

# CMDR - interactive command line interfaces in Rust

---

Mendelt Siebenga

or Procedural macro's for fun and profit

Who am I?

- Recovering Python programmer
- I program Scala for money
- I play with Rust for fun

# This presentation

What do I want to show you

- Part 1: Show off CMDR
- Part 2: How does it work
  - Procedural macros

## Some Python (Cmd)

```
import cmd

class HelloWorld(cmd.Cmd):
    """Simple command processor example."""

    def do_greet(self, line):
        print "hello"

    def do_EOF(self, line):
        return True

if __name__ == '__main__':
    HelloWorld().cmdloop()
```

How do we do this in rust?

- No class based inheritance
  - But we do have traits
- No introspection/reflection
  - But we do have macros

- Handling commands
- Help function
- Changing the prompt
- Hooks
- Command scopes

# Planned functionality for 1.0

What is coming?

- Quality of life improvements;
  - More examples
  - Better documentation
  - Better error messages
- Tab completion
  - Commands
  - Parameters
- Typed argument parsing



## Some usage suggestions

What can you use Cmdr for?

- Old school text based adventure/dungeon crawlers
- Management application for your system
- Test harness for an API

What you would have to type without `#[cmdr]`

A bit more about macro's

- `macro_rules!` macros
  - `println!`
- derive macros
  - `serde`
- procedural macros

# The #cmdr macro

Dive into `cmdr_macro`

A handy tool to show macro expansion

- [crates.io crates.io/crates/cmdr](https://crates.io/crates/cmdr)
- code [github.com/mendelt/cmdr](https://github.com/mendelt/cmdr)
  - with examples
- [docs.rs/cmdr](https://docs.rs/cmdr)
- cargo generate template: [github.com/mendelt/cmdr-template](https://github.com/mendelt/cmdr-template)
- This presentation: [github.com/mendelt/cmdr-presentation](https://github.com/mendelt/cmdr-presentation)

# Questions?