# Questions

If you have issues with the APIs or have a question about the Watson services, see [Stack Overflow](https://stackoverflow.com/questions/tagged/ibm-watson+node.js).

# Code

\* Our style guide is based on [Google's](https://google.github.io/styleguide/jsguide.html), most of it is automaticaly enforced (and can be automatically applied with `npm run tslint:fix` or `npm run eslint:fix`)

\* Commits should follow the [Angular commit message guidelines](https://github.com/angular/angular/blob/master/CONTRIBUTING.md#-commit-message-guidelines). This is because our release tool uses this format for determining release versions and generating changelogs. To make this easier, we recommend using the [Commitizen CLI](https://github.com/commitizen/cz-cli) with the `cz-conventional-changelog` adapter.

# Issues

If you encounter an issue with the Node.js library, you are welcome to submit

a [bug report](https://github.com/watson-developer-cloud/node-sdk/issues).

Before that, please search for similar issues. It's possible somebody has

already encountered this issue.

# Pull Requests

If you want to contribute to the repository, follow these steps:

1. Fork the repo.

2. Install dependencies: `npm install`

3. Build the code: `npm run build`

4. Verify the build before beginning your changes: `npm run test-unit`

2. Develop and test your code changes.

3. Travis-CI will run the tests for all services once your changes are merged.

4. Add a test for your changes. Only refactoring and documentation changes require no new tests.

5. Make the test pass.

6. Commit your changes. Remember the follow the correct commit message guidelines.

7. Push to your fork and submit a pull request.

8. Be sure to sign the CLA.

# Developer's Certificate of Origin 1.1

By making a contribution to this project, I certify that:

(a) The contribution was created in whole or in part by me and I

have the right to submit it under the open source license

indicated in the file; or

(b) The contribution is based upon previous work that, to the best

of my knowledge, is covered under an appropriate open source

license and I have the right under that license to submit that

work with modifications, whether created in whole or in part

by me, under the same open source license (unless I am

permitted to submit under a different license), as indicated

in the file; or

(c) The contribution was provided directly to me by some other

person who certified (a), (b) or (c) and I have not modified

it.

(d) I understand and agree that this project and the contribution

are public and that a record of the contribution (including all

personal information I submit with it, including my sign-off) is

maintained indefinitely and may be redistributed consistent with

this project or the open source license(s) involved.

## Tests

Out of the box, `npm test` runs linting, unit tests, and integration tests (which require credentials). To run only the unit tests (sufficient for most cases), use `npm run test-unit`.

To run integration tests, copy `test/resources/auth.example.js` to `test/resources/auth.js` and fill in credentials for the service(s) you wish to test.

Currently this enables integration tests for all services so, unless all credentials are supplied, some tests will fail.

(This will be improved eventually.)

To run only specific tests, pass the file name to `jest`. For example:

`npm i -g jest`

`jest test/integration/personality\_insights.v3.test.js`

See [this page](https://jestjs.io/docs/en/cli) for all the options you can use to further configure `jest`.