

# 12in23 - Prolog - Wielding Prolog during Mindshifting May

Written by Seth Corker on Benevolent Bytes

Cover photo by Dawid Tkocz on Unsplash

## Initial thoughts

I chose prolog because I've heard about it here and there, but never looked at it seriously. I think Exercism had the right idea calling it Mindshifting May and not just for the alliteration, Prolog was mind-bending. It's nothing like any language I've used before, and it's fascinating, I just have no idea where to use it!

## Language features

The whole selling point of prolog is that it's a logical and declarative language which you use to build up rules and represent your facts then you can query those facts and rules. This is a really intriguing idea. You essentially build a program that can answer very specific questions.

It looks a little something like this:

```
coffee(flat_white).
coffee(espresso).
juice(pineapple).
juice(sugar_cane).

drink(X) :- juice(X).
drink(X) :- coffee(X).
```

That's the program. Coming from traditional C-style languages, this is really challenging to grasp. Our program is a set of rules, this one creates some facts like `flat_white` is a `coffee` and some rules like `drink(X) :- coffee(X)` which means any `coffee` is also a `drink`. Now what can we do with this?

We can query it! When we run our program, we can ask it questions.

```
?- drink(flat_white).
true.

?- juice(flat_white).
false.

?- drink(almond).
false.
```

The closest analogous language to me is SQL. You provide a query of what you want to receive back, and the database will make it happen.

## Developer experience

The most challenging part was Prolog itself, getting things up and running was okay. There's a Prolog extension for Visual Studio Code which adds syntax highlighting and autocomplete, and I was able to install Prolog easily with an `asdf` plugin. The greatest challenge was shifting my mindset, every programming problem became far more challenging as Prolog is a language that I've never encountered before. I didn't get very far into my journey, I did a few exercises that took far longer than anticipated, and before I knew it the month was over.

## Takeaways

Prolog is fascinating to me. It's the first language I've encountered as a professional programmer who forced me to throw out everything I knew. Julia was a language I could do something with by skimming the documentation and hacking something together. Prolog was not, it's something I want to look at further and give it a proper go. I found it difficult to contextualise in the tools I use day to day, I found myself asking: - What problems is Prolog good for? - How could I integrate this with other ecosystems? - How do I do X in Prolog? Near the end of my journey I found the Power of Prolog, and it goes into a lot more detail about different aspects of the language. If I had found this sooner, I may have made more progress. Overall, there's something about the language that draws me in, it's not like anything I've ever used before, and I want to revisit it when I have the headspace to do so.

## Resources

- [The Power of Prolog](#)
- [Mindshifting May](#) blog post on Exercism
- [Prolog on LearnXinYminutes](#)
- [Sample code on Replit](#)