

# Parse Request Line

---

 [online.pragmaticstudio.com/courses/elixir/steps/8](https://online.pragmaticstudio.com/courses/elixir/steps/8)

## Notes

---

### What's a Term?

---

In the video we used the word *term* as in "match the *term* on the right-hand side". If you poke around the Elixir documentation you'll see references to *term* as well.

A *term* is a value of any data type: a string, an atom, a map, a list, etc.

### Shortcut: Using Atoms As Map Keys

---

Elixir atoms are prefixed by a colon. For example, here's a map that uses atoms as keys:

```
%{ :method => "GET", :path => "/wildthings" }
```

However, it's so common to use atoms as keys that Elixir gives us a shortcut. And we love shortcuts, so we used it in the video:

```
%{ method: "GET", path: "/wildthings" }
```

This form is more concise: we removed the `=>` and put the colon (`:`) on the other side of the atom. (Wow, that sounded scientific!)

If the keys are anything but atoms, you must use the general `=>` form. For example, here's a map with strings as keys:

```
%{ "method" => "GET", "path" => "/wildthings" }
```

### Code So Far

---

The code for this video is in the `parse` directory found within the `video-code` directory of the [code bundle](#).

[Go To Next Video](#)