

What Are Data Types?

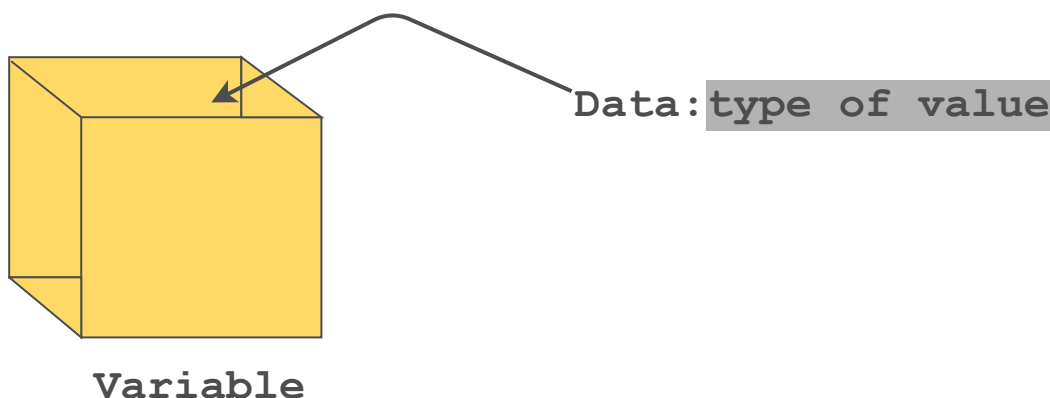
This lesson will get you acquainted with the basic data types in Rust.

We'll cover the following

- Data Type
- How to Define a Type in Rust?
 - Implicit Definition
 - Syntax
 - Explicit Definition
 - Syntax
- Primitive Types
- Quiz

Data Type

Rust is a **statically typed** language, meaning, it must know the type of all variables at compile time.



How to Define a Type in Rust?

We can define a variable in rust in two different ways:

Implicit Definition

Unlike other languages like C++ and Java, Rust can infer the type from the type of

value assigned to a variable.

Syntax

The general syntax is:

```
let variable name = value;
```

Explicit Definition

Explicitly tells the compiler about the type of variable.

Syntax

The general syntax is:

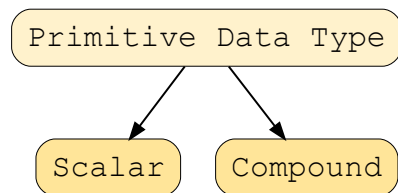
```
let variable name:datatype = value;
```

Primitive Types

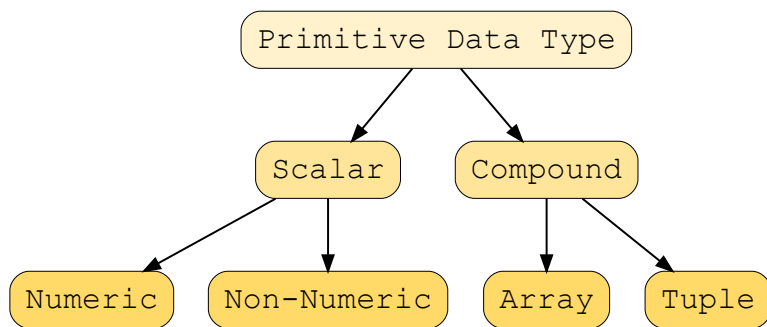
Rust has a couple of types that are considered primitive. That means they are built-in to the language. There are different data types used for different purposes.

The following illustration shows the different primitive data types in Rust:

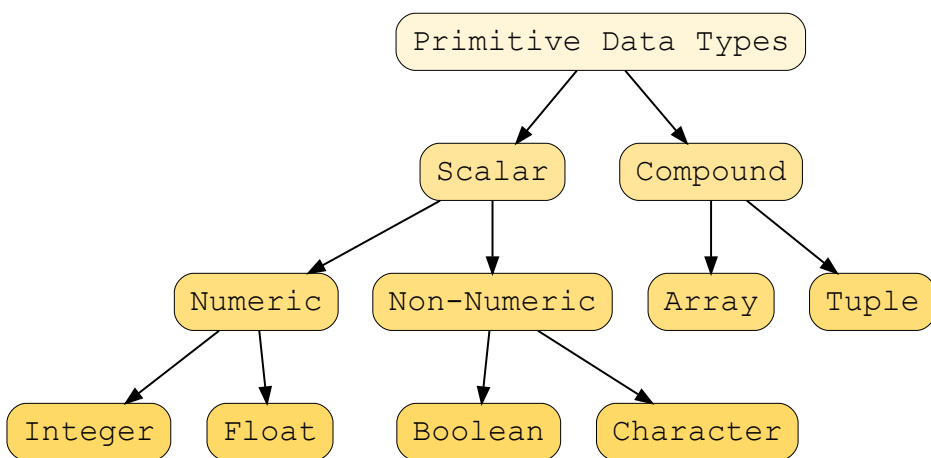
Primitive Data Types



2 of 4



3 of 4



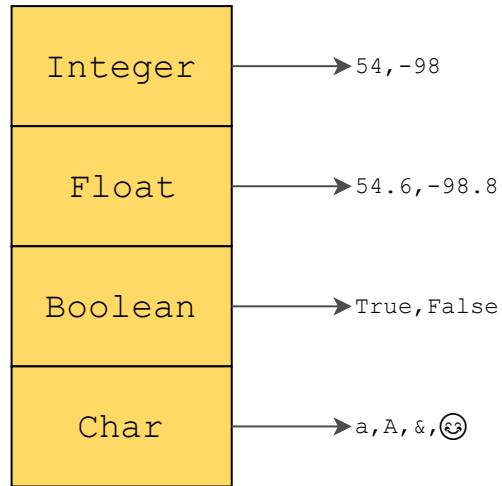
4 of 4

Scalar Type

They store a single value.

Below is the list of scalar types:

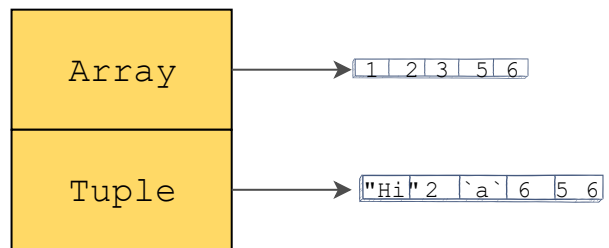
- Integers
- Floats
- Boolean
- Character



Compound Type

They group multiple values in one variable. Below is the list of compound types:

- Array
- Tuple



Quiz

Test your understanding of Data Types in Rust!

Quick Quiz on Basics of Data Types!



Which of the following is a scalar data type?



In Rust it is a must to define the type of the variable.

[Retake Quiz](#)

In the following lessons, we will learn about each data type in detail!