# ASCII Table

Computers operate using numbers and therefore there needs to be a way for a computer to convert letters (and other "characters") to and from numbers. A set of codes, known as "ASCII" (American Standard Code for Information Interchange) are used. These were initially developed for tasks such as sending documents to printers, and many of the commands make sense in this context.

Notes:

(a) ASCII is a 7-bit code, representing 128 different characters. When an ascii character is stored in a byte the most significant bit is always zero. Sometimes the extra bit is used to indicate that the byte is not an ASCII character, but is a graphics symbol, however this is not defined by ASCII.

(b) To convert a hexadecimal number using the table, take the most significant 4 bits (row) followed by the least significant 4 bits (column); e.g. 0x33 means 00110011, which is the code for the character **3.**

(b) *Some simple rules*: the decimal digits 0 - 9 are represented by the codes 30 - 39. The upper case letters run from 41 to 5A; the corresponding lower case letters run from 61 to 7A; the two codes are identical except for one bit (e.g. C is 43 and c is 63; in binary C is 1000011 and c is 1100011; the only difference is bit 5.

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 00 0000 0000<br>NUL<br>□ | 01 0000 0001<br>SOH<br>┌ | 02 0000 0010<br>STX<br>⊥ | 03 0000 0011<br>ETX<br>┘ | 04 0000 0100<br>EOT<br>↯ | 05 0000 0101<br>ENQ<br>⊠ | 06 0000 0110<br>ACK<br>✓ | 07 0000 0111<br>BEL<br>⍾ | 08 0000 1000<br>BS<br>↶ | 09 0000 1001<br>HT<br>⪢ | 10 0000 1010<br>LF<br>≡ | 11 0000 1011<br>VT<br>V | 12 0000 1100<br>FF<br>↡ | 13 0000 1101<br>CR<br>< | 14 0000 1110<br>SO<br>⊗ | 15 0000 1111<br>SI<br>⊙ | 8 |
| 1 | 16 0001 0000<br>DLE<br>⊟ | 17 0001 0001<br>DC1<br>◷ | 18 0001 0010<br>DC2<br>◶ | 19 0001 0011<br>DC3<br>◵ | 20 0001 0100<br>DC4<br>◴ | 21 0001 0101<br>NAK<br>⍻ | 22 0001 0110<br>SYN<br>⎍ | 23 0001 0111<br>ETB<br>⊣ | 24 0001 1000<br>CAN<br>⧗ | 25 0001 1001<br>EM<br>⍿ | 26 0001 1010<br>SUB<br>ʕ | 27 0001 1011<br>ESC<br>⊖ | 28 0001 1100<br>FS<br>◰ | 29 0001 1101<br>GS<br>◱ | 30 0001 1110<br>RS<br>◲ | 31 0001 1111<br>US<br>◳ | 9 |
| 2 | 32 0010 0000<br>SP | 33 0010 0001<br>! | 34 0010 0010<br>" | 35 0010 0011<br># | 36 0010 0100<br>$ | 37 0010 0101<br>% | 38 0010 0110<br>& | 39 0010 0111<br>′ | 40 0010 1000<br>( | 41 0010 1001<br>) | 42 0010 1010<br>* | 43 0010 1011<br>+ | 44 0010 1100<br>, | 45 0010 1101<br>– | 46 0010 1110<br>. | 47 0010 1111<br>/ | A |
| 3 | 48 0011 0000<br>0 | 49 0011 0001<br>1 | 50 0011 0010<br>2 | 51 0011 0011<br>3 | 52 0011 0100<br>4 | 53 0011 0101<br>5 | 54 0011 0110<br>6 | 55 0011 0111<br>7 | 56 0011 1000<br>8 | 57 0011 1001<br>9 | 58 0011 1010<br>: | 59 0011 1011<br>; | 60 0011 1100<br>< | 61 0011 1101<br>= | 62 0011 1110<br>> | 63 0011 1111<br>? | B |
| 4 | 64 0100 0000<br>@ | 65 0100 0001<br>A | 66 0100 0010<br>B | 67 0100 0011<br>C | 68 0100 0100<br>D | 69 0100 0101<br>E | 70 0100 0110<br>F | 71 0100 0111<br>G | 72 0100 1000<br>H | 73 0100 1001<br>I | 74 0100 1010<br>J | 75 0100 1011<br>K | 76 0100 1100<br>L | 77 0100 1101<br>M | 78 0100 1110<br>N | 79 0100 1111<br>O | C |
| 5 | 80 0101 0000<br>P | 81 0101 0001<br>Q | 82 0101 0010<br>R | 83 0101 0011<br>S | 84 0101 0100<br>T | 85 0101 0101<br>U | 86 0101 0110<br>V | 87 0101 0111<br>W | 88 0101 1000<br>X | 89 0101 1001<br>Y | 90 0101 1010<br>Z | 91 0101 1011<br>[ | 92 0101 1100<br>\ | 93 0101 1101<br>] | 94 0101 1110<br>^ | 95 0101 1111<br>_ | D |
| 6 | 96 0110 0000<br>` | 97 0110 0001<br>a | 98 0110 0010<br>b | 99 0110 0011<br>c | 100 0110 0100<br>d | 101 0110 0101<br>e | 102 0110 0110<br>f | 103 0110 0111<br>g | 104 0110 1000<br>h | 105 0110 1001<br>i | 106 0110 1010<br>j | 107 0110 1011<br>k | 108 0110 1100<br>l | 109 0110 1101<br>m | 110 0110 1110<br>n | 111 0110 1111<br>o | E |
| 7 | 112 0111 0000<br>p | 113 0111 0001<br>q | 114 0111 0010<br>r | 115 0111 0011<br>s | 116 0111 0100<br>t | 117 0111 0101<br>u | 118 0111 0110<br>v | 119 0111 0111<br>w | 120 0111 1000<br>x | 121 0111 1001<br>y | 122 0111 1010<br>z | 123 0111 1011<br>{ | 124 0111 1100<br>\| | 125 0111 1101<br>} | 126 0111 1110<br>~ | 127 0111 1111<br>▨<br>DEL | F |

(c) Many of the codes are not *printing* characters at all; these are the codes 00 to 1F, and FF, which are represented by groups of letters (NUL, DEL). Some are frequently used in text; for example LF (line feed) which is 0x0A (which causes a printer/display to move down one line), and CR (carriage return) which is 0x0D (which often causes a printer/display to move down one line and to the left hand side). There is also SP (space) which is 0x20; since this corresponds to an actual blank in the text it might be regarded as printing. NUL (null) has a value of zero and causes a printer/display to ignore the character. Others characters were once used to give information about messages, for example STX (start of text, 0x02,) and ETX (end of text, 0x03).

(d) Computers often have a need to store groups of chacaters (forming words or sentences). A group of chacaters is usually called a "string". In high level languages such as 'C', the end of a "string" is indicated by using a NUL character (0x00). Since this character is never actually displayed, it is safe to assume that the character will never be one of the characters in a string.