**RYBEON**

**WHAT IS RYBEON**

Rybeon is a fundamental programming technology (AKA: FPT or general-purpose programming language) that has perfect appeal, chemistry, ease, maturity, reliability, and security. It is also an FPT designed based on a good and better programming approach, compared to mainstream approaches.

All existing mainstream FPTs have only some of the above characteristics:

C [Cee] offers ease and maturity, but lacks appeal, chemistry, reliability, and security;

Java offers reliability, security, and maturity, but lacks appeal, chemistry, and ease;

C++ offers maturity, but lacks appeal, chemistry, ease, reliability, and security;

Python offers appeal, ease, and maturity, but lacks chemistry, reliability, and security.

Also, these programming languages were designed around poor programming approaches.

We need an FPT that is beautiful all-round (or at least quite beautiful all-round); Golang and Rust take shots at being this desired perfect FPT, but they are mediocres. Rybeon is that attempt to finally create a truly perfect FPT. Rybeon is the epitome of FPT.

**Composition**

Rybeon is composed of: a fundamental programming language (Rybe) and supporting technologies like - standard code repository, reapplicable (reusable) code infrastructure, language translation softwares, language migration technologies, etc.

**Potential Users**

Rybeon is meant for all types of software engineers: governments, organizations, companies, startups, small businesses, groups, individuals, AIs, aliens, etc.

**GENESIS**

The creator of Rybeon (Brian Ibrahim Qamardeen) is a Software Engineer who had used a total of 12 programming languages (C, PHP, JavaScript, SQL, C++, Java, Python, Ruby, Lua, Kotlin, Golang, and Rust).

Often, when he picked up a new programming language, the language taught him new programming philosophies, some of which he found phenomenal and epiphanic.

After picking up Golang and Rust and consequently experiencing epiphanies again, he started to wonder if there were more beautiful programming philosophies.

As a perfectionist, Brian started to wonder how he could acquire all the beautiful programming philosophies that possibly existed.

He soon realized that, if he wants to acquire them all, he has to start approaching programming according to how it should be done, and not how existing languages encourage it to be done.

When Brian started to approach programming according to how it should be done, he started to see many ways programming could be significantly better, compared to mainstream approaches.

Brian started trying to adopt his newly discovered programming approach to his day-to-day programming. Because existing languages were not designed based on his newly discovered approach, it was very difficult to adopt the approach, using any of the existing languages.

Because Brian could not adopt his approach using any of the existing languages, he decided to create a language that enables him to do so. This decision gave birth to Rybeon in October 2019.

**The Initial Idea Has Grown**

Although Brian’s initial intention was just to create a programming language, he quickly realized that he had to morph the project into a bigger idea: a suite of programming technologies (a language and supporting technologies like infrastructure that facilitate code reapplication, technologies that can help interested entities convert their codebase from other languages to Rybe, etc).

So Rybeon is a suite of programming technologies, and not just a programming language. The language within it is called Rybe.

**VISION**

Living and Mature