

Eine Zeichencodierung wählen und anwenden

Zielgruppe: HTML-Entwickler (die Editoren oder Scripte verwenden), Script-Entwickler (PHP, JSP u.a.), CSS-Entwickler, Webprojekt-Manager und alle, die eine Anleitung suchen, wie man eine Zeichencodierung wählt und anwendet

Hinweis: Das [englische Original](#) wurde nach dieser Übersetzung aktualisiert. Die Änderungen wurden in diese Seite übernommen, so dass evtl. einige Abschnitte auf englisch erscheinen, bis die Übersetzung aktualisiert wird.

Frage

Welche Zeichencodierung sollte man für seinen Inhalt wählen und wie wendet man diese auf seinen Inhalt an?

Inhalt setzt sich aus einer Sequenz von Zeichen zusammen. Zu den Zeichen gehören die Buchstaben des Alphabets, Interpunktionszeichen usw. In einem Computer wird der Inhalt jedoch als eine Sequenz von Bytes gespeichert, das sind Zahlenwerte. Manche Zeichen werden durch mehr als ein Byte dargestellt. Wie bei Chiffren in der Spionage hängt die Art, wie Sequenzen von Bytes in Zeichen umgewandelt werden, von dem Schlüssel ab, mit dem der Text codiert wurde. Den *Schlüssel* nennt man in diesem Kontext **Zeichencodierung**.

Dieser Artikel gibt Ihnen einfache Ratschläge, welche Zeichencodierung Sie für Ihre Inhalte verwenden sollten und wie man diese anwendet, d.h. wie man ein Dokument in dieser Zeichencodierung erstellt.

Wenn Sie besser verstehen möchten, was Zeichen und Zeichencodierungen sind, lesen Sie den Artikel [Zeichencodierung für Anfänger](#).

Kurze Antwort

Verwenden Sie UTF-8 für alle Ihre Inhalte. Ziehen Sie in Erwägung, Inhalte in veralteten Zeichencodierungen nach UTF-8 zu konvertieren.

Wenn Sie keine Unicode-Codierung verwenden können, prüfen Sie, ob die von Ihnen gewählte Codierung [von verschiedenen Browsern unterstützt](#) wird und ob diese Codierung nicht auf der [Liste der zu vermeidenden Codierungen](#) steht, die den aktuellen Spezifikationen zufolge nicht verwendet werden sollten.

Prüfen Sie, ob Ihre Wahl durch serverseitige [HTTP-Einstellungen](#) überschrieben wird.

In addition to declaring the encoding of the document inside the document and/or on the server, you need to [save the text in that encoding](#) to apply it to your content.

Entwickler müssen auch sicherstellen, dass die verschiedenen Teile des Systems miteinander kommunizieren können.

Details

Die Zeichencodierung auf den Inhalt anwenden

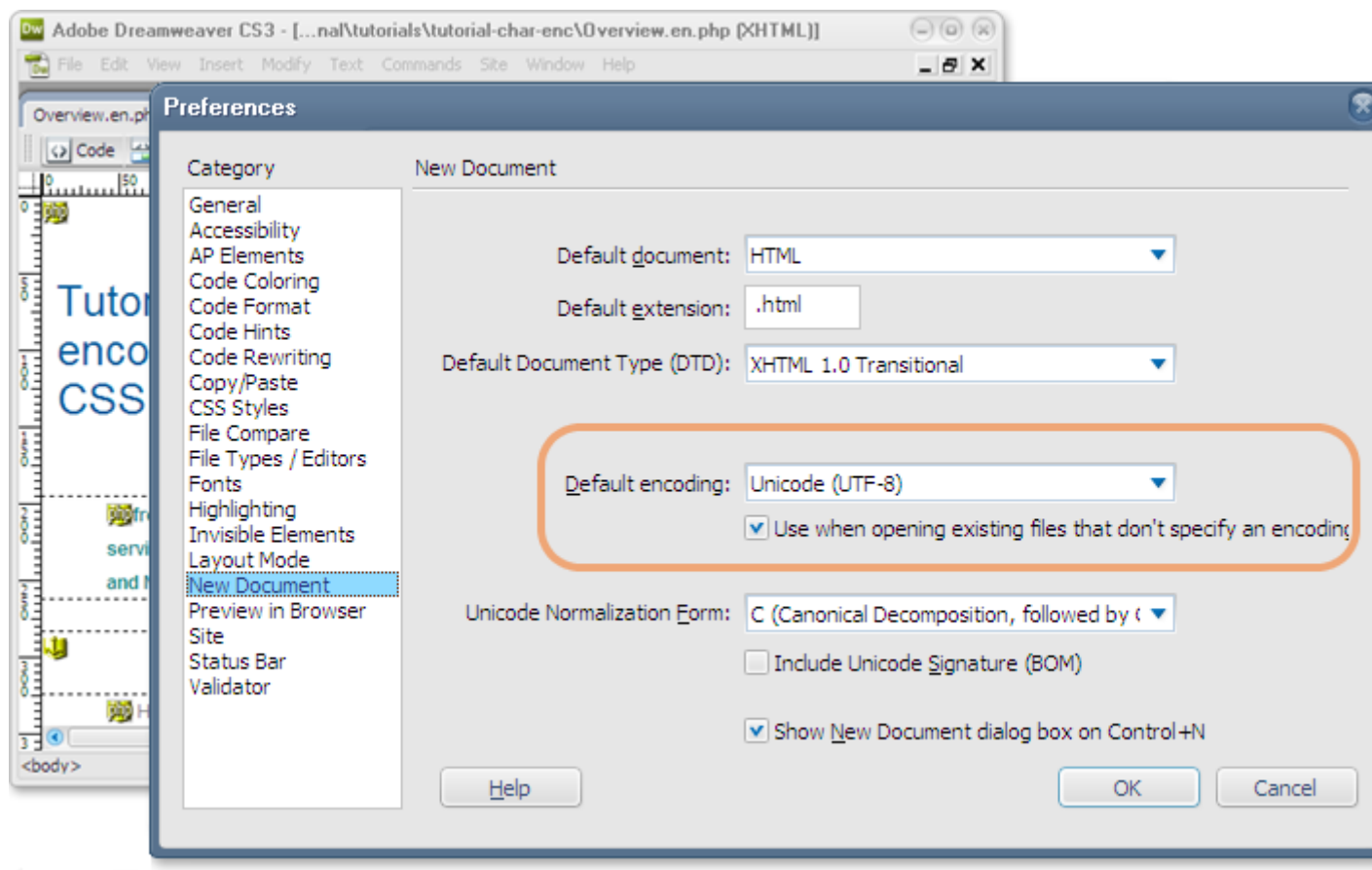
Inhaltsautoren sollten die Zeichencodierung ihrer Seiten mit einer der in [Angabe der Zeichencodierung in HTML](#) beschriebenen Methoden angeben.

Es ist aber wichtig zu verstehen, dass es nicht genügt, die Zeichencodierung im Dokument oder auf dem Server anzugeben. Das ändert nicht die Bytes; Sie müssen **den Text in dieser Zeichencodierung speichern**. (Die Angabe hilft dem Browser lediglich, die Bytesequenz zu interpretieren, in welcher der Text gespeichert ist.)

Der Artikel [Einstellung der Zeichencodierung in Web-Editoren und Texteditoren](#) gibt Ratschläge, wie man die Zeichencodierung beim Speichern für eine Reihe verschiedener Editoren einstellt.

Am besten stellt man eine Zeichencodierung wie UTF-8 in seinem Editor als Voreinstellung für neue Dokumente ein, wenn das möglich ist. Das folgende Bild zeigt, wie man das in den Einstellungen von Dreamweaver tut.

Für Informationen zu Normalisierungsformen siehe [Normalisierung in HTML und CSS](#). Für Informationen zur Unicode-Signatur (BOM) siehe [Das BOM \(byte-order mark\) in HTML](#).



Sie sollten sich auch versichern, dass Ihr Server Dokumente mit den richtigen HTTP-Angaben ausliefert, denn diese überschreiben die Angaben innerhalb des Dokuments ([siehe unten](#)).

Entwickler müssen auch sicherstellen, dass die verschiedenen Teile des Systems miteinander kommunizieren können. Webseiten müssen mit Scripten im Backend, Datenbanken usw. kommunizieren können. Das funktioniert natürlich am besten, wenn alles UTF-8-codiert ist. Was Entwickler berücksichtigen müssen finden Sie im Artikel [Migration zu Unicode](#).

Warum sollte man UTF-8 verwenden?

Eine HTML-Seite kann nur in einer Zeichencodierung codiert sein. Man kann nicht verschiedene Teile eines Dokuments in verschiedenen Zeichencodierungen codieren.

Eine Unicode-Codierung wie UTF-8 kann viele Sprachen unterstützen und Seiten und Formulare mit beliebig gemischten Sprachen in Einklang bringen. Verwendet man eine Unicode-Codierung, benötigt man keine serverseitige Logik, um für jede ausgelieferte Seite oder alle hereinkommenden Formulardaten gesondert die Zeichencodierung zu bestimmen. Das verringert den Bearbeitungsaufwand bei einer mehrsprachigen Website oder Applikation deutlich.

Eine Unicode-Codierung erlaubt es auch, viel mehr Sprachen auf einer Webseite gemischt zu verwenden als es bei jeder anderen Zeichencodierung möglich wäre.

Die Unterstützung einer bestimmten Zeichencodierung, nicht einmal einer Unicode-Codierung, heißt noch nicht unbedingt, dass ein Browser den Text richtig darstellt. Eine Reihe von Schriften, wie z.B. die arabische und indische Schriften, erfordern weitere Regeln, um eine Folge von Zeichen im Speicher in die entsprechende Folge von anzuzeigenden Schriftzeichen (Glyphen) zu überführen.

Die Hürden bei der Verwendung von Unicode sind heutzutage sehr niedrig. Im Januar 2012 gab Google bekannt, dass über 60% des Webs bei mehreren Milliarden untersuchter Webseiten jetzt UTF-8 verwenden. Zählt man die Zahl der reinen ASCII-Webseiten hinzu (ASCII ist eine Teilmenge von UTF-8), erhöht sich der Wert auf annähernd 80%.

Es gibt 3 verschiedene [Zeichencodierungen](#) für Unicode: UTF-8, UTF-16 und UTF-32. Von diesen wird nur UTF-8 zur Verwendung für Web-Inhalte empfohlen. Die HTML5-Spezifikation [sagt](#): „Autoren sollten UTF-8 verwenden. Validatoren können Autoren darauf hinweisen, keine veralteten Zeichencodierungen zu verwenden. Autorenwerkzeuge sollten UTF-8 als Voreinstellung für neue Dokumente verwenden.“

Alle ASCII-Zeichen werden in UTF-8 durch exakt dieselben Bytes codiert wie in der ASCII-Codierung, was oft hilfreich für Interoperabilität und Rückwärtskompatibilität ist.

Berücksichtigung des HTTP-Headers

Eine Angabe zur Zeichencodierung im [HTTP-Header](#) überschreibt Angaben innerhalb des Dokuments. Wenn der HTTP-Header eine Zeichencodierung angibt, die nicht der entspricht, die Sie für Ihren Inhalt verwenden möchten, stellt das ein Problem dar, wenn Sie die Servereinstellungen nicht ändern können.

Eventuell haben Sie keinen Zugriff auf die Angaben im HTTP-Header und müssen Ihre Serveradministratoren um Hilfe bitten. Andererseits können Sie aber Servereinstellungen ändern, wenn Sie beschränkten Zugang zu den Konfigurationsdateien haben oder wenn Sie Seiten mit Scriptsprachen generieren. Lesen Sie [Einstellung des HTTP-charset-Parameters](#)

für weitere Informationen zur Änderung der Angabe der Zeichencodierung für eine Anzahl von Dateien auf dem Server oder für Inhalt, der per Scriptsprache generiert wird.

Bevor Sie das tun, sollten Sie prüfen, ob der HTTP-Header eine Angabe zur Zeichencodierung enthält. Sie können den [W3C-Internationalization-Checker](#) verwenden um herauszufinden, ob eine Zeichencodierung im HTTP-Header angegeben wird und wenn ja, welche. Der Artikel [HTTP-Header überprüfen](#) verweist auf alternative Tools zur Überprüfung der Zeichencodierungsangabe des Servers.

Weitere Informationen

Dieser Abschnitt enthält Feinheiten, die Sie nicht unbedingt wissen müssen, die aber der Vollständigkeit halber hier erwähnt sind.

Was tun, wenn man nicht UTF-8 verwenden kann?

Wenn Sie es wirklich nicht vermeiden können, eine andere Codierung als UTF-8 zu verwenden, müssen Sie einen aus einer begrenzten Menge von Bezeichnern für Zeichencodierungen wählen, um maximale Interoperabilität und zukünftige Lesbarkeit Ihrer Inhalte zu gewährleisten und um Sicherheitslücken zu minimieren.

Bis vor Kurzem war das [IANA-Register](#) das Nachschlagewerk für Bezeichner von Zeichencodierungen. Das IANA-Register enthält oft mehrere Bezeichner für dieselbe Codierung. In diesen Fällen sollten Sie den als „preferred“ (bevorzugt) gekennzeichneten Bezeichner verwenden.

Die neue Spezifikation [Encoding](#) enthält eine Liste, die gegen aktuelle Browserimplementierungen getestet wurde. Sie finden Sie in der Tabelle im Abschnitt [Encodings](#). Am besten verwenden Sie die Bezeichner in der linken Spalte dieser Tabelle.

Beachten Sie: Wenn ein Bezeichner in einer dieser Quellen vorkommt, bedeutet das nicht automatisch, dass es gut wäre, diese Codierung zu verwenden. Lesen Sie im folgenden Abschnitt, welche Zeichencodierungen Sie vermeiden sollten.

Vermeiden Sie diese Zeichencodierungen

Die HTML5-Spezifikation [nennt](#) einige Zeichencodierungen, die Sie vermeiden sollten.

Dokumente dürfen nicht **JIS_C6226-1983**, **JIS_X0212-1990**, **HZ-GB-2312**, **JOHAB** (Windows code page 1361), **ISO-2022**-basierte Codierungen oder **EBCDIC**-basierte Codierungen verwenden. Der Grund ist, dass darin ASCII-Zeichencodes Nicht-ASCII-Zeichen repräsentieren, was eine Sicherheitslücke darstellt.

Dokumente dürfen auch nicht **CESU-8**, **UTF-7**, **BOCU-1** oder **SCSU**-Codierungen verwenden; diese waren nie für Web-Inhalte vorgesehen und die HTML5-Spezifikation verbietet Browsern, diese zu verwenden.

Die Spezifikation rät auch von der Verwendung von **UTF-16** ab, und von der Verwendung von **UTF-32** wird „besonders abgeraten“.

Auch andere [in der Encoding-Spezifikation aufgeführte](#) Zeichencodierungen sollten nicht verwendet werden, darunter **Big5** und **EUC-JP**, die hinsichtlich Interoperabilität problematisch sind. **ISO-8859-8** (hebräische Codierung für [visuelle Buchstabenreihenfolge](#)) sollten Sie ebenfalls nicht verwenden, sondern eine Codierung, die in [logischer Buchstabenreihenfolge](#) codiert (UTF-8; oder wenn das nicht möglich ist: ISO-8859-8-i).

Die in der *Encoding*-Spezifikation aufgeführte **replacement**-Codierung ist eigentlich keine Codierung, sondern ein Fallback, der jedes Oktett (Byte) auf den Unicode-Zeichencode U+FFFD REPLACEMENT CHARACTER abbildet. Es ist natürlich nicht sinnvoll, Daten in dieser Codierung zu übertragen.

Die **x-user-defined**-Codierung ist eine Ein-Byte-Codierung, deren untere Hälfte ASCII ist und deren obere Hälfte in den Unicode-Bereich zum privaten Gebrauch (Unicode Private Use Area, PUA) abgebildet wird. Wie der Bereich zum privaten Gebrauch im Allgemeinen sollte diese Codierung im öffentlichen Internet vermieden werden, weil sie Interoperabilität und langfristiger Verwendung abträglich ist.

Literaturhinweise

Sagen Sie uns, was Sie denken.

[Senden Sie uns Anmerkungen](#)

Folgen Sie unserem Newsfeed.



[@webi18n](#)



[RSS](#)

- Erste Schritte: [Einführung in Zeichensätze und Zeichencodierungen](#)
- Tutorial: [Umgang mit Zeichencodierungen in HTML und CSS](#)
- Verwandte Links: *HTML und CSS erstellen*
 - [Zeichen](#)
 - [Eine Zeichencodierung wählen](#)
 - [Die Zeichencodierung eines Dokuments ändern](#)
 - [Die Zeichencodierung eines Dokuments prüfen](#)