# Crystal (programming language)

**Crystal** is a general-purpose, object-oriented programming language, designed and developed by Ary Borenszweig, Juan Wajnerman, Brian Cardiff and more than 300 contributors.[4] With syntax inspired by the language Ruby,[3] it is a compiled language with static type-checking, but specifying the types of variables or method arguments is generally unneeded. Types are resolved by an advanced global type inference algorithm.[5] Crystal is currently in active development. It is released as free and open-source software under the Apache License version 2.0.

## Contents



**Crystal**

| | |
|---|---|
| **Paradigm** | Multi-paradigm: object-oriented, concurrent |
| **Designed by** | Ary Borenszweig, Juan Wajnerman, Brian Cardiff |
| **Developer** | Manas Technology Solutions |
| **First appeared** | June 18, 2014 |
| **Preview release** | 0.35.1 / June 19, 2020[1] |
| **Typing discipline** | static, inferred, structural, duck |
| **Implementation language** | Crystal |
| **Platform** | IA-32 (i386), x86-64, AArch64[2] |
| **OS** | Linux, macOS, FreeBSD, OpenBSD[2] |
| **License** | Apache License 2.0 |
| **Filename extensions** | .cr |
| **Website** | crystal-lang.org (https://crystal-lang.org) |
| **Influenced by** | |
| Ruby, C, Rust, Go,[3] C#,[3] Python[3] | |

## History

Work on the language began in June 2011,[6] with the aim of merging the elegance and productivity of Ruby with the speed, efficiency, and type safety of a compiled language.[7][6] Initially named *Joy*, it was quickly renamed to *Crystal*.[6]

The Crystal compiler was first written in Ruby, but later rewritten in Crystal, thus becoming self-hosting, as of November 2013.[8] The first official version was released in June 2014.[9] In July 2016, Crystal joined the TIOBE index.

## Description

Although resembling the Ruby language in syntax, Crystal compiles to much more efficient native code using an LLVM backend, at the cost of precluding the dynamic aspects of Ruby. However, the advanced global type inference used by the

Crystal compiler, combined with the use of union types, gives Crystal the feel of a higher-level scripting language more so than many other comparable programming languages. The language has automated garbage collection and currently offers a Boehm collector. Crystal possesses a macro system and supports generics as well as both method and operator overloading. Crystal's concurrency model is inspired by communicating sequential processes (CSP) and implements light-weight fibers and channels (for communicating between fibers) inspired by Go.[3]

# Examples

## Hello World

This is the simplest way to write the Hello World program in Crystal:

```
puts "Hello World!"
```

The same as in Ruby.

Or using an object-oriented programming style:

```
class Greeter
  def initialize(@name : String)
  end

  def salute
    puts "Hello #{@name}!"
  end
end

g = Greeter.new("world")
g.salute
```

## HTTP server

```
require "http/server"

server = HTTP::Server.new do |context|
  context.response.content_type = "text/plain"
  context.response.print "Hello world! The time is #{Time.now}"
end

server.bind_tcp("0.0.0.0", 8080)
puts "Listening on http://0.0.0.0:8080"
server.listen
```

## TCP echo server

```
require "socket"

def handle_client(client)
  message = client.gets
  client.puts message
end

server = TCPServer.new("localhost", 1234)
while client = server.accept?
  spawn handle_client(client)
end
```

### Type inference and union types

The following code defines an array containing different types with no usable common ancestor. Crystal automatically creates a union type out of the types of the individual items.

```
desired_things = [:unicorns, "butterflies", 1_000_000]
p typeof(desired_things.first) # typeof returns the compile time type, here (Int32 | String | Symbol)
p desired_things.first.class   # the class method returns the runtime type, here Symbol
```

### Concurrency

Channels can be used to communicate between fibers, which are initiated using the keyword `spawn`.

```
channel = Channel(Int32).new

spawn do
  puts "Before first send"
  channel.send(1)
  puts "Before second send"
  channel.send(2)
end

puts "Before first receive"
value = channel.receive
puts value # => 1

puts "Before second receive"
value = channel.receive
puts value # => 2
```

## Further reading

- St. Laurent, Simon; Balbaert, Ivo (February 1, 2019), *Programming Crystal* (https://pragprog.com/book/crystal/programming-crystal) (P1.0 ed.), Pragmatic Bookshelf, ISBN 978-1-68050-286-2
- Wartala, Ramon (March 2016), "Die Ruby-artige Programmiersprache Crystal" (https://www.linux-magazin.de/ausgaben/2016/03/crystal/) [The Ruby-like programming language Crystal], *Linux Magazin* (in German) (03/2016), ISSN 1432-640X (https://www.worldcat.org/issn/1432-640X)

## References


1. "Releases" (https://github.com/crystal-lang/crystal/releases). Retrieved February 18, 2020 – via GitHub.
2. "Platform Support" (https://github.com/crystal-lang/crystal/wiki/Platform-Support). *Crystal Wiki* – via GitHub.
3. Borenszweig, Ary (June 16, 2016). "Crystal 0.18.0 released!" (http://crystal-lang.org/2016/06/14/crystal-0.18.0-released.html#comment-2732771703). *crystal-lang.org*. "It's heavily inspired by Ruby, and other languages (like C#, Go and Python)."
4. "Contributors" (https://github.com/crystal-lang/crystal/graphs/contributors). Retrieved July 25, 2019 – via GitHub.
5. Brian J., Cardiff (September 9, 2013). "Type inference part 1" (http://crystal-lang.org/2013/09/23/type-inference-part-1.html). *crystal-lang.org*.

6. David, María Inti (April 1, 2016). "The story behind #CrystalLang" (https://manas.tech/blog/2016/04/01/the-story-behind-crystal/). *manas.tech*.
7. Hsieh, Adler (September 20, 2015). "Why Crystal programming language?" (http://motion-express.com/blog/why-use-crystal-lang). *motion-express.com*.
8. Borenszweig, Ary (November 14, 2013). "Good bye Ruby Thursday" (https://crystal-lang.org/2013/11/14/good-bye-ruby-thursday.html). *crystal-lang.org*.
9. Borenszweig, Ary (June 19, 2014). "Crystal 0.1.0 released!" (https://crystal-lang.org/2014/06/19/crystal-0.1.0-released.html). *crystal-lang.org*.

# External links

- Official website (http://crystal-lang.org)
- Crystal-lang (https://github.com/crystal-lang) on GitHub
- /r/crystal_programming subreddit (https://reddit.com/r/crystal_programming)
- Crystal Announcements (https://crystal-ann.com)