (empfohlen)
- Lokale Daten nach Schließen des Browsers löschen
- Speicherung von Daten für alle Websites blockieren
- Drittanbieter-Cookies und Websitedaten blockieren

Ausnahmen verwalten...    Alle Cookies und Websitedaten...

**Bilder**

- Alle Bilder anzeigen (empfohlen)
- Keine Bilder anzeigen

Ausnahmen verwalten...

**JavaScript**

- Ausführung von JavaScript für alle Websites zulassen (empfohlen)
- Ausführung von JavaScript für keine Website zulassen

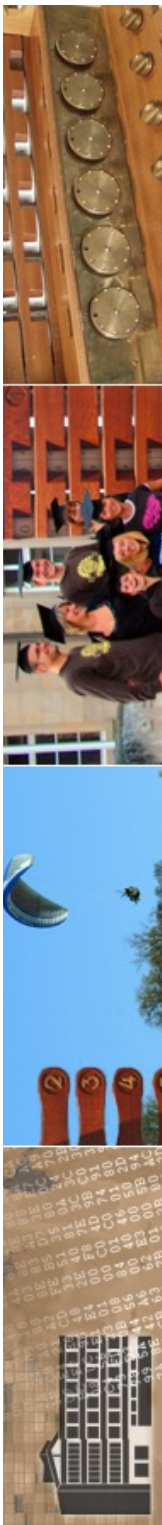
Ausnahmen verwalten...

**Schlüsselgenerierung**

- Schlüsselgenerierung in Formularen auf allen Websites zulassen

Fertig

Proxy-Einstellungen Ihres Computers, um eine Verbindung mit dem





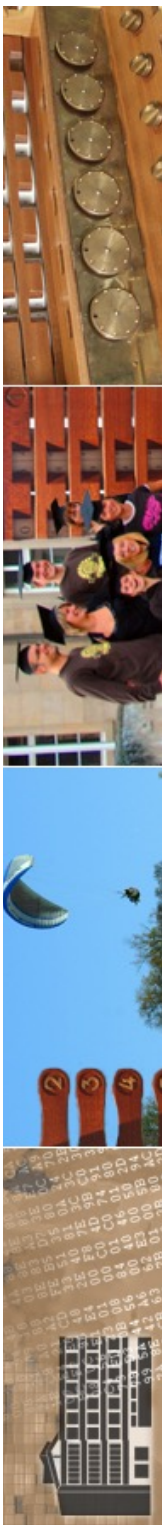
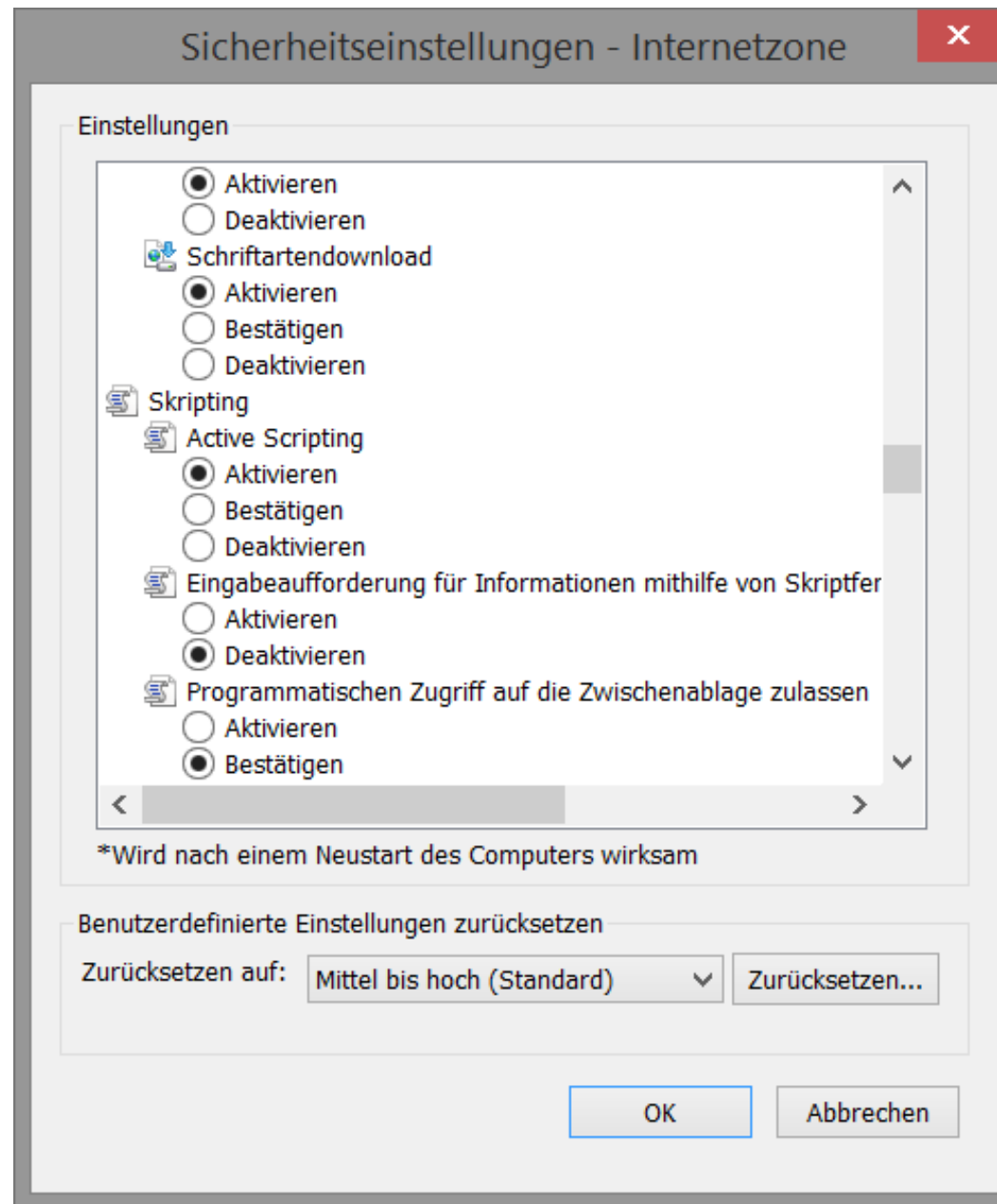
Firefox

about:config

Suchen: javascript

Einstellungsname	Status	Typ	Wert
browser.urlbar.filter.javascript	Standard	boolean	true
javascript.enabled	Standard	boolean	true
javascript.options.asmjs	Standard	boolean	true
javascript.options.baselinejit.chrome	Standard	boolean	true
javascript.options.baselinejit.content	Standard	boolean	true
javascript.options.gc_on_memory_pressure	Standard	boolean	true
javascript.options.ion.chrome	Standard	boolean	false
javascript.options.ion.content	Standard	boolean	true
javascript.options.ion.parallel_compilation	Standard	boolean	true
javascript.options.jit_hardening	Standard	boolean	true
javascript.options.mem.gc_allocation_threshold_mb	Standard	integer	30
javascript.options.mem.gc_decommit_threshold_mb	Standard	integer	32
javascript.options.mem.gc_dynamic_heap_growth	Standard	boolean	true
javascript.options.mem.gc_dynamic_mark_slice	Standard	boolean	true
javascript.options.mem.gc_high_frequency_heap_gro...	Standard	integer	300
javascript.options.mem.gc_high_frequency_heap_gro...	Standard	integer	150
javascript.options.mem.gc_high_frequency_high_limit_...	Standard	integer	500
javascript.options.mem.gc_high_frequency_low_limit_...	Standard	integer	100







🔍 In Einstellungen suchen

← JavaScript

🔍 Suchen

Zugelassen (empfohlen)



Blockieren

Hinzufügen

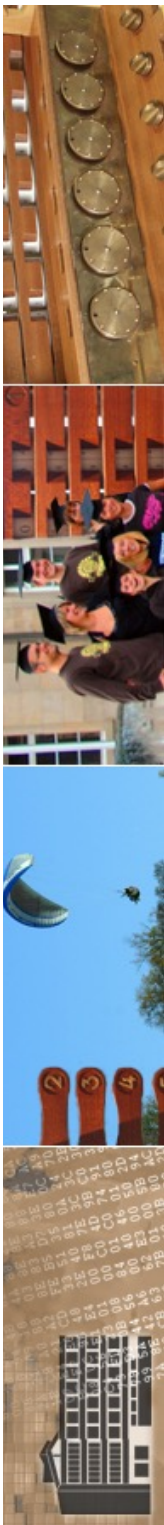
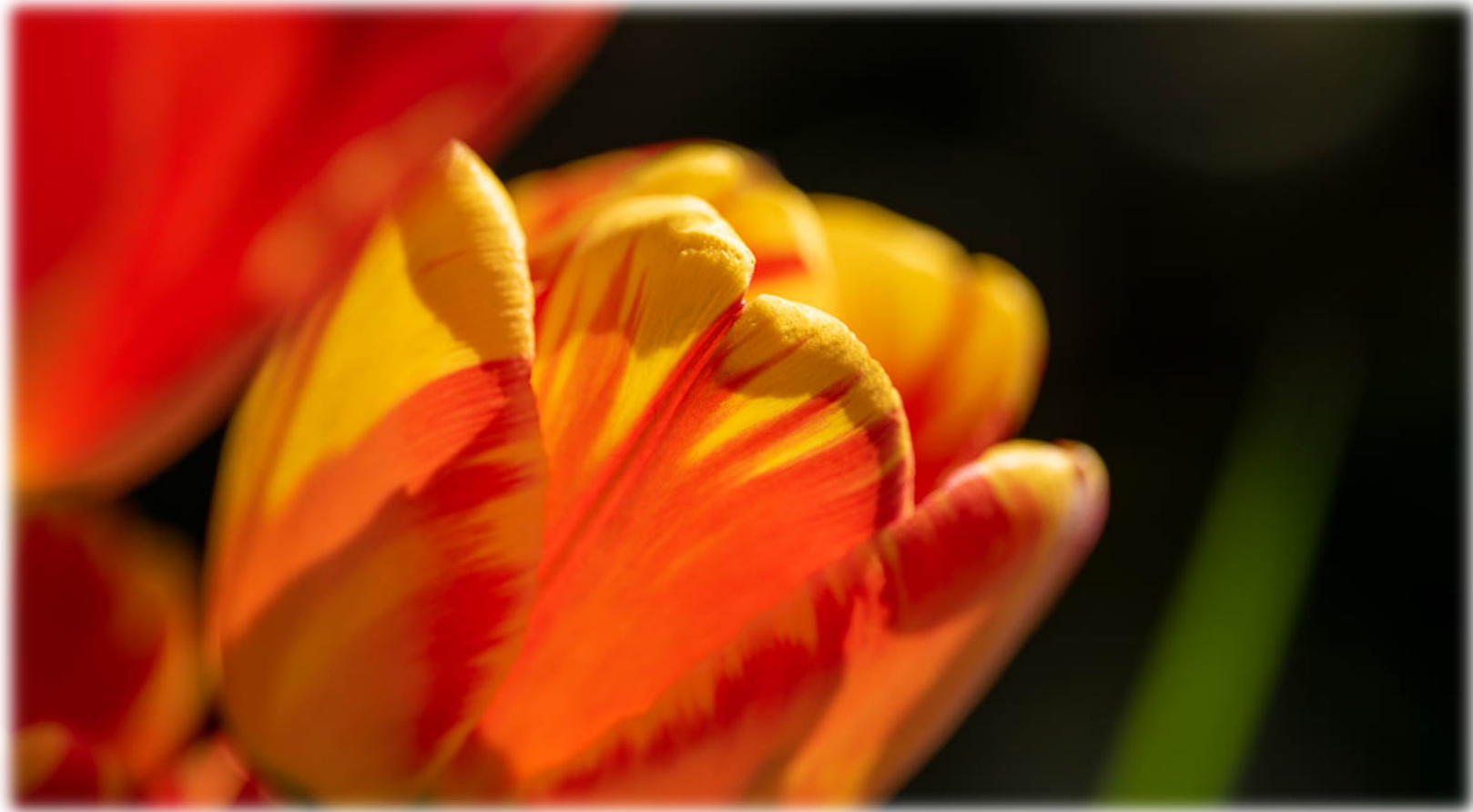
Keine Websites hinzugefügt

Zulassen

Hinzufügen

Keine Websites hinzugefügt









# Kommentare in JavaScript

- Syntax der Kommentare in JavaScript:

`//` Zeilenkommentar

`/*`

Mehrzeilen-  
kommentar

`*/`



# HTML-Tags für JavaScript

- *verschiedene* Tags zur Kennzeichnung von JavaScript
  - `<SCRIPT>`  
     // JavaScript-Code  
   `</SCRIPT>`
  - `<SCRIPT language="JavaScript"> ...`
  - `<SCRIPT type="text/javascript"> ...`
  - auch mit Versionsangabe:
    - `<SCRIPT language="JavaScript1.5"> ...`
    - `<SCRIPT type="text/javascript1.5"> ...`



# ein allererstes JavaScript



JavaScript - GIT JavaScript/HelloWorld.html - Eclipse - /Users/thomas/Documents/eclipse32/workspace

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
5 <title>Grundlagen Internet-Technologien</title>
6 </head>
7 <body>
8     <!-- ein erstes JavaScript -->
9     <script type="text/javascript">
10         // Kommentar in JavaScript
11         document.write("Grundlagen Internet-Technologien (INF3171)<BR />");
12         document.write("Hello <b>JavaScript</b>");
13     </script>
14 </body>
15 </html>
16

```

Outline: DOCTYPE:html, html

Problems: 0 items

Description	Resource	Path	Location	Type

DOCTYPE:html | Writable | Smart Insert | 1 : 1



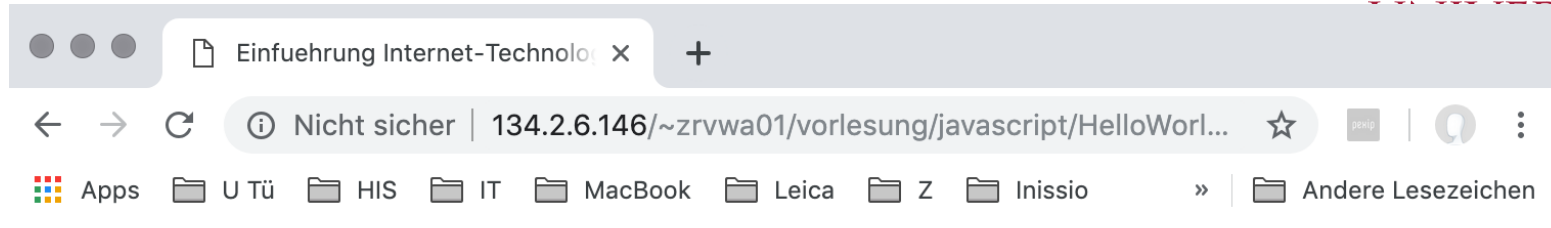


HelloWorld.html

```

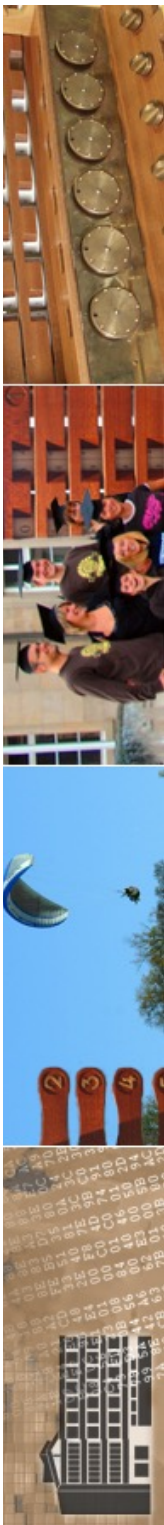
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
5 <title>Grundlagen Internet-Technologien</title>
6 </head>
7 <body>
8     <!-- ein erstes JavaScript -->
9     <script type="text/javascript">
10         // Kommentar in JavaScript
11         document.write("Grundlagen Internet-Technologien (INF3171)<BR />");
12         document.write("Hello <b>JavaScript</b>");
13     </script>
14 </body>
15 </html>

```



# Einführung Internet-Technologien (INF3171)

## Hello JavaScript





JavaScript - GIT JavaScript/HelloWorld.html - Eclipse - /Users/thomas/Documents/eclipse32/workspace

Quick Access | Java | Perl | JavaScript | PHP | Ruby | Resource | PyDev | Web

Projec | Naviga

**Git JavaScript**  
 JavaScript Resources  
 css  
 buch.css  
 buch.dtd  
 Buch.js  
 buch.xml  
 buchhandlung1.html  
 buchhandlung2.html  
 event1.html  
 event2.html  
 fakultaet.html  
 fakultaet.js  
 formulartest.html  
 hellojavascript.html  
 hellojavascript2.html  
**HelloWorld.html**  
 javascript\_template.html  
 location.html  
 navigator.html  
 navigator2.html  
 schleifen.html

```

1 <!DOCTYPE html>
2 <html>
3 <head>
4 <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
5 <title>Grundlagen Internet-Technologien</title>
6 </head>
7 <body>
8   <!-- ein erstes JavaScript -->
9   <script type="text/javascript">
10     // Kommentar in JavaScript
11     document.write("Grundlagen Internet-Technologien (INF3171)<BR />");
12     document.write("Hello <b>JavaScript</b>");
13   </script>
14 </body>
15 </html>
16
  
```

Outline

- DOCTYPE:html
- html
  - <> head
    - <> meta http-equiv=Content-Type
    - title
  - body
    - #comment
    - script type=text/javascript

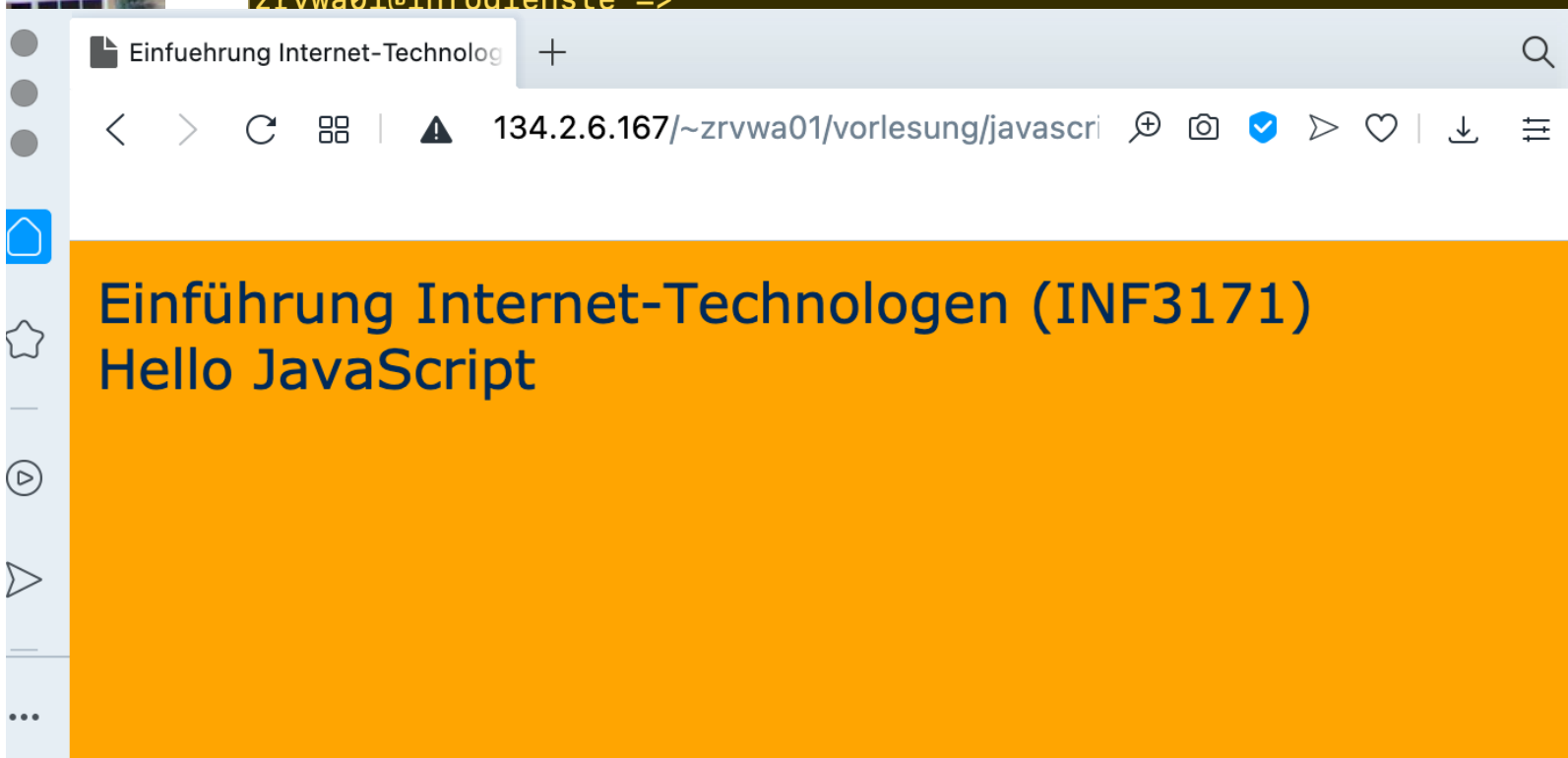
Problems | @ Documentation | Declaration

0 items

Description	Resource	Path	Location	Type
<p>&lt;&gt; html/body/script</p>				



```
zrvwa01@infodienste =>
zrvwa01@infodienste => more HelloWorld.html
<!DOCTYPE html>
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
  <title>Einfuehrung Internet-Technologien</title>
  <link rel="stylesheet" type="text/css" href="./css/webkompendium.css">
</head>
<body>
  <!-- ein erstes JavaScript -->
  <script type="text/javascript">
    document.write("Einf&uuml;hrung Internet-Technologien (INF3171)<BR />");
    document.write("Hello JavaScript");
  </script>
</body>
</html>
zrvwa01@infodienste =>
```



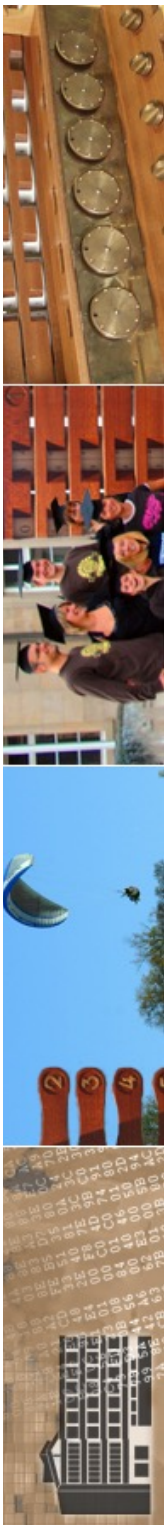


< > ↻ ☰ | ⚠ view-source:134.2.6.146/~zrvwa01/vorlesung/javascript/HelloWorld.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
5   <title>Einfuehrung Internet-Technologien</title>
6   <link rel="stylesheet" type="text/css" href="./css/webkompendium.css">
7 </head>
8 <body>
9   <!-- ein erstes JavaScript -->
10  <script type="text/javascript">
11    document.write("Einf&uuml;hrung Internet-Technologen (INF3171)<BR />");
12    document.write("Hello JavaScript");
13  </script>
14 </body>
15 </html>
16
```







# Testscript version. html

- testen der JavaScript-Version, die der Browser unterstützt

```
zrvwa01@infodienste => more version.html
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="./css/webkompendium.css">
    <link rel="shortcut icon" href="./css/favicon.ico">
    <title>Internet-Technologien: erstes JavaScript</title>
  </head>
  <body>
    <hr><center>
      <H2>Grundlagen Internet-Technologien</H2>
      <H3>Unterstützte JavaScript-Versionen</H3>

      <script language="JavaScript1.1">
        document.write("JavaScript 1.1 m&ouml;glich<BR>");
      </script>

      <script language="JavaScript1.2">
        document.write("JavaScript 1.2 m&ouml;glich<BR>");
      </script>

      <script language="JavaScript1.3">
        document.write("JavaScript 1.3 m&ouml;glich<BR>");
      </script>

      <script language="JavaScript1.4">
        document.write("JavaScript 1.4 m&ouml;glich<BR>");
      </script>

      <script language="JavaScript1.5">
        document.write("JavaScript 1.5 m&ouml;glich<BR>");
      </script>

      <script language="JavaScript1.6">
        document.write("JavaScript 1.6 m&ouml;glich<BR>");
      </script>

      <script language="JavaScript1.7">
        document.write("JavaScript 1.7 m&ouml;glich<BR>");
      </script>

      <script language="JavaScript1.8">
        document.write("JavaScript 1.8 m&ouml;glich<BR>");
      </script>

    </center><hr>

  </body>
</html>
zrvwa01@infodienste =>
```



Internet-Technologi... x +

← → ↻ 🏠 ☆ <http://134.2.2.38/~zrvwa01> ▶ 📄 🔧

## Grundlagen Internet-Technologien

### Unterstützte JavaScript-Versionen

- JavaScript 1.1 möglich
- JavaScript 1.2 möglich
- JavaScript 1.3 möglich
- JavaScript 1.4 möglich
- JavaScript 1.5 möglich
- JavaScript 1.6 möglich
- JavaScript 1.7 möglich

Internet-Technologien: erstes JavaScript - Microsoft I...

Datei Bearbeiten Ansicht Favoriten Extras ?

← Zurück → ↻ 🏠 🔍 Suchen ☆ Favoriten >>

Adresse <http://134.2.2.38/~zrvwa01/vorlesung/jc> ➔ Wechseln zu Links >>

## Grundlagen Internet-Technologien

### Unterstützte JavaScript-Versionen

- JavaScript 1.1 möglich
- JavaScript 1.2 möglich
- JavaScript 1.3 möglich

Fertig Internet

Internet-Technologien: erstes JavaScript -

Datei Bearbeiten Ansicht Chronik Leseze...

← → ↻ 🏠 <http://134.2.2.38/~zrvwa01/vorlesung/jc>

## Grundlagen Internet-Technologien

### Unterstützte JavaScript-Versionen

- JavaScript 1.1 möglich
- JavaScript 1.2 möglich
- JavaScript 1.3 möglich
- JavaScript 1.4 möglich
- JavaScript 1.5 möglich
- JavaScript 1.6 möglich
- JavaScript 1.7 möglich
- JavaScript 1.8 möglich



## Grundlagen Internet-Technologien

### Unterstützte JavaScript-Versionen

JavaScript 1.1 möglich  
JavaScript 1.2 möglich  
JavaScript 1.3 möglich

## Grundlagen Internet-Technologien

### Unterstützte JavaScript-Versionen

JavaScript 1.1 möglich  
JavaScript 1.2 möglich  
JavaScript 1.3 möglich  
JavaScript 1.4 möglich  
JavaScript 1.5 möglich  
JavaScript 1.6 möglich  
JavaScript 1.7 möglich

## Grundlagen Internet-Technol

### Unterstützte JavaScript-Versionen

JavaScript 1.1 möglich  
JavaScript 1.2 möglich  
JavaScript 1.3 möglich  
JavaScript 1.4 möglich  
JavaScript 1.5 möglich



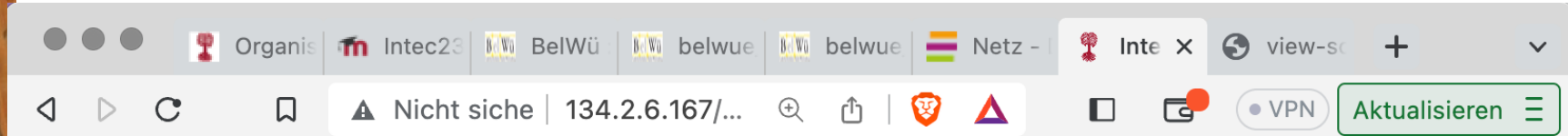
Internet-Technologien: erstes J x +

Nicht sicher | 134.2.6.167/~zrvwa01/vorlesung/javascript/version.html

# Grundlagen Internet-Technologien

## Unterstützte JavaScript-Versionen

- JavaScript 1.1 möglich
- JavaScript 1.2 möglich
- JavaScript 1.3 möglich
- JavaScript 1.4 möglich
- JavaScript 1.5 möglich
- JavaScript 1.6 möglich
- JavaScript 1.7 möglich



# Grundlagen Internet- Technologien

## Unterstützte JavaScript-Versionen

JavaScript 1.1 möglich  
JavaScript 1.2 möglich  
JavaScript 1.3 möglich  
JavaScript 1.4 möglich  
JavaScript 1.5 möglich



# des „Anti-Tag“

- es gibt auch die umgekehrte Intention:
  - `<noscript>`
  - ...
  - `</noscript>`
- wird *nur* ausgeführt, wenn kein JavaScript ausgeführt werden kann
  - sinnvoll für Warnungen, dass JavaScript fehlt





# Debugging von JavaScript

- über Browser-Funktionen
  - Fehler-Konsole von Firefox
  - JavaScript-Konsole von Chrome
  - Firebug-Plugin
  - Entwickler-Tools von Chrome





# Grundlagen Internet- Technologien

## Unterstützte JavaScript- Versionen

JavaScript 1.1 möglich  
JavaScript 1.2 möglich  
JavaScript 1.3 möglich  
JavaScript 1.4 möglich  
JavaScript 1.5 möglich

```

11 <H3>Unterstützte JavaScript-Versionen</
12
13 <script language="JavaScript1.1">
14   document.write("JavaScript 1.1 möglich");
15 </script>
16
17 <script language="JavaScript1.2">
18   document.write("JavaScript 1.2 möglich");
19 </script>
20
21 <script language="JavaScript1.3">
22   document.write("JavaScript 1.3 möglich");
23 </script>
24
25 <script language="JavaScript1.4">
26   document.write("JavaScript 1.4 möglich");
27 </script>
28
29 <script language="JavaScript1.5">
30   document.write("JavaScript 1.5 möglich");
31 </script>
32
33 <script language="JavaScript1.6">

```







# Grundlagen Internet-Technologien

## Unterstützte JavaScript-Versionen

- JavaScript 1.1 möglich
- JavaScript 1.2 möglich
- JavaScript 1.3 möglich
- JavaScript 1.4 möglich
- JavaScript 1.5 möglich
- JavaScript 1.6 möglich
- JavaScript 1.7 möglich

Elements Console Sources Network Performance

```
<!doctype html>
<html>
  <head>...</head>
  <body>
    <hr>
    <center>
      <h2>Grundlagen Internet-Technologien</h2>
      <h3>Unterstützte JavaScript-Versionen</h3>
      <script language="JavaScript1.1">
        document.write("JavaScript 1.1 möglich<br>");
      </script> == $0
      "JavaScript 1.1 möglich"
      <br>
      <script language="JavaScript1.2">
        document.write("JavaScript 1.2 möglich<br>");
      </script>
      "JavaScript 1.2 möglich"
      <br>
      <script language="JavaScript1.3">
        document.write("JavaScript 1.3 möglich<br>");
      </script>
      "JavaScript 1.3 möglich"
      <br>
      <script language="JavaScript1.4" ">
        document.write("JavaScript 1.4 möglich<br>");
      </script>
    </center>
  </body>
</html>
```

html body center script

Styles Event Listeners DOM Breakpoints Properties Accessibility

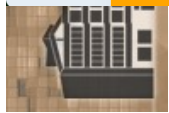
Filter :hov .cls +

```
element.style {
}
script {
  display: none;
}
Inherited from center
center {
  text-align: -webkit-center;
}
Inherited from body
body {
  background-color: #FFA500;
  font-family: Verdana, Arial, Helvetica, sans-serif;
  color: #002A5A;
}
```

margin border padding auto x auto

Filter Show all

- color: rgb(0, 42, 90)
- display: none
- font-family





# Variablen in JavaScript

- JavaScript ist eine **Scriptsprache**:  
Variablen sind **nicht explizit typisiert**
- optionales Schlüsselwort **var** :  
`var sinn_des_lebens = 42;`
- **Gültigkeit**:
  - Variablen in Methoden sind **lokal**
  - Variablen im Hauptblock sind **global**



# dynamische Datentypen

- intern verwendet JavaScript folgende Datentypen:
  - ganze Zahlen
  - Gleitkommazahlen
  - Zeichenketten
  - Boolesche Werte
  - Objekte
- Typabfrage: **`typeof` variablenbezeichner**





# Zeichenketten

- Zeichenketten werden durch ' oder " begrenzt
- Zeichenkettenkomposition durch + Operator

```
- var s = "Hello "; var t = "World";  
  s = s + t;
```

```
var n = s.length;    // n bekommt Wert 11  
var u = s.substring(1,4); // ell
```

```
// fuer Informatiker:  
var v = new String("FBI");
```



```

zrvwa01@infodienste => more variablen.html
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="./css/webkompendium.css">
    <link rel="shortcut icon" href="./css/favicon.ico">
    <title>Internet-Technologien: JavaScript</title>
  </head>
  <body>

    <hr><center>
      <h2>Einführung Internet-Technologien</h2>
      <h3>Variablen in JavaScript</h3>

      <script type="text/javascript">
        <!--
          var sinn_des_lebens = 042;
          var wk = "Grundlagen " + "Internet-Technologien";
          var s = wk.substring(11,19)

          document.write("Substring: "+s);
          document.write("<BR /> Wert der Variablen: "+sinn_des_lebens);

          // -->
        </script>

    </center><hr>

  </body>
</html>
zrvwa01@infodienste =>

```



```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="./css/webkompendium.css">
    <link rel="shortcut icon" href="./css/favicon.ico">
    <title>Internet-Technologien: JavaScript</title>
  </head>
  <body>

    <hr><center>
      <h2>Einführung Internet-Technologien</h2>
      <h3>Variablen in JavaScript</h3>

      <script type="text/javascript">
        <!--
        var sinn_des_lebens = 042;
        var wk = "Grundlagen " + "Internet-Technologien";
        var s = wk.substring(11,19)

        document.write("Substring: "+s);
        document.write("<BR /> Wert der Variablen: "+sinn_des_lebens);

        // -->
      </script>

    </center><hr>

  </body>
</html>
```

~





Internet-Technologien: JavaScript x +

Nicht sicher | 134.2.6.167/~zrvwa01/vorlesung/javascript/variablen.html

# Einführung Internet-Technologien

## Variablen in JavaScript

Substring: Internet  
Wert der Variablen: 34



# Arrays in JavaScript

- Arrays sind Listen
- numerischer Index genauso wie assoziatives Array
  - `var liste = new array();`
  - `liste[0] = 42;`
  - `liste["tuebingen"] = "FBI";`
- Listen-Methoden wie
  - `push`   `pop`   `shift`   `sort`   ...





# Logik, Kontrollstrukturen

- ...das übliche aus Java:
  - !    ||    &&
- Kontrollstrukturen: ebenfalls (fast) wie in Java
  - { ... }
  - if (boole)    {...}    else    {...}
  - switch (n) {    case ...    default ... }
  - while (boole) {...}
  - do {...}    while (boole);
  - for (init;boole;expr)
  - break    continue
  - for (laufvariable in struktur)





# Grundlagen Internet-Technologien

## Schleifen in JavaScript

Summierung mittels while: 5050  
Summierung mittels do-while: 5050  
Summierung mittels for: 5050

```
zrvwa01@infodienste => more schleifen.html
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="./css/webkompendium.css">
    <link rel="shortcut icon" href="./...>
    <title>Intenret-Technologien: JavaS
  </head>
  <body>
    <hr><center>
      <h2>Grundlagen Internet-Tec
      <h3>Schleifen in JavaScript

      <script type="text/javascri

        var max = 100;

        // while-Schleife
        var summe = 0;
        var i = 1;

        while (i <= max)
          summe += i;
        document.write("<BR>Summierung mittels while: "+summe);

        // do-while
        summe = 0; i = 1;
        do {
          summe += i++;
        } while (i <= max);
        document.write("<BR>Summierung mittels do-while: "+summe);

        // for-schleife
        summe = 0;
        for (i = 1; i <= max; i++)
          summe += i;
        document.write("<BR>Summierung mittels for: "+summe);

      </script>

    </center><hr>

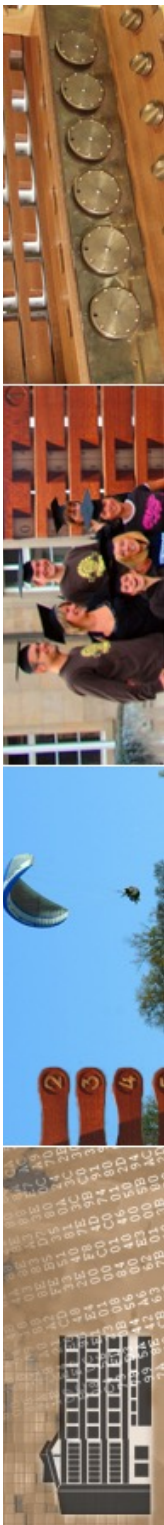
  </body>
</html>
zrvwa01@infodienste =>
```



$$\sum_{i=1}^{100} i = \frac{n(n-1)}{2}$$

$$n = 100$$







# Methoden in JavaScript

- Schlüsselwort `function` :
  - `function fakultaet(n) {`
  - `if (n < 1) return 1;`
  - `return n * fakultaet(n-1);`
  - `}`
  
- einige besondere Methoden:
  - `alert(arg)`
  - `parseInt(arg)`
  - `parseFloat(arg)`
  - `isNaN(arg)`



# Importieren von JavaScript

- JavaScript-Code kann nachgeladen werden
  - `<SCRIPT language="JavaScript" src="file.js"/>`
- zentrale Funktionalität für verteilten Code
- src von gleichem Server wie Ausgangsdokument oder mittels http von anderem Server

```
zrvwa01@infodienste => more fakultaet.html
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="./css/webkompendium.css">
    <link rel="shortcut icon" href="./css/favicon.ico">
    <title>Internet-Technologien: JavaScript</title>

    <script src="http://134.2.6.146/~zrvwa01/vorlesung/javascript/fakultaet.js"></script>

  </head>
  <body>

    <hr><center>
      <h2>Grundlagen Internet-Technologien</h2>
      <h3>Externes JavaScript</h3>

      <h4> Fakult&auml;
zrvwa01@infodienste => more fakultaet.js
      <script type="text/javascript">
        * Grundlagen Internet-Technologien
        * rekursive Fakult&auml;t in JavaScript
        */
      </script>
    </h4></center><hr>

  </body>
</html>
```

```
zrvwa01@infodienste =>
function fakultaet(n) {
  if (n < 1) return 1;
  return n * fakultaet(n-1);
}
zrvwa01@infodienste =>
```





Internet-Technologien: JavaScript x +

Nicht sicher | 134.2.6.167/~zrvwa01/vorlesung/javascript/fakultaet.html

# Grundlagen Internet-Technologien

## Externes JavaScript

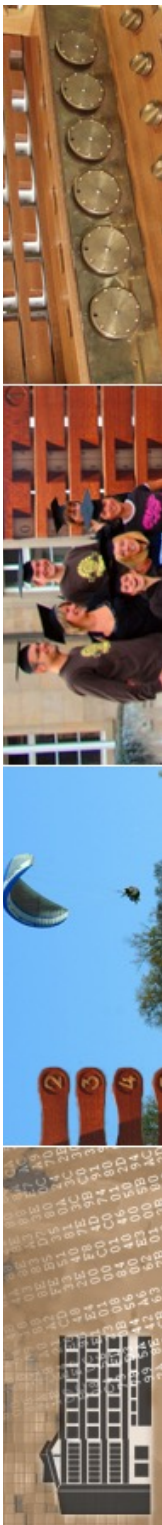
### Fakultät(5) = 120





# Objektorientierung in JavaScript

- JavaScript ist OO
  - ...zumindest in der Werbung...
- **kein** Schlüsselwort `class`
- *Klassen werden über eine function definiert, die den Konstruktor bildet*
  - keine Klassen, Instanzen, ... - rein funktionaler Zugang
  - keine Vererbung im eigentlichen Sinne da keine Klasse
  - keine Datenkapselung







```
[zrvwa01@infodienste => more Buch.js
/*
 * Grundlagen Internet-Technologien
 * Objektorientierung in JavaScript
 */

// Konstruktor
function Buch(titel, autor, verlag, jahr) {

    this.titel = titel;
    this.autor = autor;
    this.verlag = verlag;
    this.jahr = jahr;

    this.toString = toString;
    this.htmlRow = htmlRow;
} // Konstruktor

// reine Textausgabe
function toString() {
    return this.autor + ": "+this.titel+" ("+this.verlag+", "+this.jahr+");
} // toString

// HTML-Tabellenzeile
function htmlRow() {
    return "<tr><td>"+this.titel+"</td><td>"+this.autor+
        "</td><td>"+this.verlag+"</td><td>"+this.jahr+"</td></tr>";
} // htmlRow
[zrvwa01@infodienste =>
```



```
zrvwa01@infodienste => more buchhandlung2.html
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="./css/webkompendium.css">
    <link rel="shortcut icon" href="./css/springer_icon.ico">
    <title>Internet-Technologien: JavaScript</title>

    <script src="Buch.js" type="text/javascript"></script>

    <script type="text/javascript">
      var wk = new Buch("Webkompendium", "Thomas Walter", "Springer", 2008);
      var mf = new Buch("MediaFotografie", "Thomas Walter", "Springer", 2005);
    </script>

  </head>
  <body>

    <hr><center>
      <h2>Grundlagen Internet-Technologien</h2>
      <h3>Objektorientierung in JavaScript</h3>

      <table>
        <script type="text/javascript">
          document.write(wk.htmlRow());
          document.write(mf.htmlRow());
        </script>

      </table>
    </center><hr>

  </body>
</html>
```

zrvwa01@infodienste =>



Internet-Technologien: JavaScript x +

Nicht sicher | 134.2.6.167/~zrvwa01/vorlesung/javascript/buchhandlun...

# Grundlagen Internet-Technologien

## Objektorientierung in JavaScript

Webkompendium Thomas Walter Springer 2008  
MediaFotografie Thomas Walter Springer 2005



# ...und nun...

- haben wir das Prinzip der clientseitigen  
Scriptsprache JavaScript kennen gelernt

- als nächstes:

JavaScript und DOM:

die **Verbindung von  
JavaScript zu HTML**

