# Contributing to GraphQL PHP

## Workflow

If your contribution requires significant or breaking changes, or if you plan to propose a major new feature,

we recommend you to create an issue on the [GitHub](https://github.com/webonyx/graphql-php/issues) with

a brief proposal and discuss it with us first.

For smaller contributions just use this workflow:

\* Fork the project.

\* Add your features and or bug fixes.

\* Add tests. Tests are important for us.

\* Check your changes using `composer check`.

\* Add an entry to the [Changelog's Unreleases section](CHANGELOG.md#unreleased).

\* Send a pull request.

## Setup the Development Environment

First, copy the URL of your fork and `git clone` it to your local machine.

```sh

cd graphql-php

composer install

```

## Running tests

```sh

./vendor/bin/phpunit

```

Some tests have annotation `@see it('<description>')`. It is used for reference to same tests in [graphql-js implementation](https://github.com/graphql/graphql-js) with the same description.

## Coding Standard

The coding standard of this project is based on [Doctrine CS](https://github.com/doctrine/coding-standard).

Run the inspections:

```sh

./vendor/bin/phpcs

```

Apply automatic code style fixes:

```sh

./vendor/bin/phpcbf

```

## Static analysis

Based on [PHPStan](https://github.com/phpstan/phpstan).

```sh

./vendor/bin/phpstan analyse --ansi --memory-limit 256M

```

## Running benchmarks

Benchmarks are run via [PHPBench](https://github.com/phpbench/phpbench).

```sh

./vendor/bin/phpbench run benchmarks

```