

# Adding tests to a project with no tests

Tom Rochette <tom.rochette@coreteks.org>

November 2, 2024 — [36c8eb68](#)

## 1 Problem

There are no tests on the project I joined. How do I get started?

## 2 Solution

When joining a new project without tests, here is the value you need to provide through the addition of tests:

1. the application works and doesn't crash
2. the application works and supports a few input cases
3. the application works and supports a variety of input cases
4. the application works and is robust to most input cases

Start by finding the main entrypoint of the program and call it in a test. Your test doesn't have to do much, other than ensuring you can start the program and possibly exit. Your goal should not be to assert anything yet, but to exert the code. Create a few tests that do very few things other than starting and terminating the program. Once you've covered a few use cases, you can use those tests to ensure that the application can start, do a few things, then terminate without crashing.

Start unit testing the various parts of the code that are critical. To determine what is critical, you'll have to dig into the code. With the tests you initially wrote you will get a sense of the "critical" pieces, simply due to the fact that they are executed whenever the program starts and stops, which are two things you always want to work.

When writing unit tests, always aim to write a test case that covers the happy path first. You want to demonstrate that the functionality a class or function is supposed to provide is there first and foremost. Then you want to test its robustness and its ability to handle a variety of input cases. Given a large codebase, start by covering most of the code with the happy paths before you start to dig into the special cases.

## 3 References

- [Writing tests](#)