# Contributing to a Project

Now that you’ve found the material for understanding the project, here is how you can take action.

### Create an Issue

If you find a bug in a project you’re using (and you don’t know how to fix it), have trouble following the documentation or have a question about the project – create an issue! There’s nothing to it and whatever issue you’re having, you’re likely not the only one, so others will find your issue helpful, too. For more information on how issues work, check out our Issues guide.

#### Issues Pro Tips

Check existing issues for your issue. Duplicating an issue is slower for both parties so search through open and closed issues to see if what you’re running into has been addressed already.

Be clear about what your problem is: what was the expected outcome, what happened instead? Detail how someone else can recreate the problem.

Link to demos recreating the problem on things like JSFiddle or CodePen.

Include system details like what the browser, library or operating system you’re using and its version.

Paste error output or logs in your issue or in a Gist. If pasting them in the issue, wrap it in three backticks: ``` so that it renders nicely.

### Pull Request

If you’re able to patch the bug or add the feature yourself – fantastic, make a pull request with the code! Be sure you’ve read any documents on contributing, understand the license and have signed a CLA if required. Once you’ve submitted a pull request the maintainer(s) can compare your branch to the existing one and decide whether or not to incorporate (pull in) your changes.

#### Pull Request Pro Tips

Fork the repository and clone it locally. Connect your local to the original ‘upstream’ repository by adding it as a remote. Pull in changes from ‘upstream’ often so that you stay up to date so that when you submit your pull request, merge conflicts will be less likely. See more detailed instructions here.

Create a branch for your edits.

Be clear about what problem is occurring and how someone can recreate that problem or why your feature will help. Then be equally as clear about the steps you took to make your changes.

It’s best to test. Run your changes against any existing tests if they exist and create new ones when needed. Whether tests exist or not, make sure your changes don’t break the existing project.

Include screenshots of the before and after if your changes include differences in HTML/CSS. Drag and drop the images into the body of your pull request.

Contribute in the style of the project to the best of your abilities. This may mean using indents, semi colons or comments differently than you would in your own repository, but makes it easier for the maintainer to merge, others to understand and maintain in the future.

#### Open Pull Requests

Once you’ve opened a pull request a discussion will start around your proposed changes. Other contributors and users may chime in, but ultimately the decision is made by the maintainer(s). You may be asked to make some changes to your pull request, if so, add more commits to your branch and push them – they’ll automatically go into the existing pull request.

If your pull request is merged – great! If it is not, no sweat, it may not be what the project maintainer had in mind, or they were already working on it. This happens, so our recommendation is to take any feedback you’ve received and go forth and pull request again – or create your own open source project.