

Thema: Übersicht und Vergleich von Programmiersprachen in der Webentwicklung

"Das Web ist das Betriebssystem der Welt." - Marc Benioff, CEO von Salesforce

1. HTML5 & CSS3 | Markup-Sprachen | Veröffentlicht 1992 - 1994¹ | Entwickelt von W3C²

HTML (Hypertext Markup Language) und CSS (Cascading Style Sheets Beschreibung) bilden das Grundgerüst und die Gestaltungselemente für Webseiten. HTML strukturiert den Inhalt, während CSS das visuelle Design, Farben, Schriftarten und Layouts definiert.

```
p.info {
  font-family: arial, sans-serif;
  line-height: 150%;
  margin-left: 2em;
  padding: 1em;
  border: 3px solid #f8d7da;
  background-color: #f8d7da;
  display: inline-block;
}
```

- Einsatzszenario: Erstellung einer eigenen Webseite
- Erfolgreiche Projekte: Wikipedia, Google, Facebook, Amazon, Netflix, Twitter
- Schlüsselmerkmale: Klare und leicht verständliche Syntax, verwendet Tags, plattformunabhängig

Vorzüge	Herausforderungen
<ul style="list-style-type: none">Einfache Strukturierung von InhaltenKlare Trennung von Inhalt und DarstellungUnterstützung für Responsive Design	<ul style="list-style-type: none">Begrenzte MöglichkeitenBrowserkompatibilitätAbhängigkeit von anderen Technologien

2. JavaScript | Multiparadigmatische Sprache | Veröffentlicht 1999³ | Entwickelt von Brendan Eich⁴

"JavaScript ist die Sprache der Webentwicklung."
- Douglas Crockford, Mitentwickler von JavaScript.
JavaScript ermöglicht Webseiten interaktiv zu gestalten, Benutzeraktionen zu verarbeiten und dynamische Inhalte zu erstellen.

```
function prüfeZahl(zahl) {
  if (zahl > 0) {
    console.log("Die Zahl ist positiv.");
  } else if (zahl < 0) {
    console.log("Die Zahl ist negativ.");
  } else {
    console.log("Die Zahl ist null.");
  }
}
```

- Einsatzszenario: Formulare, Animationen, Videoplayer, interaktive Bildergalerie, Web-Server
- Schlüsselmerkmale: Klare und leicht verständliche Syntax, Unterstützung für objektorientierte, clientseitige Skriptsprache und plattformunabhängig
- Erfolgreiche Projekte: Google Maps, Facebook, Airbnb, Twitter, Spotify, LinkedIn, Uber

Vorzüge	Herausforderungen
<ul style="list-style-type: none">Vielseitige AnwendungDynamische Aktualisierung von InhaltenUmfangreiche Community und Ressourcen	<ul style="list-style-type: none">BrowserkompatibilitätCross-Site Scripting (XSS)Komplexität asynchroner Operationen

¹ Wenz, Christian / Prevezanos, Christoph: Jetzt lerne ich HTML5 und CSS3, Paderborn 2018, S.19.
² Vgl. Ebd., S.21.
³ Crockford, Douglas: JavaScript: The Definitive Guide, 1999, S.265.
⁴ Vgl. Ebd., S.265.

3. Python | Multiparadigmatische Sprache | Veröffentlicht 1994⁵ | Entwickelt von Guido van Rossum⁶

Python, eine vielseitige Programmiersprache, findet breite Anwendung von Webentwicklung bis hin zur Datenanalyse. Ihre klare und lesbare Syntax ermöglicht effiziente Programmierung, während sie sich für komplexe Aufgaben wie maschinelles Lernen und wissenschaftliche Berechnungen eignet.

```
for i in words:
    if len(i) >= 3:
        i = i + "%say" % (i[0])
        i = i[1:]
        print i
    else:
        pass
```

Einsatzszenario: Chatbot, maschinelles Lernen, Automatisierung von Aufgaben

Schlüsselmerkmale: Betont Lesbarkeit, reduzierter Programmieraufwand, Unterstützung für objektorientierte, imperative und funktionale Programmierung und plattformunabhängig, Umfangreiche Bibliotheken

Erfolgreiche Projekte: Instagram, YouTube, Dropbox, Spotify, NASA, Reddit, Netflix

Vorzüge	Herausforderungen
<ul style="list-style-type: none">• Vielseitig einsetzbar• Klare und lesbare Syntax• Umfangreiche Standardbibliothek	<ul style="list-style-type: none">• Nicht für das Frontend geeignet• Skalierbarkeit bei großen Anwendungen• Probleme mit Version Kompatibilität

4. PHP | Prozedurale Sprache | Veröffentlicht 1995⁷ | Entwickelt von Rasmus Lerdorf⁸

PHP (Hypertext Preprocessor), eine serverseitige Skriptsprache, ist maßgeblich in der Webentwicklung präsent. Es ermöglicht die dynamische Generierung von Webseiten und Interaktionen mit Datenbanken. Durch seine einfache Integration in HTML ist PHP besonders für die Erstellung von dynamischen Webseiten und Anwendungen geeignet.

```
$result = $client->completions()->create([
    'model' => 'davinci',
    'prompt' => 'PHP is',
]);
dd($result['choices'][0]['text']);
```

Einsatzszenario: Kontaktformular, APIs, Content-Management-Systeme, Webshops

Schlüsselmerkmale: Serverseitige Skriptsprache, plattformübergreifend und integrierte Unterstützung für Datenbanken

Erfolgreiche Projekte: WordPress, Facebook, Wikipedia, Magento, Drupal, Joomla

Vorzüge	Herausforderungen
<ul style="list-style-type: none">• Schnell für dynamische Webseiten• Breite Unterstützung und Verfügbarkeit in Hosting-Umgebungen• Gute Integration mit Datenbanken	<ul style="list-style-type: none">• Syntax weniger elegant• Aufgrund seiner Flexibilität kann PHP anfällig für unsicheren Code sein• Fragmentierung durch verschiedene PHP-Versionen

⁵ Vgl. Rheinwerk Verlag: Die Programmiersprache Python, online verfügbar unter: https://openbook.rheinwerk-verlag.de/python/02_001.html, 2024, zugegriffen am 26.12.2023.

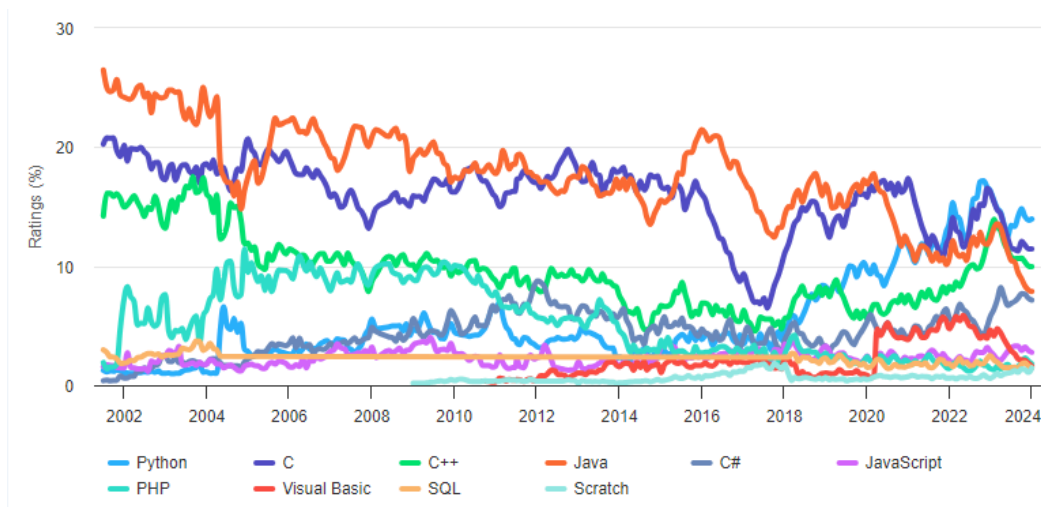
⁶ Vgl. Wikipedia: Guido von Rossum, online verfügbar unter: https://de.wikipedia.org/wiki/Guido_van_Rossum, 2023, zugegriffen am 26.12.2023.

⁷ Vgl. Offizielle PHP Webseite: Die Geschichte von PHP, online verfügbar unter: <https://www.php.net/manual/de/history.php.php>, 2024, zugegriffen am 28.12.2023.

⁸ Vgl. Wikipedia: Rasmus Lerdorf, online verfügbar unter: https://de.wikipedia.org/wiki/Rasmus_Lerdorf, 2023, zugegriffen am 08.01.2024.

5. Fazit

Abschließend zeigt sich: HTML und CSS bilden das Grundgerüst, JavaScript bringt Interaktivität, Python steht für Vielseitigkeit, und PHP macht das Web dynamisch. Jede Sprache hat ihren Platz und, wie folgendes Diagramm zeigt, große Beliebtheit in der Softwareentwickler Community.



TIOBE-Index: TIOBE Programming Community Index, online verfügbar unter: <https://www.tiobe.com/tiobe-index/>, Stand Januar 2024, zugegriffen am 02.02.2024.

6. Quellenverzeichnis

6.1. Literaturquellen:

- Crockford, Douglas: JavaScript: The Definitive Guide, 1999.
- Wenz, Christian / Prevezanos, Christoph: Jetzt lerne ich HTML5 und CSS3, Paderborn 2018.

6.2. Internetquellen:

- Rheinwerk Verlag: Die Programmiersprache Python, online verfügbar unter: https://openbook.rheinwerk-verlag.de/python/02_001.html, 2024, zugegriffen am 26.12.2023.
- Wikipedia: Guido von Rossum, online verfügbar unter: https://de.wikipedia.org/wiki/Guido_van_Rossum, Stand November 2023, zugegriffen am 26.12.2023.
- Offizielle PHP Webseite: Die Geschichte von PHP, online verfügbar unter: <https://www.php.net/manual/de/history.php.php>, Stand 2024, zugegriffen am 28.12.2023.
- Wikipedia: Rasmus Lerdorf, online verfügbar unter: https://de.wikipedia.org/wiki/Rasmus_Lerdorf, Stand Januar 2023, zugegriffen am 08.01.2024.
- Die Angaben zu "Erfolgreiche Projekte" wurden mithilfe von Stack Share erhoben, unter <https://stackshare.io/stacks>, Stand 2024, zugegriffen am 03.02.2024.
- TIOBE: TIOBE Index for January 2024, online verfügbar unter: <https://www.tiobe.com/tiobe-index/>, Stand Januar 2023, zugegriffen am 08.01.2024.

7. Abbildungsverzeichnis:

- Alle Abbildungen wurden von Silas Noah Stöhr erstellt, mit dem Programm: Visual Studio Code.
- Das Diagramm: "Die beliebtesten Programmiersprachen weltweit" wurde von Silas Noah Stöhr erstellt, mit dem Programm: Excel.

Auch die heute bekannten *Cascading Style Sheets (CSS)* haben sich erst im Laufe der Zeit entwickelt. Am Anfang gab es keine Versionen, und es wurde schlicht von CSS gesprochen. Erst mit der strikteren Trennung von Inhalten und Layout durch HTML war man auch bei CSS gezwungen, klare Standards zu schaffen.

- Die erste konkrete Definition von CSS 1.0 erfolgte im Jahr 1996. Ein Jahr später, also 1997, wurde sie offiziell in HTML 4.0 aufgenommen.
- Etwa zwei Jahre später, Mitte 1998, erschien die Version CSS 2.0 und erweiterte die Möglichkeiten der Formatierung und Gestaltung um viele neue Befehle. Vor allem sollten die Kompatibilitätsprobleme der verschiedenen Browser durch strikte Standards gelöst werden.
- Seit 2007 ist die Version CSS 2.1 der neue Standard und beinhaltet vor allem kleinere Änderungen, Korrekturen und Optimierungen im Befehlssatz. Außerdem werden die neuen Möglichkeiten der aktuellen Browser besser berücksichtigt.
- An der Version CSS 3.0 wird bereits seit einigen Jahren gearbeitet. Sie soll viele neue Möglichkeiten bringen, modulares Arbeiten ermöglichen und zwischen Anzeigegeräten wie PC, Notebook, Tablet-PC, Handy usw. unterscheiden können. Allerdings wurde bis heute nie eine offizielle Version 3.0 verabschiedet. Stattdessen denkt das W3C darüber nach, CSS ganz ohne Versionsnummer zu entwickeln und dem jeweils aktuellen HTML anzupassen.

1.2.2 Der Umstieg von HTML 4 zu HTML5

Bei neuen Programmen und Standards ist immer einiges an Arbeit notwendig, um sich mit den aktuellen Entwicklungen vertraut zu machen. Das ist bei HTML nicht anders, aber der Umstieg von HTML 4 auf HTML5 ist doch eine Besonderheit. Mit der neuen Version geht es nicht nur um ein paar Optimierungen und Korrekturen, sondern um sehr grundsätzliche Funktionen und Prinzipien der Sprache.

- HTML5 bringt einen großen Schwung ganz neuer Befehle mit sich. Sie erweitern die Möglichkeiten bei der Webseitengestaltung teilweise erheblich.
- Viele bestehende Befehle wurden ganz neu definiert. So funktioniert z. B. das Einbinden von Multimedia-Inhalten wie Video und Musik völlig anders.
- Manche Befehle wurden ersatzlos gestrichen. Einige bisher bekannte Funktionen lassen sich einfach nicht mehr durchführen.
- Die Trennung von Inhalten und Layout erfährt mit HTML5 einen großen Sprung. HTML selbst besitzt praktisch keine Layoutbefehle mehr, und es muss zwingend CSS zum Einsatz kommen.
- HTML, XHTML und CSS rücken immer näher zusammen. Im Grunde vereint HTML5 alle drei Entwicklungen unter einem Dach. Man kann also nicht mehr einfach nur HTML lernen, sondern muss sich auch mit den anderen Themen beschäftigen.

Mit der Zeit wurden die Dokumente immer komplexer und der Code immer unüberschaubarer. Die Kombination von Inhalt und Formatierung führte zu großen Problemen und fast jeder Browser interpretierte die Anweisungen unterschiedlich. Auf verschiedenen Computern konnte ein HTML-Dokument also anders aussehen. Mit der Version HTML 4.0 kehrte man im Grunde zu den Anfängen zurück und entfernte einen Großteil der Formatierungen aus dem HTML-Code. Es fand wieder eine Trennung zwischen den Inhalten und der Darstellung statt. So enthielt eine HTML-Datei jetzt nur noch den Inhalt und die Formatierungen wurden in eine externe Datei ausgelagert. Das Prinzip der *Cascading Style Sheets (CSS)* war erfunden. Diese Trennung von Inhalten und Formatierungen stellt bis heute eine besonders wichtige Entwicklung dar, und sie ist immer noch nicht vollständig abgeschlossen. Nur so lassen sich komplexe Webseiten erstellen und verwalten, weil ihre Formatierung und das Layout nicht mehr einzeln per Hand gepflegt werden müssen.

Inzwischen sind wir bei HTML5 angekommen, und diese Version stellt mit ihren vielen grundlegenden Veränderungen einen sehr großen Sprung dar. HTML ist eben eine überaus lebendige und dynamische Sprache. Trotzdem werden Sie alles, was Sie heute über HTML lernen, auch in Zukunft gebrauchen können. Selbst wenn viele neue Befehle hinzukommen oder bestehende wieder entfallen, so bleibt das Grundprinzip doch immer gleich ... zumindest solange es das World Wide Web in der heute bekannten Form gibt und dies auf den Wurzeln von HTML basiert.

- Das Grundkonzept und somit die erste Version von HTML wurde im Jahre 1989 von Tim Berners-Lee festgelegt.
- Im Jahre 1992 wurde eine verbesserte Version von HTML veröffentlicht. Sie trug noch keine Versionsnummer und diente nur der Strukturierung von Text.
- Im Jahre 1993 wurde HTML um neue Möglichkeiten wie Fettschrift, kursive Darstellung und die Einbindung von Bildern und Grafiken erweitert. Eine Versionsnummer gab es weiterhin nicht.
- Ende 1993 wurde an einer grunderneuerten Version mit dem Namen HTML+ gearbeitet. Allerdings wurde diese nie offiziell verabschiedet oder zum Standard erklärt.
- Erst Ende 1995 erschien mit HTML 2.0 eine völlig neue Version von HTML, die grundlegend überarbeitet worden war und viele neue Funktionen bot, z. B. die Dateneingabe durch den Benutzer anhand von Formularen.
- Fast ein Jahr später sollte die Version HTML 3.0 verabschiedet werden. Allerdings wurde zu diesem Zeitpunkt der Browser Netscape Navigator in der Version 3 veröffentlicht, der so viele neue Funktionen bot, dass der neue HTML-Standard bereits veraltet war. HTML 3.0 wurde deshalb nie offiziell verabschiedet.
- Aus diesem Grund folgte Anfang 1997 die Version HTML 3.2, die an die neuen Möglichkeiten der aktuellen Browser angepasst war. Es gab zahlreiche neue Funktionen wie z. B.

Until now, this book has described the complete and official JavaScript language, as standardized by ECMAScript 3 and ECMAScript 5. This chapter instead describes subsets and supersets of JavaScript. The subsets have been defined, for the most part, for security purposes: a script written using only a secure language subset can be executed safely even if it comes from an untrusted source such as an ad server. §11.1 describes a few of these subsets.

The ECMAScript 3 standard was published in 1999 and a decade elapsed before the standard was updated to ECMAScript 5 in 2009. Brendan Eich, the creator of JavaScript, continued to evolve the language during that decade (the ECMAScript specification explicitly allows language extensions) and, with the Mozilla project, released JavaScript versions 1.5, 1.6, 1.7, 1.8, and 1.8.1 in Firefox 1.0, 1.5, 2, 3, and 3.5. Some of the features of these extensions to JavaScript have been codified in ECMAScript 5, but many remain nonstandard. Future versions of ECMAScript are expected to standardize at least some of the remaining nonstandard features.

The Firefox browser supports these extensions, as does the Spidermonkey JavaScript interpreter that Firefox is based on. Mozilla's Java-based JavaScript interpreter, Rhino, (see §12.1) also supports most of the extensions. Because these language extensions are nonstandard, however, they will not be useful to web developers who require language compatibility across all browsers. They are documented in this chapter because:

- they are quite powerful;
- they may become standard in the future;
- they can be used to write Firefox extensions;
- they can be used in server-side JavaScript programming, when the underlying JavaScript engine is Spidermonkey or Rhino (see §12.1).

After a preliminary section on language subsets, the rest of this chapter describes these language extensions. Because they are nonstandard, they are documented in tutorial style with less rigor than the language features described elsewhere in the book.