

Unit Testing

Testing Levels

01

Unit Testing

By Developers

...

02

Integration Testing

By Testers

...

03

System Testing

By Testers

...

04

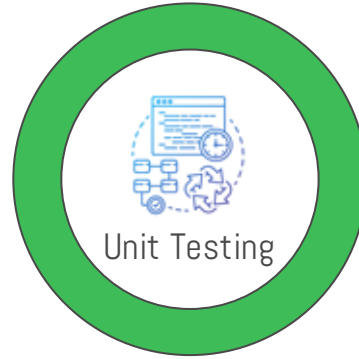
Acceptance Testing

By End User

...



Unit Testing



What?

Software development process in which the smallest testable parts of an application, called units, are individually and independently scrutinized for proper operation. This testing methodology is done during the development process by the software developers and sometimes QA staff.

...



REPRESENT AS

01 functions



02 Reports



03 Form



04 Sql Query



Unit Testing

- **When ?**

After modules are coded.

- **Who ?**

Developers.

Unit Testing

- Faster Debugging
- Easier to fix bugs and pickup earlier
- Unit tests smaller amount of code
- Encouraging more refactoring
- Reduce future cost

Unit Testing Life Cycle

Internal Application Design



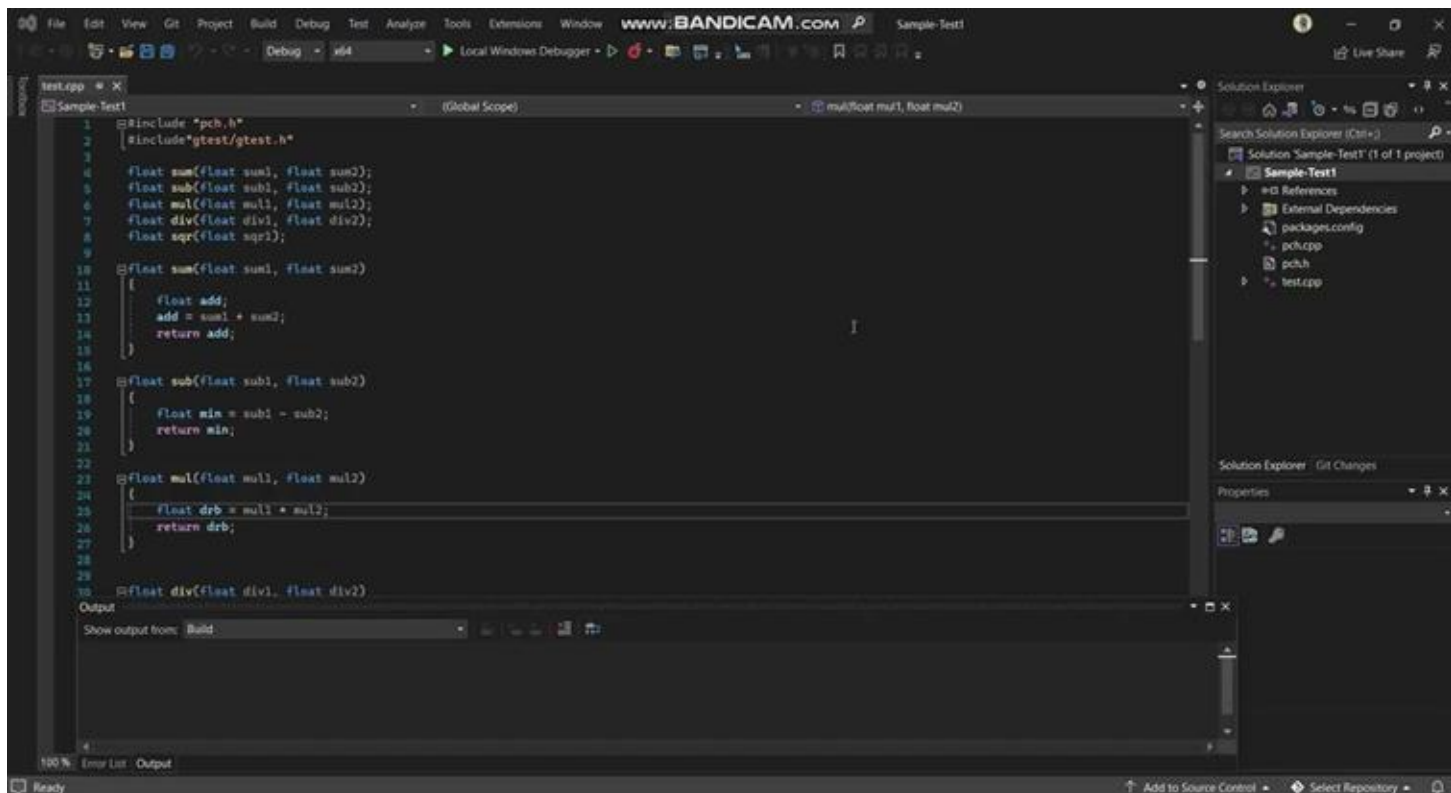
Unit Test Report

Tools

- Debug
- Re-structure
- Code Analyzer
- Path/statement coverage tools



Demo Unit Testing



```
test.cpp
1 #include "pch.h"
2 #include "gtest/gtest.h"
3
4 float sum(float sum1, float sum2);
5 float sub(float sub1, float sub2);
6 float mul(float mul1, float mul2);
7 float div(float div1, float div2);
8 float sqr(float nqr1);
9
10 float sum(float sum1, float sum2)
11 {
12     float add;
13     add = sum1 + sum2;
14     return add;
15 }
16
17 float sub(float sub1, float sub2)
18 {
19     float min = sub1 - sub2;
20     return min;
21 }
22
23 float mul(float mul1, float mul2)
24 {
25     float drb = mul1 * mul2;
26     return drb;
27 }
28
29 float div(float div1, float div2)
```

Output

Show output from: Build

100% Error List Output

Ready Add to Source Control Select Repository



THANKS!