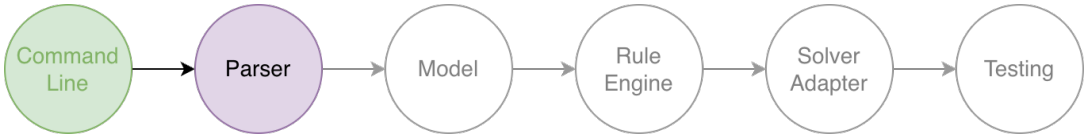
Oxidelings: **Parser**

Task 2 of 6

Last Modified 28/12/2025



1 Background

Next on our journey through oxide's path of execution is the **Parser**. Previously, we made the CLI call a method `run_solve_command` on encountering the command `conjure-oxide solve <file>`. One of the first steps of `run_solve_command` is to create a model by **parsing**. Up to date details can be found on the wiki (link needs updated). Essentially the input (e.g. 'file.essence') is just a string of characters. This string needs checked to ensure validity, and it needs stored in a structured data type (i.e. the model object). The `tree-sitter-essence` crate describes the Essence language, and the `conjure-cp-essence-parser` crate is conjure-oxide using this description to *do* the parsing.

2 The Task

Hexer-the-Vexer is back! The cunning conjurer wants to stop all the other conjurers from *finding* variable assignments. He tried to get rid of `solve` on the CLI but thankfully you stopped him. But this time, he's destroyed oxide's ability to parse `find_statement`s!

Files: You will need to edit `conjure-cp-essence-parser::parser::find.rs`. You will **not** need to make any changes in `tree-sitter-essence`.

Grammar: It may be useful to consider the grammar pertaining to find statements. Essentially, a valid Essence string is made up of top-level statements (find statements being one of them). The find statement comprises a *list* of variables, a colon, and a domain.

```
        $Program -> find  $.find_statement | ...
$.find_statement -> $.variable_list : $.domain
 $.variable_list -> $variable | $variable, $variable_list
```

At the top of `find.rs` there is a long-form explanation, and there are `//TODO:` comments in the function with a little bit of guidance. There are also hints on the next page, but you will get the most out of it if you give it a go first so hold off on turning the page just yet.

3 Hints

3.1 Hint A

To find the domain node, you will need to use the `child_by_field_name` function on the `find_statement` node, with the name of the domain node (i.e. "domain"). This is similar for the variable list node.