

# Documentation with Haddock

Artem Chernyak

2020

# Why Document?

- ▶ Explain why the code exists
- ▶ Provide examples
- ▶ Demonstrate workflow

# Why Haddock?

- ▶ Light weight format
- ▶ Documentation lives with the code

# Understands Haskell

- ▶ Automatic hyperlinking
- ▶ Only exports exposed documentation

# Integrates with Hoogle

- ▶ Provides search for all haskell documentation
- ▶ Integrates with your project
- ▶ Index all of Stackage: `hoogle generate`

# Minimal Requirements

## Example (shell.nix)

```
{ pkgs ? import <nixpkgs> { } }:  
  
mkShell {  
  buildInputs = with pkgs; [  
    ghc  
    cabal-install  
    haskellPackages.hoogle  
  ];  
}
```

# Generate Haddock

```
$ cabal haddock --haddock-hoogle --haddock-html
```

- ▶ `--haddock-hoogle` generates a `.txt` file read by Hoogle
- ▶ `--haddock-html` generates the web content

# Update Hoogle

```
$ hoogle generate --local=dist-newstyle/
```

- ▶ You have to specify the local dist directory to properly link documentation



# Start Hoogle server

```
$ hoogle server -p 4444 --local
```

- ▶ `--local` provides links directly to documentation files in the project