

Clean Tests

Clean Code, Episode 20

Link¹

Timing:

- 00:00-05:50 introduction
- 05:50-08:30 overview
- 08:30-20:50 astronomy lesson: history of the earth (may skip)
- 20:50-29:50 anatomy of a test
- 29:50-40:57 the arrange
- 40:57-52:55 setup struggles
- 52:55-1:06:05 test hierarchy
- 1:06:05-1:10:30 clean composition
- 1:10:30-1:20:55 the assertion
- 1:20:55-1:23:10 summary
- 1:23:10-1:24:10 end and credits (may skip)

Vocabulary:

- **degenerate test**
- **setup function** - used when there is a group of test functions that share a common arrangement
- **teardown function** - used when there is a group of test functions that share a common teardown
- **fixture** - system state needed to run a test
- **transient fixture** - fixture created and destroyed for a single test
- **persistent fixture** - fixture that survives from test to test (i.e., shared)
- **fresh fixture** - fixture that is initialized every time it is used
- **shared fixture** - fixture that allows some state to accumulate from test to test

¹https://www.safaribooksonline.com/videos/clean-code/9780134661742/9780134661742-CODE_01_05_01

- **test hierarchy**
- **composed action**
- **composed assert**

Concepts:

- Four phases of every test function (The Four A's):
 1. arrange (setup) - set the system into the state necessary to run the test (a.k.a., test fixture)
 2. act (execute)
 3. assert (check)
 4. annihilate (teardown) - undo everything done in the arrange

Also: given, when, then

- Types of fixtures:
 - a. **transient fresh** - fixture that will be newly created at the beginning of a test method; no teardown is needed (preferred)
 - Means test methods cannot communicate with each other and can be run in any order
 - b. **persistent fresh** - fixture that will be newly created at the beginning of a test method, but that has some persistent state (e.g., open file or other OS resources) that must be managed in the teardown
 - c. **persistent shared** - fixture created and then used by multiple tests (e.g., database connection)
- The Single Assert Rule