# Migrating to TypeScript

# Current State of Affairs

If you're not careful, JavaScript can become a write-only language

# What Do We Need?

- Better tooling
    - Prevent simple mistakes
    - Improved code discoverability and navigation
    - Support for refactoring
- Language features
    - Ability to leverage newer JavaScript features

# Possible Solutions

- Testing
  - Unit testing
  - Integration testing
- Tooling
  - WebStorm
  - Babel
- Static typing
  - Flow
  - TypeScript

# What is TypeScript?

- Superset of JavaScript
- Optional static typing
- Features from the future

# Incremental Migration

- Not an "all or nothing" proposition
- Migrate as you want to or have time

# Demos

# Wrap Up

# Common Migration Issues

- Module imports and exports
- Arguments object
- Sequentially added properties
- And more :)

Adding static types to your code doesn't negate the need for testing

# Toolchain

- Front end task runners
  - Gulp
  - Grunt
  - Webpack
- Linting
  - TSLint

# Resources

- Official TypeScript Website
  - https://www.typescriptlang.org/
- Basarat Ali Syed's TypeScript Deep Dive
  - https://www.gitbook.com/book/basarat/typescript

# Thanks!

James Churchill

- Treehouse: https://teamtreehouse.com/jameschurchill
- Twitter: @SmashDev
- GitHub: smashdevcode

https://github.com/smashdevcode/boise-code-camp-2017