REFLECTIONS: USE CASES

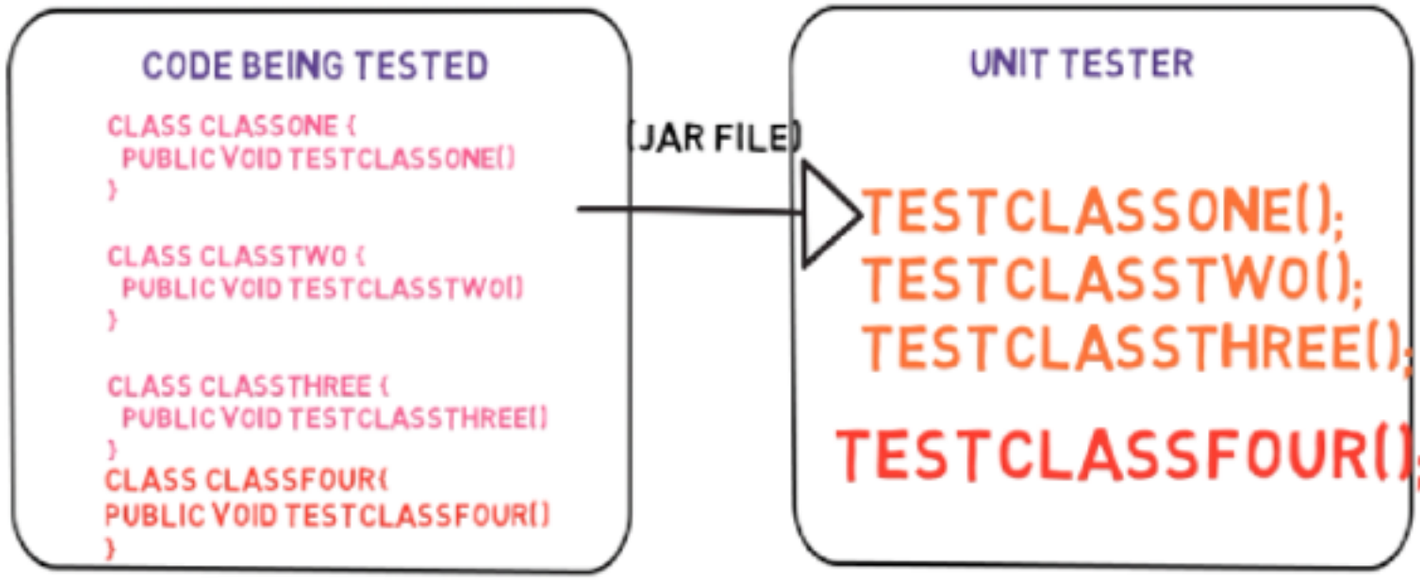
LET'S SAY WE HAVE A SMALL BIT OF CODE, WHOSE ONLY JOB IS TO RUN "UNIT TESTS" ON A MUCH BIGGER BIT OF CODE

BTW, A "UNIT TEST" IS A TEST THAT APPLIES TO A SINGLE, SMALL UNIT, FOR INSTANCE A CLASS.

THE IDEA IS TO TEST EACH INDIVIDUAL UNIT USING UNIT TESTING, AND THEN TEST THE INTER-WORKINGS OF THESE UNITS VIA INTEGRATION TESTING

AFTER BOTH UNIT AND INTEGRATION TESTING, THE OUTPUT OF THE ENTIRE SYSTEM IS TESTED USING VALIDATION TESTING.

OLD-SCHOOL APPROACH TO UNIT TESTING

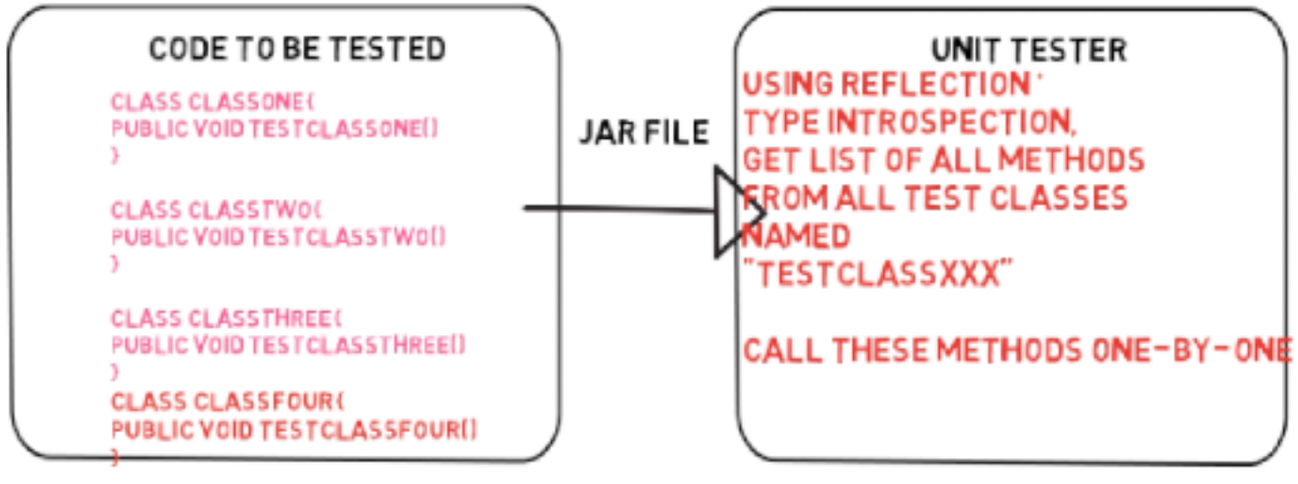


SO FAR SO GOOD.. BUT THEN

IF A NEW CLASS GETS ADDED TO THE CODE BEING TESTED, THE ONLY WAY FOR THE UNIT TESTER TO KEEP UP IS - FOR THE CODE OF THE UNIT TESTER TO BE MODIFIED, AND THE UNIT TESTER REBUILT

THIS NEED TO RECOMPILE AND RERUN THE UNIT TESTER IS A PROBLEM.

REFLECTION TO THE RESCUE!



NOW IF A CLASS IS ADDED,
NOTHING CHANGES ON THE UNIT
TESTER

THIS, IN FACT, IS EXACTLY HOW THE FAMOUS
UNIT TESTING TOOL JUNIT USED TO WORK