To run tests

Make sure your environment is set up with:

```
yarn
```

In a separate process, run the WebSockets server:

```
yarn ws-server
```

Run the browser tests using Karma:

```
yarn test
```

Run the node.js tests:

```
yarn test-node
```

Run tslint:

```
yarn lint
```

Run format with clang-format:

```
yarn format
```

Run all checks (lint/format/browser test/test-node):

```
yarn ci
```

Before Commit

Please make sure you pass all following checks before commit

- yarn gulp lint (tslint)
- yarn gulp format (clang-format)
- yarn promisetest (promise a+ test)
- yarn bazel test //packages/zone.js/... (all tests)

Webdriver Test

`zone.js` also supports running webdriver e2e tests.

1. run locally

```
yarn webdriver-start
yarn webdriver-http
yarn webdriver-test
```

2. run locally with sauce connect

```
// export SAUCE_USERNAME and SAUCE_ACCESS_KEY
export SAUCE_USERNAME=XXXX
export SAUCE_ACCESS_KEY=XXX
```

```
sc -u $SAUCE_USERNAME -k $SAUCE_ACCESS_KEY
yarn webdriver-http
yarn webdriver-sauce-test
```

Releasing

Releasing `zone.js` is a two step process.

1. Create a PR which updates the changelog, and get it merged using normal merge process.
2. Once the PR is merged check out the merge SHA of the PR and release `zone.js` from that SHA and tag it.

1. Creating a PR for release

```
rm -rf node_modules && yarn install
export PREVIOUS_ZONE_TAG=`git tag -l 'zone.js-0.11.*' | tail -n1`
export VERSION=`(cd packages/zone.js; npm version patch --no-git-tag-version)`
export VERSION=${VERSION#v}
export TAG="zone.js-${VERSION}"
echo "Releasing zone.js version ${TAG}. Last release was ${PREVIOUS_ZONE_TAG}."
yarn gulp changelog:zonejs
```

Inspect the `packages/zone.js/CHANGELOG.md` for any issues and than commit it with this command.

Create a dry run build to make sure everything is ready.

```
yarn bazel --output_base=$(mktemp -d) run //packages/zone.js:npm_package.pack --workspace_st
```

If everything looks good, commit the changes and push them to your origin to create a PR.

```
git checkout -b "release_${TAG}"
git add packages/zone.js/CHANGELOG.md packages/zone.js/package.json
git commit -m "release: cut the ${TAG} release"
git push origin "release_${TAG}"
```

2. Cutting a release Check out the SHA on master which has the changelog commit of the zone.js

```
git fetch upstream
git checkout upstream/master
rm -rf node_modules && yarn install
export VERSION=`(node -e "console.log(require('./packages/zone.js/package.json').version)")`
export TAG="zone.js-${VERSION}"
export SHA=`git log upstream/master --oneline -n 1000 | grep "release: cut the ${TAG} releas
echo "Releasing '$VERSION' which will be tagged as '$TAG' from SHA '$SHA'."
git checkout ${SHA}
```

```
npm login --registry https://wombat-dressing-room.appspot.com
yarn bazel -- run --config=release -- //packages/zone.js:npm_package.publish --access public
git tag ${TAG} ${SHA}
git push upstream ${TAG}
```