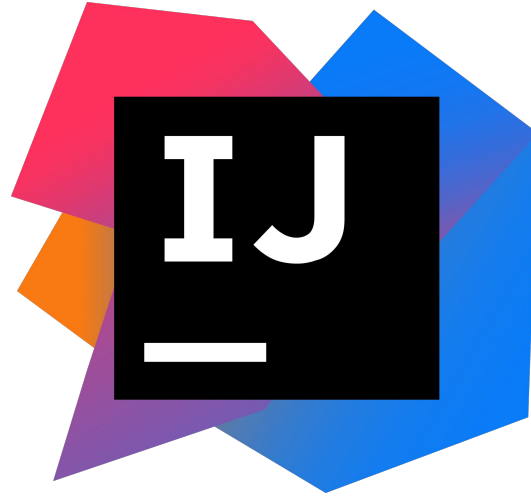


# IntelliJ-Haskell plugin



# Agenda

1. Overview
2. IntelliJ Platform SDK
3. Features
4. Integrated tools
5. In short
6. #TODO
7. Questions

# Overview 1

- Support Stack projects
- Only prerequisite is stack installed
- Existing (multi-package) project can be imported
- New project can be created based on stack template
- IntelliJ modules are created by reading stack.yaml and Cabal file(s)
- After opening project, project is built and cache is filled with library modules info. Package info by ghc-pkg tool.

# Overview 2

- Library sources are downloaded from hackage
- GHCi in background for type checking, (type) info and definition location
- REPL per component/stanza and REPL for library modules
- Less support in library files
- Project libraries will be rebuilt in background after code changes *and* when user navigates to module which depends on library, like test module

# IntelliJ Platform SDK

- IntelliJ platform, provides infra for IDE's
- Lexer
- Parser
- Program Structure Interface (PSI)
- Extension points to enable IDE feature for custom language
- Annotator for compiler messages
- Indexing

For more info: <http://www.jetbrains.org/intellij/sdk/docs/welcome.html>

# Features 1

- Syntax highlighting
- Error/warning annotations
- Code completion (also for import declarations and language extensions)
- Various kind of navigation (by id, name of declaration, Hoogle, instance declaration)

# Features 2

- Find usages
- Show documentation
- Renaming
- Code inspection: HLint suggestions and applying
- Smart completion
- Reformatting

# Integrated tools 1

- HLint
  - Linter/inspection runs after each code change
- Hindent
  - Reformatter
- Stylish-haskell
  - Imports prettifier
- Hoogle
  - Haddock documentation
  - Search by name and type signature



## Integrated tools 2

- Haskell tools are automatically built with Stackage LTS in background.
- They are installed by stack inside IntelliJ-Haskell sandbox.
- User can request to update tools (within LTS)
- Same tools are used for every project.
- Hoogle integration has to be triggered by creating Hoogle database

# IntelliJ-Haskell in short

Main selling point is that no GHC backend is needed.

Haskell tools are automatically installed.

A lot of IntelliJ features are implemented.

Navigation to library code is supported.

*But there is still a lot of room for improvement!*

Contributions are welcome and needed!

# #TODO

- Improve HLint integration
- Layout sensitive parser
- Much more tests
- Integration with GHC 8.8 which generates hie-files
- Support Cabal and Nix projects
- Reformatter by using PSI
- Improve test integration
- Website with documentation and tutorials
- Debugger?

# Thanks for your attention!

Try IntelliJ-Haskell and I would like to hear your feedback!

<https://plugins.jetbrains.com/plugin/8258-intellij-haskell>

<https://github.com/rikvdkleij/intellij-haskell>

## Questions???