GMOCK

**What is gMock?**

**Mocks are used for testing the behaviour of API or interfaces wich will be used in component under test**

* When you write a prototype or test, often it's not feasible or wise to rely on real objects entirely. A mock object implements the same interface as a real object (so it can be used as one), but lets you specify at run time how it will be used and what it should do (which methods will be called? in which order? how many times? with what arguments? what will they return? etc).
* Mocks are objects pre-programmed with expectations, which form a specification of the calls they are expected to receive.
* mock objects help you remove unnecessary dependencies in tests and make them fast and reliable
* the most important thing to remember is that a mock allows you to check the interaction between itself and code that uses it.
* gMock is a library (sometimes we also call it a “framework” to make it sound cool) for creating mock classes and using them. It does to C++ what jMock/EasyMock does to Java (well, more or less).

# Compile all targets:

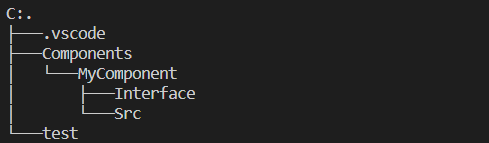
This will also download gtest from its github repo and compile it

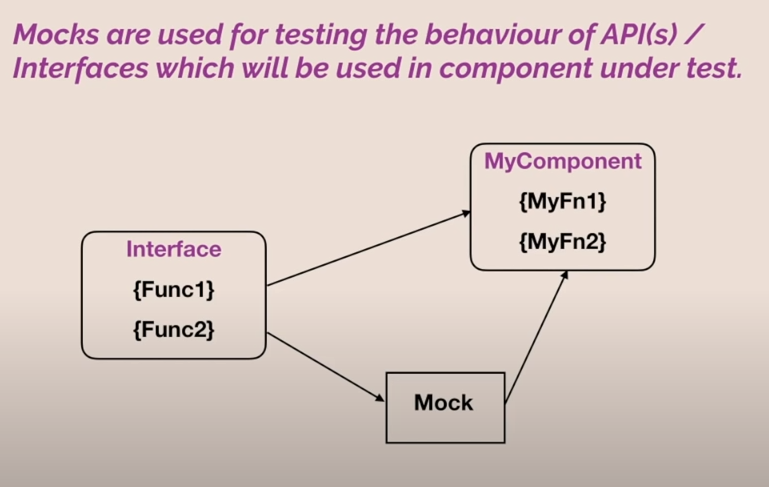
1. bazel build ...
2. Running your tests using GTest

bazel test ... --test\_output=errors

# Primer GTEST\_GMOCK\_EXAMPLE

Struktura projekta data u nastavku



* U root-u se nalazi *WORKSPACE.bazel* file koji sadrzi referencu na spoljne zavisnosti koje su neophodne za pokretanje projekta. Konkretno, u ovom slucaju potrebno je pribaviti spoljasnji repozitorijum koji sadrzi zavisnosti neophodne za pokretanje googletest-a.
* U root-u projekta se nalazi i fajl *gmock.BUILD* koji sadrzi pravila neophodna za kompajliranje gtest/gmocka.
* Projekat se sastoji iz direktorijuma Components pri cemu postoji jedna komponenta MyComponent koja je predmet testiranja:
  + Ova komponenta sadrzi MyDataBase.h i MyDataBase.cpp source fajlove kao i konekciju ka bazi (DataBaseConnect objekat) koji predstavlja spoljasnju vezu ove komponente koju ne treba ukljuciti u sklopu unit testa. Zato je potrebno kreirati mock objekat prilikom testiranja koji ce zameniti stvarnu konekciju mock-ovanom.
* Test folder poseduje
  + GmockDB.h – mock klasa menja originalnu klasu DataBaseConnect, koju koristi komponenta cije ponasanje testiramo. Fakticki ova klasa ce vrsiti simulaciju konekcije i zato je neophodno da nasledjuje klasu DataBaseConnect.
  + gmock\_test.cpp fajl u kome se vrsi testiranje. U njemu je nophodno da prilikom instanciranja MyDataBaase objekta umesto originalne konekcije prosledimo MockDB objekat koji predstavlja njenu simulaciju.

Kratak tutorijal za gmock i gtest (part1-part4) <https://www.youtube.com/watch?v=nbFXI9SDfbk&ab_channel=CodeSports>

Dodatne informacije u vezi gmock-a

https://chromium.googlesource.com/external/github.com/google/googletest/+/refs/tags/release-1.8.0/googlemock/docs/ForDummies.md