**Task:**

Implement a test case that increments variable by 10, 20,100 and checks if the updated values are correct.

**TEST.CPP:**

#include "pch.h"

#include "gtest/gtest.h"

#include "C:\Users\TEMP\Desktop\fixture\fixture\fix.cpp"

using namespace testing;

class calTest :public Test {

protected:

void SetUp() override{

xyz = new cal();}

void TearDown()override {

delete xyz;}

cal\* xyz;};

TEST\_F(calTest, addition) {EXPECT\_EQ(xyz->add(25, 25), 50);

EXPECT\_EQ(xyz->add(25, -25), 0);}

TEST\_F(calTest, increment) {EXPECT\_EQ(xyz->inc(0), 10);

EXPECT\_EQ(xyz->inc(10), 20);

EXPECT\_EQ(xyz->inc(90), 100);}

**Fix.cpp:**

#include "fix.h"

double cal::add(double a, double b) {

return a + b;}

double cal::mul(double a, double b)

{return a\*b;}

double cal::inc(double b) {

int a = 10;

return a + b;}

**Fix.h:**

#pragma once

class cal {

public:

double add(double a, double b);

double mul(double a, double b);

double inc(double a);};

**Fixture.cpp:**

#include <iostream>

using namespace std;

#include "fix.h"

int main(){ cal xyz;

cout << xyz.add(3, 2)<<endl;

cout << xyz.mul(3, 4)<<endl;

cout << xyz.inc(2) << endl;}

