

Test Coverage Analyzer

Docs

app.codereview-suite.dev/coverage

FileEditViewCoverageTestsHelp

https://github.com/your-repo/blob/main/src/MyService.java

68% covered

Run coverage

Generate tests

Export report

Java

Python

TypeScript

MyService.java68% covered

MyServiceTest.javaGenerated · 3 cases

1public List<String> getActiveUsers(List<User> users) {

2List<String> result = new ArrayList<>();

3for (int i = 0; i < users.size(); i++) {

4User u = users.get(i);

5if (u != null) {

6if (u.isActive()) {

7if (u.getEmail() != null) {

8result.add(u.getEmail());

9}

10}

11}

1// Generated test cases – covers lines 6-10

2@Test

3public void getActiveUsers\_returnsEmail\_whenUserActive() {

4User u = new User("a@x.com", true);

5List<String> r = getActiveUsers(List.of(u));

6assertEquals(List.of("a@x.com"), r);

7}

8@Test

9public void getActiveUsers\_skips\_whenInactive() {

10User u = new User("a@x.com", false);

11assertTrue(getActiveUsers(List.of(u)).isEmpty());

Uncovered methods (12)

Coverage report

Test history

●

getEmail() null branch — never executed

No test covers the path where user.getEmail() returns null.

Generate test

Generate

Line 7

●

result.add() — uncovered

The happy-path append is reached only when all three guards pass.

Branch miss

Generate

Line 8

●

isActive() — only covered for active users

Add a fixture with isActive() == false.

Partial

Generate

Line 6

●

Empty input list — no test

getActiveUsers([]) should return an empty list without errors.

Function-wide

Ready

Java

JaCoCo 0.8.11

MyService.java — 15 lines