# Label Bugs

The highest priority is getting unlabeled bugs to zero.

Query: Open Unlabeled Issues

# How to Label a Bug

### Change the Title

Most issues have pretty bad titles, hindering future searches. If needed, edit the issue title to align better with the [[Preferred Issue Titles]] format.

If you can't figure out from the issue report what the title *should* be, then you'll definitely need clarification from the user (see "Needs More Info" below).

### Add a Label

Classify the bug accordingly: * Duplicate: Many issues are duplicates, so try to find an original * If you do, add the `Duplicate` label and add a comment e.g. "See #1234567" * If it's clearly an exact duplicate, Close * Optional: If the user would have found this with a trivial search (e.g. searching for the title of their own bug), gently remind them to search before logging new issues * Legitimate bug (crash, incorrect behavior, etc.): Add the `Bug` label * Optionally add `High Priority` if it's an easy-to-hit crash or incorrect emit * Optionally add one of the `Domain:` labels if you'd like to * Suggestion * Add `Suggestion` and `In Discussion` if this is something that can be looked at immediately * Add `Suggestion`, `Out of Scope`, and close if the suggestion is not something we would ever do (change JS runtime semantics, emit Python, switch to Haskell's type system, etc). Add a comment pointing to the Design Goals Wiki page * Add `Suggestion`, `Needs Proposal` if something requiring a more formal description is required. Add a comment indicating what's needed * Question (the user is explicitly asking for help) * Add the `Question` label * Provide an answer if it's easy for you to do so, otherwise point them at Stack Overflow and remind that the issue tracker is not a support forum * Close the bug if it's egregiously out of scope (e.g. asking for help getting Angular2 working or whatever) * If the question is about the compiler API and you can't answer it immediately, assign to a dev * Not a bug * Add `Working as Intended` if the behavior is truly done on purpose, or `Design Limitation` if it's something we *would* fix in a perfect world but are unable to do so * Post a comment explaining why. Try to reference things from the FAQ if applicable; consider updating the FAQ if you think it's a common question * It's not clear what the issue is describing * Add the `Needs More Info` label * Add a comment explaining why the issue isn't actionable yet * Issue is in external component (e.g. tslint, awesome-typescript-loader, etc) * Add the `External` label * Explain why * Completely useless * If the issue is completely unsalvageable (e.g. it's just "Why can TypeScript not for C# now?" with no other info), add the `Unactionable` label and Close. * Fallback: Not

sure * Add "Needs Investigation" label * Optional: Post comment with your thoughts (e.g. "Might be a type inference bug, need to investigate")

## Investigate

Once there are no new unlabeled bugs, start looking at issues which need investigation: Query: Bugs needing investigation