

# Bytes

A bit is the smallest piece of information in a computer, a single value storing either 0 or 1.

A **byte** is a unit of digital information that consists of **8** of those bits.

Here's a single byte of information:

11110110

Here are three more bytes of information:

000010100101010011011011

## From bits to bytes

Conversion between bits and bytes is a simple calculation: divide by 8 to convert from bits to bytes or multiply by 8 to convert from bytes to bits. Try it yourself!

### CHECK YOUR UNDERSTANDING

How many bytes long is this binary sequence?

1011100111011011011010010011010101111111

**Answer:** Let's count the number of 8-bit groups in the binary sequence, by adding a space after every 8 bits:

1011100111011011011010010011010101111111

There are 5 groups of 8 in the sequence, so it is 5 bytes long.

### CHECK YOUR UNDERSTANDING

How many bits are in 8 bytes of information?

**Answer:** There are 8 bits in a byte.

To figure out how many bits are in 8 bytes, we can just multiply:

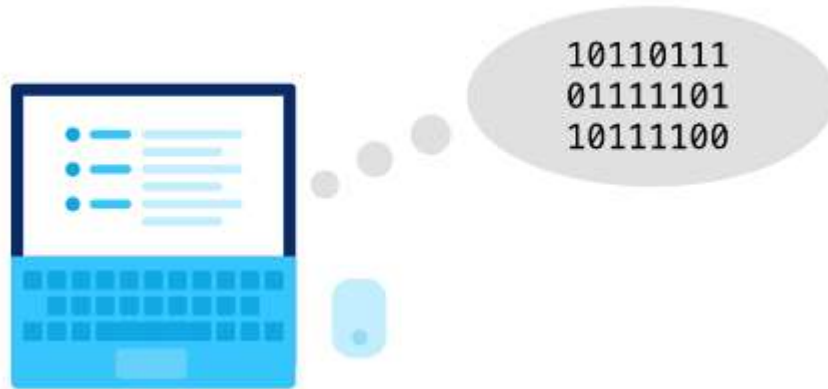
$8 \times 8 = 64$ .

Thus, there are 64 bits in 8 bytes.

## Why bytes?

What is so special about 8 bits that it deserves its own name?

Computers do process all data as bits, but they prefer to process bits in byte-sized groupings. Or to put it another way: a byte is how much a computer likes to "bite" at once.



The byte is also the smallest addressable unit of memory in most modern computers. A computer with byte-addressable memory can *not* store an individual piece of data that is smaller than a byte.

Memory	
Address	Data
1	11110000
2	00001010
3	00100110
4	11000001

## What's in a byte?

A byte represents different types of information depending on the context. It might represent a number, a letter, or a program instruction. It might even represent part of an audio recording or a pixel in an image.

We'll explore how computers can use bits and bytes to represent all types of information in this unit.