



Einführung Internet-Technologien

INF3171

Clientseitige Web-Programmierung
mit JavaScript

Version 1.0

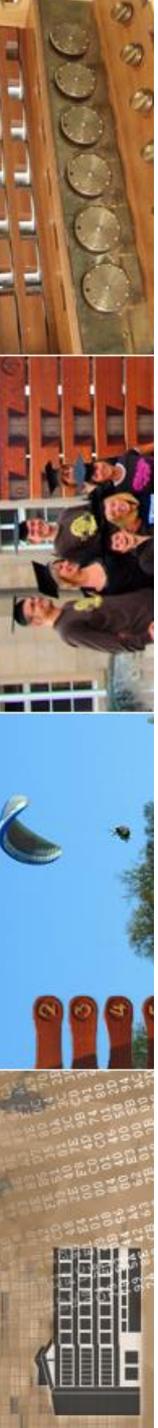
06.05.2013





aktuelles

Vielen Dank an den Fachbereich
Physik für die Bereitstellung des
Übungsraums D2 A38.





clientseitige Web-Programmierung

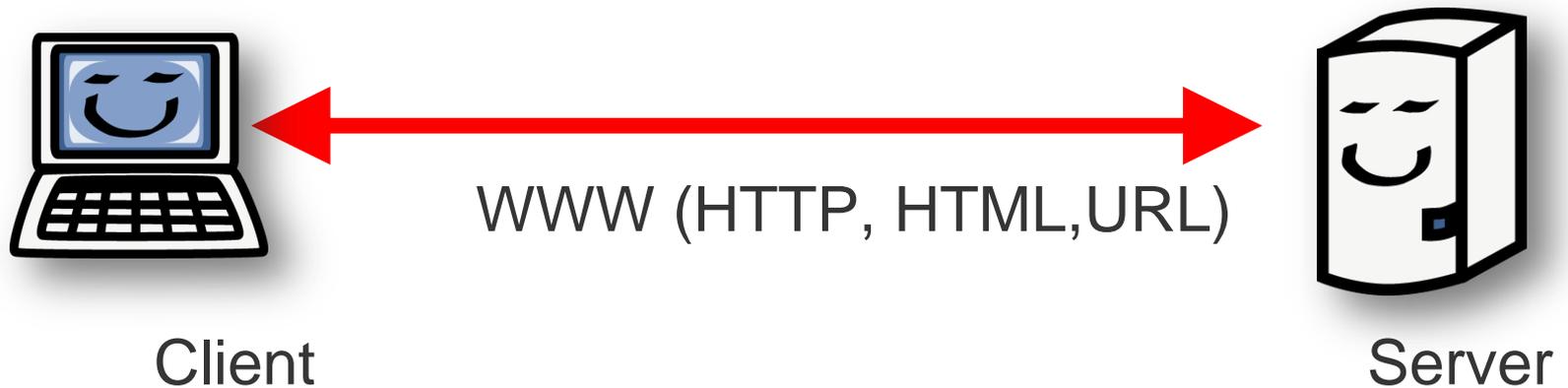
- verschiedene Ansätze
 - JavaScript
 - Java Applet
 - Adobe Flash
 - Microsoft Silverlight
 - RIA: rich internet application





Architektur

- Situation:



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



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



Programmierung / Skripting

- Grundlegende Elemente
 - Anweisung
 - Ist eine Vorschrift, die im Rahmen der Abarbeitung eines Programms ausgeführt werden soll. Der Syntax ist abhängig von der eingesetzten Sprache.
 - Variable
 - Ist ein Behälter für Werte, die während der Abarbeitung eines Programms auftreten.
 - Unterscheidung Deklaration und Initialisierung
 - *Typbezeichner Bezeichner = Wert*



Programmierung / Skripting

– Kontrollstrukturen

- Steuern den Ablauf eines Programms
- Werden über Schlüsselwörter im Programm gekennzeichnet
 - Sind explizit für ihre Kontrollstruktur vorgesehen
 - Können nicht für eigene Namensdefinitionen (z.B. Variablenbezeichner) verwendet werden
- ***Startpunkt und Endpunkt***
 - Definieren den Bereich, in dem Programmcode vorliegt. Abarbeitung: Vom Start- zum Endpunkt



Programmierung / Skripting

- *Bedingte Anweisungen und Verzweigungen*

- Programmteile werden nur dann ausgeführt, wenn eine Bedingung erfüllt ist.
- In der Regel als „If-Statement“ bezeichnet

if(Bedingung1){ tue etwas; }

else if(Bedingung2){tue etwas anderes;}

else if(Bedingung3){tue etwas drittes;}

...

else if(Bedingungn){tue etwas ganz anderes;}

else{wenn keine der vorherigen Bedingungen erfüllt wurde;}

```
int i = 1;
if(i>=0 && i<5){
    i = 5;
}
else if(i>=5 && i<10){
    i = 10;
}
else{
    i = 20;
}
```



Programmierung / Skripting

- *Wiederholungen (Schleifen)*

- Wiederholen den umschlossenen Programmcode so lange, bis eine Bedingung erfüllt ist.
- Kopfgesteuerte vs. fußgesteuerte Schleifen

```
int i = 1;
while (i<10){
    i = i + 1;
}
```

`while(Bedingung){tue etwas;}`

`for(Bedingung){ tue etwas; }`

```
int j = 0;
for(int i=0; i<10; i=i+1){
    j = j + i;
}
```

```
int i = 1;
do{
    i = i + 1;
}while (i<10);
```

`do{tue etwas;}while(Bedingung);`



Modellierung eines Programmablaufs

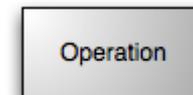
- **Flussdiagramme**

- Normung in DIN 66001, bereits 1983
- Vereinfachung von Programmabläufen über grafische Veranschaulichung
- Modellierung der zuvor eingeführten Elemente

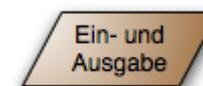
- **Programmstart- und -endpunkt**
 - Tritt einmalig auf



- **Anweisung**



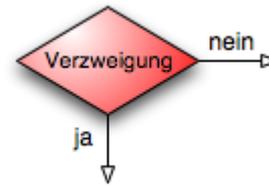
- **(Nutzer-) Interaktion**





Modellierung eines Programmablaufs

- Verzweigung



- Programmfluss

– Läuft vom Startpunkt zum Endpunkt



- Zusätzlich: Unterprozeduraufrufe

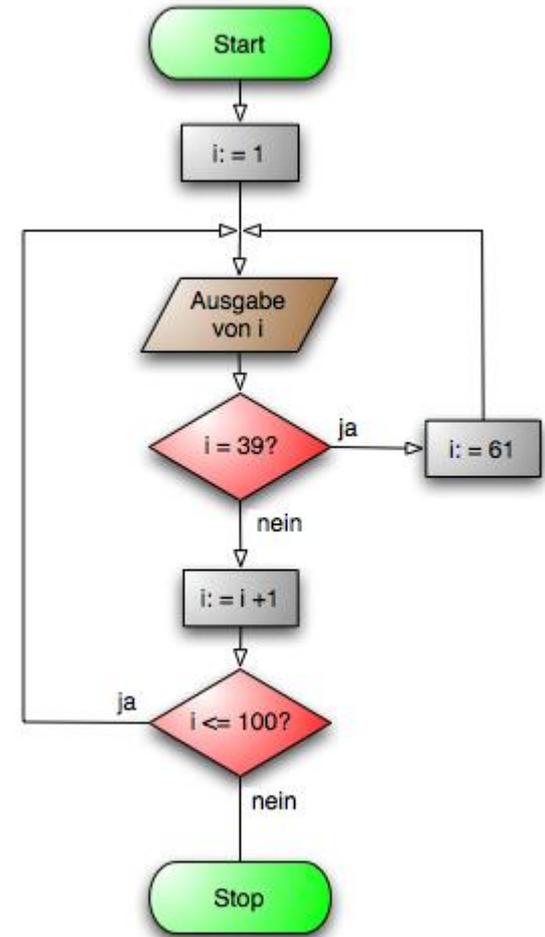
– Wenn Algorithmen komplex sind unterteilt man sie in eigenständige Unteralgorithmen, welche wiederum unterteilt werden können





Modellierung eines Programmablaufs

- **Wo sind die Schleifen?**
 - Programmfluss, der entgegen des Programmablaufs läuft.





Programmiersprachen

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



Muhammed al-Chwarizmi
783 - 850



Programmiersprachen

- Hauptbestandteile von Programmiersprachen
 - Variablen: Name für Ort im Speicher
 - 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



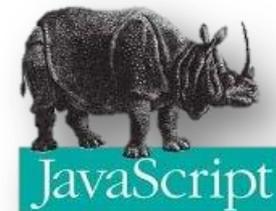
Scriptsprachen

- in dieser Veranstaltung benutzen wir drei Scriptsprachen vertieft:
 - JavaScript
 - clientseitige Web-Programmierung
 - Perl
 - CGI-Programmierung
 - PHP
 - serverseitige Web-Programmierung



JavaScript

- 1995 Brendan Eich: LiveScript (Netscape)
- 1996 Umbenennung in JavaScript
- Klassiker: Version 1.5
- aktuell Version 1.8.5
- Literatur:
 - RRZN: JavaScript
 - Flanagan, David:
JavaScript - The Definitiv Guide, O'Reilly





Selfhtml

SELFHTML: JavaScript, x

de.selfhtml.org/javascript/index.htm

SELFHTML/Navigationshilfen

JavaScript/DOM

Sinnvolle Vorkenntnisse zum Verständnis: [HTML](#) und die einführenden Bemerkungen zu [JavaScript](#).

Dieser Abschnitt dokumentiert die Programmiersprache JavaScript, die in HTML-Dateien eingebunden werden kann.

Der Abschnitt besteht aus folgenden Seiten und Unterabschnitten:

- [Einführung in JavaScript und DOM](#)
- [JavaScript-Sprachelemente](#)
- [Objektreferenz](#)
- [Anwendungsbeispiele](#)

SELFHTML aktuell

Im Online-Angebot von SELFHTML aktuell finden Sie weitere Informationen und Links zu anderen Quellen im Web. Folgende Inhalte kommen in Frage:

- [Artikel zu JavaScript](#)
- [Links zu JavaScript und DOM](#)



JavaScript

- zunächst ist JavaScript eine moderne, aber einfache Skriptsprache
 - 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



Editoren für JavaScript

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





JavaScript und der Browser

- JavaScript kann vom Benutzer erlaubt/verboten werden
- je nach Browser werden verschiedene Versionen von JavaScript unterstützt
 - Problem für den Diensteanbieter



Browsereinstellungen

Sicherheit

Allgemein Erscheinungsbild Lesezeichen Tabs RSS Autom. ausfüllen Sicherheit **Erweitert**

Sites mit betrügerischen Inhalten: Bei betrügerischen Inhalten warnen
Google Safe Browsing Service ist nicht verfügbar.
 Seit 8 Tage fand keine Aktualisierung statt.

Webinhalt: Plug-Ins aktivieren
 Java aktivieren
 JavaScript aktivieren
 Pop-Ups unterdrücken

Cookies akzeptieren: Immer
 Nie
 Nur von Websites, die ich besuche
Cookies von Dritten oder Werbeanbietern unterdrücken.

Standardspeichergöße für Datenbank:

Vor dem Senden unsicherer Formulare an sichere Websites nachfragen

Einstellungen

Allgemein Tabs **Inhalt** Anwendungen Datenschutz Sicherheit **Erweitert**

Pop-up-Fenster blockieren
 Grafiken laden
 JavaScript aktivieren

Schriftarten & Farben
 Standard-Schriftart: Größe:

Sprachen
 Bevorzugte Sprachen für die Darstellung von Websites wählen

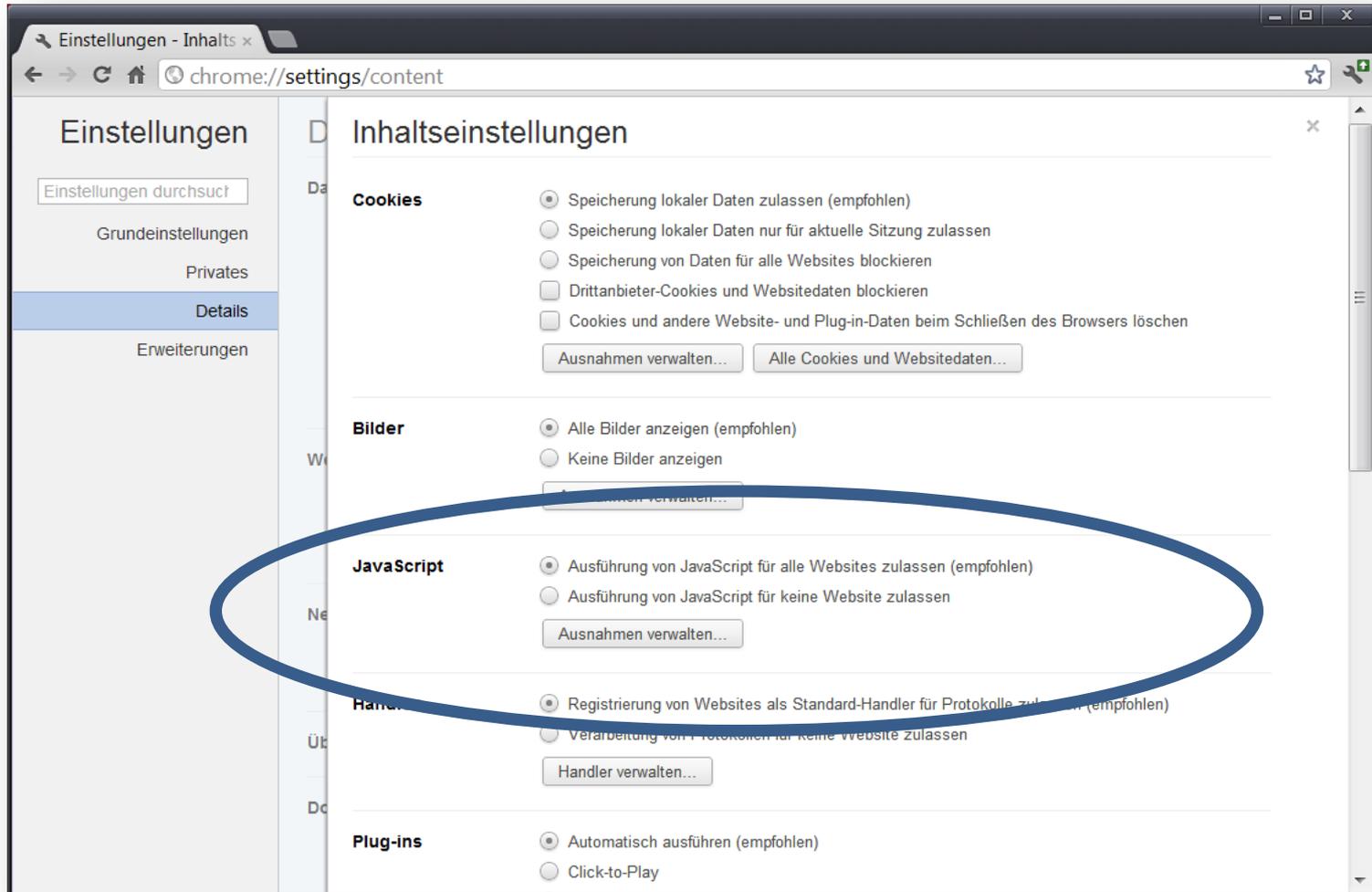
Erweiterte JavaScript-Einstellungen

Skripten folgende Aktionen erlauben:

Existierende Fenster verschieben oder deren Größe ändern
 Fenster vor oder hinter andere Fenster legen
 Das Kontextmenü deaktivieren oder ersetzen
 Statusleiste ausblenden
 Statusleistentext ändern



Browsereinstellungen





Kommentare in JavaScript

- Syntax der Kommentare in JavaScript:

// Zeilenkommentar

/*

Mehrzeilen-
kommentar

*/



HTML-Tags für JavaScript

- *verschiedene* Tags zur Kennzeichnung von JavaScript
 - `<SCRIPT>`

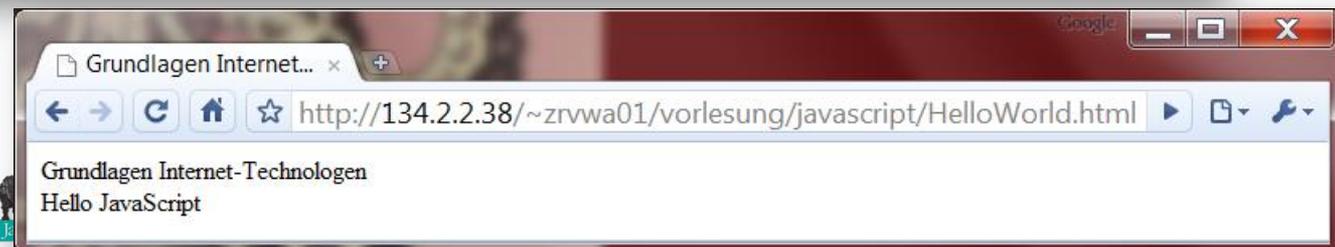
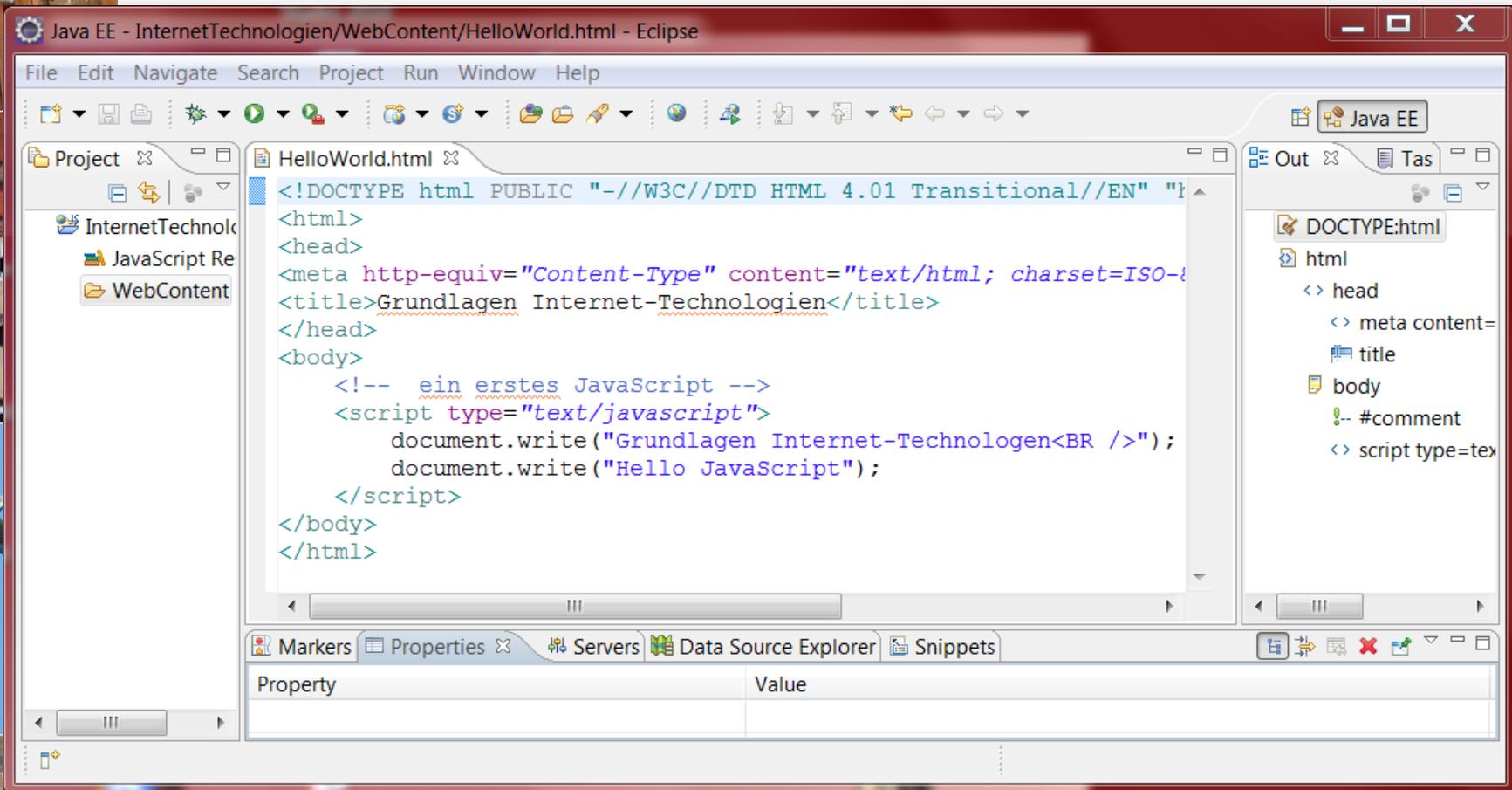
```

                // 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



Testscript version.html

- testen der JS-Version, die der Browser unterstützt

```

Java EE - InternetTechnologien/WebContent/version.html - Eclipse
File Edit Source Navigate Search Project Run Window Help
HelloWorld.html version.html
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.
<html>
  <head>
    <link rel="stylesheet" type="text/css" href="./css/webkompodium.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>
DOCTYPE:html Writable Smart Insert 1:1

```



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/ja> → Wechseln zu Links

Grundlagen Internet-Technologien

Unterstützte JavaScript-Versionen

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

Internet-Technologien: erstes JavaScript -

Datei Bearbeiten Ansicht Chronik Leseze

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

Internet-Technologien: erste...

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



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
 - Firebug-Plugin





Variablen in JavaScript

- JavaScript ist eine **Scriptsprache**:
Variablen sind **nicht 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("WSI");
```



Java EE - InternetTechnologien/WebContent/variablen.html - Eclipse

File Edit Source Navigate Search Project Run Window Help

Java EE

HelloWorld.html version.html variablen.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/favicon.ico">
    <title>Internet-Technologien: JavaScript</title>
  </head>
  <body>

    <hr><center>
      <h2>Grundlagen 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>

```

DOCTYPE:html Writable Smart Insert 1:1





Arrays in JavaScript

- Arrays sind Listen
- numerischer Index genauso wie assoziatives Array
 - `var liste = new array();`
 - `liste[0] = 42;`
 - `liste["tuebingen"] = "WSI";`
- 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)



Java EE - InternetTechnologien/WebContent/schleifen.html - Eclipse

File Edit Source Navigate Search Project Run Window Help

HelloWorld.html version.html variablen.html

```

<script type="text/javascript">

    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>

```

Writable Sm...rt

Internet-Technologi... x

Google

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

Grundlagen Internet-Technologien

Schleifen in JavaScript

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





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



Java EE - InternetTechnologien/WebContent/fakultaet.html - Eclipse

```

File Edit Source Navigate Search Project Run Window Help
HelloWorld.html version.html variablen.html schleifen.html fakultaet.html x fakultaet.js
1<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/lo
2<html>
3  <head>
4    <link rel="stylesheet" type="text/css" href="./css/webkompendium.css">
5    <link rel="shortcut icon" href="./css/favicon.ico">
6    <title>Internet-Technologien: JavaScript</title>
7
8    <script src="http://134.2.2.38/~zrvwa01/vorlesung/javascript/fakultaet.js"></script>
9
10  </head>
11  <body>
12
13    <hr><center>
14      <h2>Grundlagen Internet-Technologien</h2>
15      <h3>Externes JavaScript</h3>
16
17      <h4> Fakult&auml;t(5) =
18
19      <script type="text/javascript">
20        document.write(fakultaet(5));
21      </script>
22
23    </h4></center><hr>
24
25  </body>
26</html>

```

DOCTYPE:html Writable Smart Insert 1:

Java EE - InternetTechnologien/WebContent/fakultaet.js - Eclipse

```

File Edit Source Refactor Navigate Search Project Run Window Help
schleifen.html fakultaet.html fakultaet.js x
1e/*
2 * Grundlagen Internet-Technologien
3 * rekursive Fakult&auml;t in JavaScript
4 */
5function fakultaet(n) {
6
7    if (n < 1) return 1;
8    return n * fakultaet(n-1);
9
10 }

```

Writable Smart Insert 1:1





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



```

Java EE - InternetTechnologien/WebContent/Buch.js - Eclipse
File Edit Source Refactor Navigate Search Project Run Window Help
variablen.html schleifen.html fakultaet.html fakultaet.js Buch.js x 2
1 | /*
2 | * Grundlagen Internet-Technologien
3 | * Objektorientierung in JavaScript
4 | */
5 |
6 | // Konstruktor
7 | function Buch(titel, autor, verlag, jahr) {
8 |
9 |     this.titel = titel;
10 |    this.autor = autor;
11 |    this.verlag = verlag;
12 |    this.jahr = jahr;
13 |
14 |    this.toString = toString;
15 |    this.htmlRow = htmlRow;
16 |
17 | } // Konstruktor
18 |
19 |
20 | // reine Textausgabe
21 | function toString() {
22 |     return this.autor + ": " + this.titel + " (" + this.verlag + ", " + this.jahr + ")";
23 | } // toString
24 |
25 |
26 | // HTML-Tabellenzeile
27 | function htmlRow() {
28 |     return "<tr><td>" + this.titel + "</td><td>" + this.autor +
29 |           "</td><td>" + this.verlag + "</td><td>" + this.jahr + "</td></tr>";
30 | } // htmlRow
    
```





...und nun...

- haben wir das Prinzip der clientseitigen
Scriptsprache JavaScript kennen gelernt

- als nächstes:

JavaScript und DOM:

**Die Verbindung von
JavaScript zu HTML**

