

Data Types in Solidity Smart Contract

Part 18 - Booleans and Integers

Programming Languages by Typing

Statically-typed programming languages

- Type checking performed during compile time
- Slower for the first run, due to compilation
- Much faster later, due to machine-code execution

Dynamically-typed programming languages

- Type checking performed during run time
- Approximately the same speed on every run
- Generally slower than compiled code

Types of Data

Value types

- Simple data types
- The value is always copied and passed to the called function or in an assignment expression

Reference types

- Usually more complex data type
- The reference (address-equivalent) is passed to the called function or in an assignment expression

Default values - no null or undefined values

Boolean Data Type

The simplest value type

- Only two possible values, `true` and `false`

Available operators:

- `!` (logical negation, changes `true` into `false` and vice versa)
- `&&` (logical conjunction, "and")
- `||` (logical disjunction, "or")
- `==` (equality test)
- `!=` (inequality test)

Integer Data Type

A more complex data type

- Possible values are determined by the type of the variable, `(u)int + suffix 8`
: 256

Four operation categories with specific operators:

- Comparisons: `<=`, `<`, `==`, `!=`, `>=`, `>` (evaluate to bool)
- Bit operators: `&`, `|`, `^` (bitwise exclusive or), `~` (bitwise negation)
- Shift operators: `<<` (left shift), `>>` (right shift)
- Arithmetic operators: `+`, `-`, unary `-` (only for signed integers), `*`, `/`, `%` (modulo), `**` (exponentiation)