Character encoding:  
Character encoding is a system for representing characters using a set of binary values. Each character is assigned a unique code, which is then used to represent the character in digital form. This allows computers and other electronic devices to store, process, and transmit text data.

Some of the most common character encodings include:

• ASCII (American Standard Code for Information Interchange): ASCII is a 7-bit character encoding that represents 128 characters, including the English alphabet, numbers, and some punctuation marks. ASCII is widely used in computing and telecommunications.

• Unicode: Unicode is a 16-bit character encoding that represents over 1 million characters from a wide range of languages and scripts. Unicode is the most comprehensive character encoding and is widely used in modern operating systems and software applications.

• UTF-8: UTF-8 is a variable-length character encoding that represents Unicode characters using 1 to 4 bytes per character. UTF-8 is a popular character encoding for the internet because it is compatible with ASCII and can represent a wide range of characters.

Character encoding works by assigning a unique code to each character in a given character set. The code is typically a binary number, which is then used to represent the character in digital form. For example, in ASCII, the letter "A" is represented by the binary code 01000001.

Byte order mark (BOM):

A byte order mark (BOM) is a Unicode character that is used to indicate the byte order of a Unicode text file. The byte order of a text file refers to the order in which the bytes that represent each character are stored in the file.

There are two main byte orders:

• Big-endian: In big-endian byte order, the most significant byte of each character is stored first, followed by the least significant byte.

• Little-endian: In little-endian byte order, the least significant byte of each character is stored first, followed by the most significant byte.

UTF-8 is a variable-length character encoding that represents Unicode characters using 1 to 4 bytes per character. UTF-8 is a popular character encoding for the internet because it is compatible with ASCII and can represent a wide range of characters.

The main differences between UTF-8 with and without a BOM are byte order, without BOM it’s normally big-endian for UTF-8; And BOM serves as a sign that Unicode is being used to encode text – it’s faster to check for BOM than to check every file with UTF-8, and it’s a bit more reliable. Without BOM it would just be trying to read the files using UTF-8 until it fails. Raw ASCII (7 bit) gets interpreted as UTF-8.

ASCII art is a form of digital art that uses only the characters found on a standard ASCII keyboard. ASCII art is often used to create simple images, logos, and text-based animations.

Here is an example of ASCII art:

:-)

This ASCII art represents a simple smiley face.

ASCII art can be created using a variety of software programs, including text editors, image editors, and dedicated ASCII art generators.

Here’s an example representing a face of a TWITCH streamer. Color is inverted for better visibility:

⠀⠀⠀⡯⡯⡾⠝⠘⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⠀⢊⠘⡮⣣⠪⠢⡑⡌

⠀⠀⠀⠟⠝⠈⠀⠀⠀⠡⠀⠠⢈⠠⢐⢠⢂⢔⣐⢄⡂⢔⠀⡁⢉⠸⢨⢑⠕⡌

⠀⠀⡀⠁⠀⠀⠀⡀⢂⠡⠈⡔⣕⢮⣳⢯⣿⣻⣟⣯⣯⢷⣫⣆⡂⠀⠀⢐⠑⡌

⢀⠠⠐⠈⠀⢀⢂⠢⡂⠕⡁⣝⢮⣳⢽⡽⣾⣻⣿⣯⡯⣟⣞⢾⢜⢆⠀⡀⠀⠪

⣬⠂⠀⠀⢀⢂⢪⠨⢂⠥⣺⡪⣗⢗⣽⢽⡯⣿⣽⣷⢿⡽⡾⡽⣝⢎⠀⠀⠀⢡

⣿⠀⠀⠀⢂⠢⢂⢥⢱⡹⣪⢞⡵⣻⡪⡯⡯⣟⡾⣿⣻⡽⣯⡻⣪⠧⠑⠀⠁⢐

⣿⠀⠀⠀⠢⢑⠠⠑⠕⡝⡎⡗⡝⡎⣞⢽⡹⣕⢯⢻⠹⡹⢚⠝⡷⡽⡨⠀⠀⢔

⣿⡯⠀⢈⠈⢄⠂⠂⠐⠀⠌⠠⢑⠱⡱⡱⡑⢔⠁⠀⡀⠐⠐⠐⡡⡹⣪⠀⠀⢘

⣿⣽⠀⡀⡊⠀⠐⠨⠈⡁⠂⢈⠠⡱⡽⣷⡑⠁⠠⠑⠀⢉⢇⣤⢘⣪⢽⠀⢌⢎

⣿⢾⠀⢌⠌⠀⡁⠢⠂⠐⡀⠀⢀⢳⢽⣽⡺⣨⢄⣑⢉⢃⢭⡲⣕⡭⣹⠠⢐⢗

⣿⡗⠀⠢⠡⡱⡸⣔⢵⢱⢸⠈⠀⡪⣳⣳⢹⢜⡵⣱⢱⡱⣳⡹⣵⣻⢔⢅⢬⡷

⣷⡇⡂⠡⡑⢕⢕⠕⡑⠡⢂⢊⢐⢕⡝⡮⡧⡳⣝⢴⡐⣁⠃⡫⡒⣕⢏⡮⣷⡟

⣷⣻⣅⠑⢌⠢⠁⢐⠠⠑⡐⠐⠌⡪⠮⡫⠪⡪⡪⣺⢸⠰⠡⠠⠐⢱⠨⡪⡪⡰

⣯⢷⣟⣇⡂⡂⡌⡀⠀⠁⡂⠅⠂⠀⡑⡄⢇⠇⢝⡨⡠⡁⢐⠠⢀⢪⡐⡜⡪⡊

⣿⢽⡾⢹⡄⠕⡅⢇⠂⠑⣴⡬⣬⣬⣆⢮⣦⣷⣵⣷⡗⢃⢮⠱⡸⢰⢱⢸⢨⢌

⣯⢯⣟⠸⣳⡅⠜⠔⡌⡐⠈⠻⠟⣿⢿⣿⣿⠿⡻⣃⠢⣱⡳⡱⡩⢢⠣⡃⠢⠁

⡯⣟⣞⡇⡿⣽⡪⡘⡰⠨⢐⢀⠢⢢⢄⢤⣰⠼⡾⢕⢕⡵⣝⠎⢌⢪⠪⡘⡌⠀

⡯⣳⠯⠚⢊⠡⡂⢂⠨⠊⠔⡑⠬⡸⣘⢬⢪⣪⡺⡼⣕⢯⢞⢕⢝⠎⢻⢼⣀⠀

⠁⡂⠔⡁⡢⠣⢀⠢⠀⠅⠱⡐⡱⡘⡔⡕⡕⣲⡹⣎⡮⡏⡑⢜⢼⡱⢩⣗⣯⣟

⢀⢂⢑⠀⡂⡃⠅⠊⢄⢑⠠⠑⢕⢕⢝⢮⢺⢕⢟⢮⢊⢢⢱⢄⠃⣇⣞⢞⣞⢾

⢀⠢⡑⡀⢂⢊⠠⠁⡂⡐⠀⠅⡈⠪⠪⠪⠣⠫⠑⡁⢔⠕⣜⣜⢦⡰⡎⡯⡾⡽

Here is the original photo:  


HTML Character Entities

HTML character entities are special codes used to represent characters that cannot be directly typed in HTML documents. They are used to insert special characters, such as accents, symbols, and mathematical symbols, into a webpage.

Character entities are written in the following format:

&character-name;  
For example - &lt; and &rt; representing < (less-than sign) and > (greater-than sign).

In addition to named character entities, Unicode character entities can also be used. They use the following syntax:

&#decimal-code;

For example, the Unicode entity for the copyright symbol is:

&#xA9;

This allows for the insertion of special characters that you can not type in directly in html, for example due to possible code errors with < and >. It improves readability.

However, it may not be supported by all platforms.

<pre> and <code>  
<pre> and <code> are container tags that are used to display text the way it was meant – with every space and line break that is included, since HTML normally ignores multiple spaces and line breaks. You can use both at the same time to display code samples on a website – this is how most tutorial websites or blogs on programming show you examples.