JASON ALAN SMITH • ANVILABS

NOTE

What is a Data Type?

October 24, 2026 • First Principles

data-types memory bits first-principles

Introduction

A *data type* classifies a datum stored in a computer's memory. But what exactly does that mean? Why is classification useful?

> **DEFINITION**
>
> A **data type** classifies a datum in memory, giving meaning and structure to an undifferentiated sequence of bytes.

> **DEFINITION**
>
> A **datum** is a single piece of data—the singular of *data*.

Information / Data

Before we explore what this means and why data types are useful, we must first understand how computers encode information or data. A computer operates on sequences of 0s and 1s. That's it. These 0s and 1s manifest as electrical signals inside the hardware. Every 0 and 1 in a computer is a binary digit, or *bit*. Computers encode information into sequences of bits. A *byte* is a sequence of eight bits.

Memory

A computer's memory is arranged as a long row of boxes—an array—where each box is called an element of the array. Each element contains exactly one byte.

Interpretation

Computers are relatively simple machines that execute sets of precise instructions called programs. On its own, a computer does nothing useful. A program loads into memory as a sequence of bytes, bringing the computer to life. The computer itself doesn't interpret data types; rather, programs perform the interpretation.

A program contains instructions that read data from and write data to memory. Programs interpret data types from sequences of bytes stored in memory. So how does a program

Copyright © 2026 Jason Alan Smith

interpret a sequence of bytes as the word *cat*, for example?

This answer shows why data types are useful: they give meaning and structure to an undifferentiated sequence of bytes in memory. For a program to read from or write to memory, it must know where in memory the datum resides or will reside—the starting position.

Programs use the datum's data type to determine the span of bytes, starting at the starting position, that it can read from and write to in memory.

A datum can be a sequence of characters, a number with or without a decimal point, or a boolean value that represents true or false conditions. The data type defines the size—how many bytes—that the datum consumes.

Given a starting position in memory, the size determines the span of bytes that the program will read or write, and this span includes the starting position. But a data type is much richer than this simple calculation.

Operations

For example, a datum's data type also determines what actions, or operations, the program can perform on this value. Programs can add, subtract, multiply, and divide *numeric* data but not string data. Programs can also increment or decrement numbers. *String* data, however, can be concatenated while numeric data cannot. These operations represent a small fraction of the total operations programs have at their disposal.

Summary

So why are data types useful? Data types are essential tools that computer programs use to manipulate bytes of data in memory. At their core, data types enable programs to read and write data in memory. However, data types go further and designate allowed operations on specific classifications of data.

Copyright © 2026 Jason Alan Smith