Unicode and the Web

Nathan Schneider

Special Text

- In our interactions with computers, it is often desirable to use characters other than the standard English alphabet and common punctuation
- When do we use different forms notation?
  - Other languages with slightly or completely different alphabets
  - Mathematical and scientific notation (e.g. chemical compounds)

Special Text

  - Notation particular to a specific field
  - Graphical features, such as arrows and bullets, that help us organize information
- Sometimes it’s appropriate to use graphics or special software to view and edit this text, but ideally it should be fairly easy to put special text onto a web page so that it displays correctly and can be edited, copied/pasted, displayed in different sizes and styles, and laid out properly without special software

Ways to Enter Text

- Directly associate keys on the keyboard with characters
- Use a sequence of keys (e.g. Ctrl+'+e => é)
- Represent it with other characters already on the keyboard (e.g. transliterating Egyptian Arabic with Latin characters)

Ways to Enter Text

- Use some graphical mechanism or special software to select characters (e.g. Windows Character Map)
- Scan it from some printed or digital format (e.g. Optical Character Recognition)
- Write it with a stylus: Handwriting Recognition
- Voice recognition technology

Ways to Enter Text

- Each of these methods has advantages and disadvantages
  - Scanning, handwriting, and voice recognition may be easier to use (more natural) but less reliable technologies, ESPECIALLY for “non-standard” text
  - Typing and graphical character selection may be cumbersome and time-consuming

Goal

- In order to ensure that computers will make our lives easier, in part by simplifying and enhancing our ability to communicate, we need to overcome these obstacles
- Computers need to (1) support special text and display it properly, as well as (2) provide convenient and reliable mechanisms for us to input special text

Old Implementation of Text

- Limitations of older computers/software: support of special text
  - Originally, most computers only supported what is known as the ASCII character set. (American Standard Code for Information Interchange) ASCII-I contains 128 characters: some control characters, and all the letters and punctuation that appear on standard American keyboards
  - Computers see each character as a number. Capital A, for example, is 65. A space is 32. ASCII contains a newline character, a tab character, and (oddly enough) a “beep” character

Old Implementation of Text

- – ASCII-II, ANSI (American National Standards Institute), and other character sets came about later
- – This was sufficient for writing computer programs, but not designed for personal use by people around the world

Problems

- There were many problems with attempts to use characters beyond the standard ASCII characters on American keyboards
  - Different computer and software systems used different representations for characters, making it difficult to translate between them
  - Using special fonts to display certain characters (where the computer sees A-Z, etc. but the font displays them as something else) restricts users to a particular font

- See http://wwwwbs.cs.tu-berlin.de/user/czyborra/charsets/

ISO-8859

- ISO-8859: This is a group character sets established by the International Standards Organization which implements various languages by mapping several sets of characters to a single range of numeric values. This leaves it up to the viewer (i.e. a web browser) to determine which set of characters to display
- Latin1 (West European); Latin2 (East European); Latin3 (South European); Latin4 (North European); Cyrillic; Arabic; Greek; Hebrew; Latin5 (Turkish); Latin6 (Nordic)

- It’s all online at http://www.unicode.org

The New Way

Unicode!

➢It’s all online at http://www.unicode.org

Unicode

- In order to solve this problem, experts have worked over the past 10 or so years to develop what’s known as The Unicode Standard. This seeks to standardize how the computer recognizes special characters. The most recent version is 4.0.
  - It does this by creating a unique identifier (a hexadecimal number) for each character in the system

➢It’s all online at http://www.unicode.org

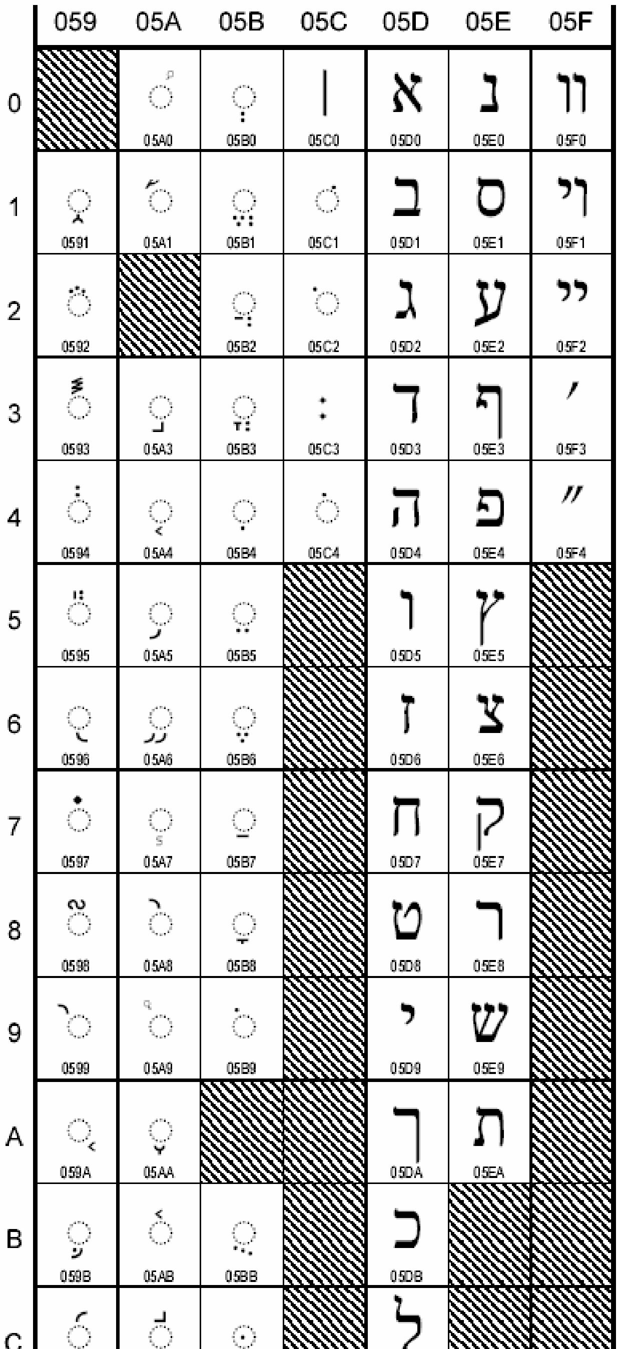
The New Way

- Unicode maps only ONE character to each numeric value, which requires more memory (if a large number of characters are to be supported), but makes things MUCH less confusing
  – Hey, memory is cheap now anyway
- It is standardized so that it should be consistent regardless of the user’s platform or system configuration

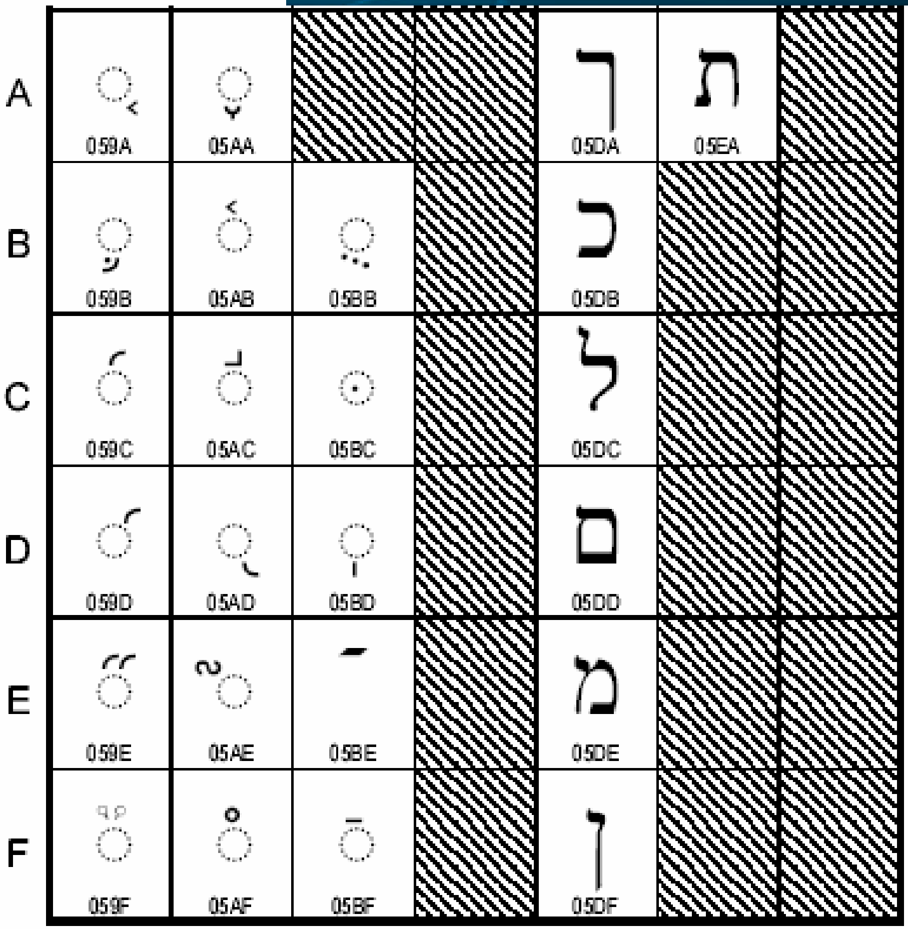
Code Points

- A Unicode code point is a hexadecimal number identifying a particular character
  - Hexadecimal is a base-16 system (as opposed to binary, or the base-10 decimal system that we normally use); hexadecimal numbers are sometimes prefixed with `0x` or x
  - In hexadecimal ("hex"), the letters A – F represent the values 10 – 15
  - `0x215C` = $12+5(16)+1(16^2)+2(16^3) = 8540$
  - 16^4-1 = 655535 possibilities (with 4 digits)

0590 Hebrew 05FF



	059	05A	05B	05C	05D	05E	05F
0	▨	◌֠ 05A0	◌ְ 05B0	׀ 05C0	א 05D0	נ 05E0	װ 05F0
1	◌֑ 0591	◌֡ 05A1	◌ֱ 05B1	◌ׁ 05C1	ב 05D1	ס 05E1	ױ 05F1
2	◌֒ 0592	▨	◌ֲ 05B2	◌ׂ 05C2	ג 05D2	ע 05E2	ײ 05F2
3	◌֓ 0593	◌֣ 05A3	◌ֳ 05B3	׃ 05C3	ד 05D3	ף 05E3	׳ 05F3
4	◌֔ 0594	◌֤ 05A4	◌ִ 05B4	◌ׄ 05C4	ה 05D4	פ 05E4	״ 05F4
5	◌֕ 0595	◌֥ 05A5	◌ֵ 05B5	▨	ו 05D5	ץ 05E5	▨
6	◌֖ 0596	◌֦ 05A6	◌ֶ 05B6	▨	ז 05D6	צ 05E6	▨
7	◌֗ 0597	◌֧ 05A7	◌ַ 05B7	▨	ח 05D7	ק 05E7	▨
8	◌֘ 0598	◌֨ 05A8	◌ָ 05B8	▨	ט 05D8	ר 05E8	▨
9	◌֙ 0599	◌֩ 05A9	◌ֹ 05B9	▨	י 05D9	ש 05E9	▨
A	◌֚ 059A	◌֪ 05AA	▨	▨	ך 05DA	ת 05EA	▨
B	◌֛ 059B	◌֫ 05AB	◌ֻ 05BB	▨	כ 05DB	▨	▨
C	◌֜	◌֬	◌ּ	▨	ל	▨	▨



A	◌֚ 059A	◌֪ 05AA	▨	▨	ך 05DA	ת 05EA	▨
B	◌֛ 059B	◌֫ 05AB	◌ֻ 05BB	▨	כ 05DB	▨	▨
C	◌֜ 059C	◌֬ 05AC	◌ּ 05BC	▨	ל 05DC	▨	▨
D	◌֝ 059D	◌֭ 05AD	◌ֽ 05BD	▨	ם 05DD	▨	▨
E	◌֞ 059E	◌֮ 05AE	־ 05BE	▨	מ 05DE	▨	▨
F	◌֟ 059F	◌֯ 05AF	◌ֿ 05BF	▨	ן 05DF	▨	▨

05A2 ▨ <reserved>

05A3 ◌֣ HEBREW ACCENT MUNAH

05A4 ◌֤ HEBREW ACCENT MAHAPAKH

05A5 ◌֥ HEBREW ACCENT MERKHA
= yored

05A6 ◌֦ HEBREW ACCENT MERKHA KEFULA

05A7 ◌֧ HEBREW ACCENT DARGA

05A8 ◌֨ HEBREW ACCENT QADMA
= azla

05A9 ◌֩ HEBREW ACCENT TELISHA QETANA

05AA ◌֪ HEBREW ACCENT YERAH BEN YOMO
= galgal

05AB ◌֫ HEBREW ACCENT OLE

05AC ◌֬ HEBREW ACCENT ILUY

05AD ◌֭ HEBREW ACCENT DEHI

05AE ◌֮ HEBREW ACCENT ZINOR
= tsinor; zarqa
- This character is to be used when Zarqa or Tsinor are placed above left.

→ 0598 ◌֘ hebrew accent zarqa

05AF ◌֯ HEBREW MARK MASORA CIRCLE

Points and punctuation

05B0 ◌ְ HEBREW POINT SHEVA

05B1 ◌ֱ HEBREW POINT HATAF SEGOL

05B2 ◌ֲ HEBREW POINT HATAF PATAH

05B3 ◌ֳ HEBREW POINT HATAF QAMATS

05B4 ◌ִ HEBREW POINT HIRIQ

05B5 ◌ֵ HEBREW POINT TSERE

05B6 ◌ֶ HEBREW POINT SEGOL

05B7 ◌ַ HEBREW POINT PATAH
- furtive patah is not a distinct character

05C4 ◌ׄ HEBREW MARK UPPER DOT

Based on ISO 8859-8

05D0 א HEBREW LETTER ALEF
= aleph
→ 2135 ℵ alef symbol

05D1 ב HEBREW LETTER BET
→ 2136 ℶ bet symbol

05D2 ג HEBREW LETTER GIMEL
→ 2137 ℷ gimel symbol

05D3 ד HEBREW LETTER DALET
→ 2138 ℸ dalet symbol

05D4 ה HEBREW LETTER HE

05D5 ו HEBREW LETTER VAV

05D6 ז HEBREW LETTER ZAYIN

05D7 ח HEBREW LETTER HET

05D8 ט HEBREW LETTER TET

05D9 י HEBREW LETTER YOD

05DA ך HEBREW LETTER FINAL KAF

05DB כ HEBREW LETTER KAF

05DC ל HEBREW LETTER LAMED

05DD ם HEBREW LETTER FINAL MEM

05DE מ HEBREW LETTER MEM

05DF ן HEBREW LETTER FINAL NUN

05E0 נ HEBREW LETTER NUN

05E1 ס HEBREW LETTER SAMEKH

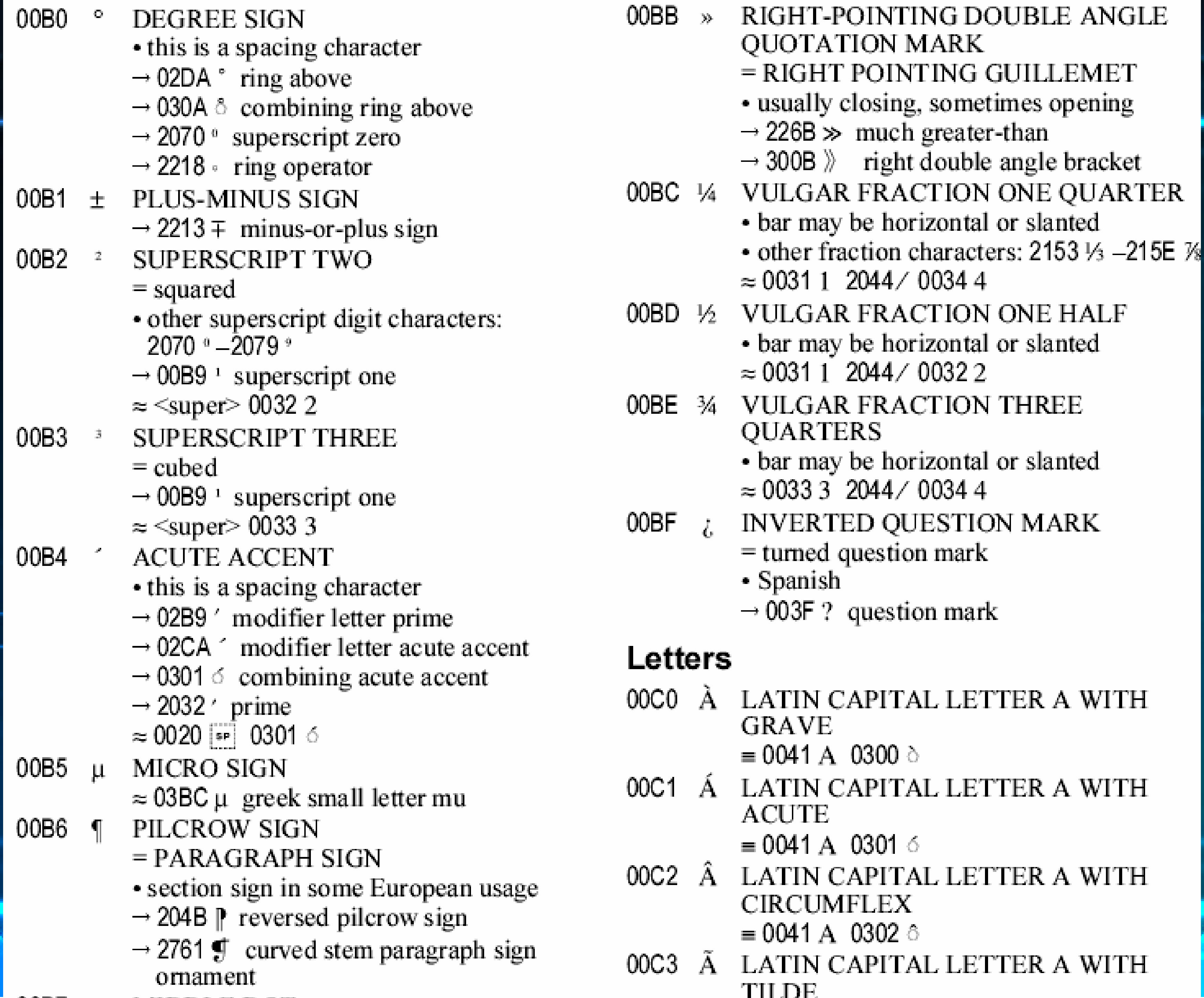
05E2 ע HEBREW LETTER AYIN

05E3 ף HEBREW LETTER FINAL PE

05E4 פ HEBREW LETTER PE

05E5 ץ HEBREW LETTER FINAL TSADI

05E6 צ HEBREW LETTER TSADI
= zade



00B0 ° DEGREE SIGN
• this is a spacing character
→ 02DA ˚ ring above
→ 030A ̊ combining ring above
→ 2070 ⁰ superscript zero
→ 2218 ∘ ring operator

00B1 ± PLUS-MINUS SIGN
→ 2213 ∓ minus-or-plus sign

00B2 ² SUPERSCRIPT TWO
= squared
• other superscript digit characters: 2070 ⁰ –2079 ⁹
→ 00B9 ¹ superscript one
≈ <super> 0032 2

00B3 ³ SUPERSCRIPT THREE
= cubed
→ 00B9 ¹ superscript one
≈ <super> 0033 3

00B4 ´ ACUTE ACCENT
• this is a spacing character
→ 02B9 ʹ modifier letter prime
→ 02CA ˊ modifier letter acute accent
→ 0301 ́ combining acute accent
→ 2032 ′ prime
≈ 0020 SP 0301 ́

00B5 µ MICRO SIGN
≈ 03BC μ greek small letter mu

00B6 ¶ PILCROW SIGN
= PARAGRAPH SIGN
• section sign in some European usage
→ 204B ⁋ reversed pilcrow sign
→ 2761 ❡ curved stem paragraph sign ornament

00BB » RIGHT-POINTING DOUBLE ANGLE QUOTATION MARK
= RIGHT POINTING GUILLEMET
• usually closing, sometimes opening
→ 226B ≫ much greater-than
→ 300B 》 right double angle bracket

00BC ¼ VULGAR FRACTION ONE QUARTER
• bar may be horizontal or slanted
• other fraction characters: 2153 ⅓ –215E ⅞
≈ 0031 1 2044 ⁄ 0034 4

00BD ½ VULGAR FRACTION ONE HALF
• bar may be horizontal or slanted
≈ 0031 1 2044 ⁄ 0032 2

00BE ¾ VULGAR FRACTION THREE QUARTERS
• bar may be horizontal or slanted
≈ 0033 3 2044 ⁄ 0034 4

00BF ¿ INVERTED QUESTION MARK
= turned question mark
• Spanish
→ 003F ? question mark

Letters

00C0 À LATIN CAPITAL LETTER A WITH GRAVE
≡ 0041 A 0300 ̀

00C1 Á LATIN CAPITAL LETTER A WITH ACUTE
≡ 0041 A 0301 ́

00C2 Â LATIN CAPITAL LETTER A WITH CIRCUMFLEX
≡ 0041 A 0302 ̂

00C3 Ã LATIN CAPITAL LETTER A WITH TILDE

Browse Unicode Character Charts

- http://www.unicode.org/charts

How the Web Works

- You type in the URL of the site you want (or click on a hyperlink)
- Your browser requests the IP address of the site with that DNS name
- Your browser sends a page request to the server
- The server generates the page (perhaps a script) and your computer downloads it
- Your browser displays the page

HTML in a Nutshell

- HTML is the standard language that browsers read to display web pages
- It stands for *Hypertext Markup Language*
- Consists primarily of tags surrounding text
  - `<b>my text goes here</b>` - bold
  - `Line1<br />Line2 blah <br />Line 3`
  - CSS (*Cascading Style Sheets*) – often used to “style” the text (fonts, colors, positioning, etc.)

➢ Click on “View Source” in your browser

Using Unicode on Web Pages

- Fortunately, HTML offers us a convenient way to represent special characters on web pages
- *HTML Entities* begin with an ampersand (&) and end with a semicolon; there are built-in *named entities*, and designers can specify Unicode characters by entering the character’s number after the # sign

Sample HTML Entities

- The five most important entities essentially “escape” the characters that have significance in HTML:
  - `&lt;` (less than) displays as <
  - `&gt;` (greater than) displays as >
  - `&amp;` displays as &
  - `&quot;` displays as "
  - `&apos;` displays as ' (for XML/XHTML only)

Sample HTML Entities

- Others include: &infin; (8), &hellip; (horizontal ellipsis, …), &copy; (©), &aacute; (à), &Euml; (Ë), &ucirc; (û), &Ccedil; (Ç), &ntilde; (ñ)
  - Note that some of these are case-sensitive
- Numbered entities for Unicode: &#8359; or &#x20A7;-k (Peseta), &#x069C;-?
- One drawback is that each font only supports a limited number of characters
  - Arial Unicode MS has broad Unicode support

Demo

- My encoder tool
  - What it does
  - Which encodings it supports
  - Symbols, X-SAMPA example
  - ISO-8859 example
  - Hebrew example
  - Written using: PHP (server-side), HTML/CSS/Javascript (client-side)
  - Show the code
  - Show the dictionary files