



Writing command line apps in Rust

Quick intro

bit.ly/rust-cli-intro

Some examples

bit.ly/rust-quickli

Useful libraries

Argument parsing

[clap.rs](https://crates.io/clap)

Deal with JSON, TOML, XML, etc.

[serde.rs](https://crates.io/serde)

Find more libraries

crates.io

Learn more about the Rust project

rust-lang.org

Stay informed about Rust

youtube.com/RustVideos

this-week-in-rust.org

► `cargo new tiny-grep`

► `echo 'structopt = "0.2"' >> Cargo.toml`

► *# Write the following in src/main.rs:*

```
#[macro_use] extern crate structopt;
```

```
use structopt::StructOpt;
```

```
/// Search for a pattern like "ulalala" in a file
```

```
#[derive(StructOpt)]
```

```
struct Cli {
```

```
    /// The pattern to look for
```

```
    pattern: String,
```

```
    /// The path to the file to read
```

```
    path: String,
```

```
}
```

```
fn main() {
```

```
    let args = Cli::from_args();
```

```
    let content = std::fs::read_to_string(&args.path)
```

```
        .expect("could not read file");
```

```
    for (n, line) in content.lines().enumerate() {
```

```
        if line.contains(&args.pattern) {
```

```
            println!("{}", n, line);
```

```
        }
```

```
    }
```

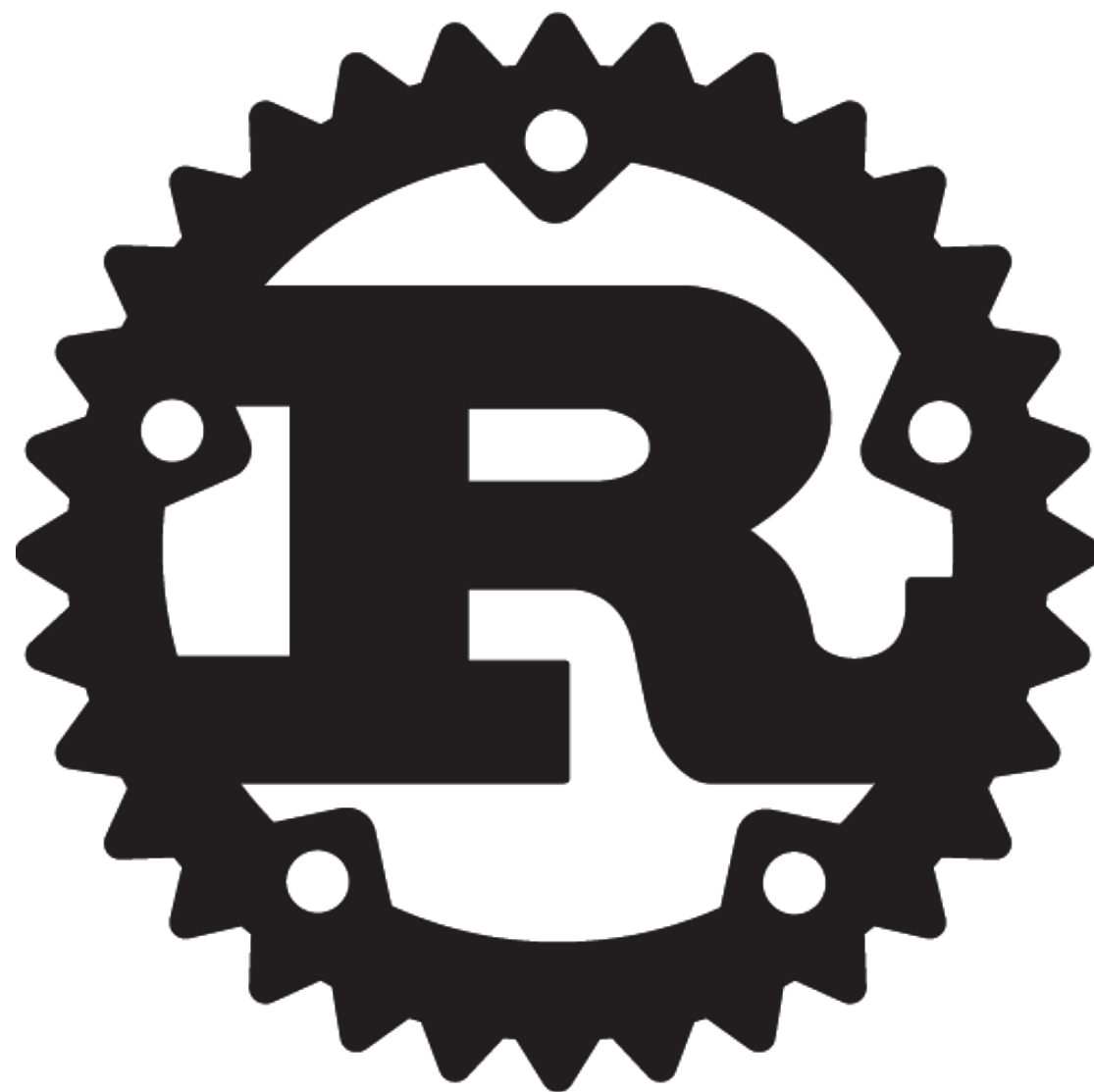
```
}
```

► `cargo run`

► `cargo run -- --help`

► `cargo run -- ulalala src/main.rs`

3: */// Search for a pattern like "ulalala" in a file*



`rust-lang.org`