



Testing Driven Development (TDD)

Squashing software bugs

Gregory Green & Morgan Iverson



<https://via.vmw.com/bhm-tdd>

Zoom

Look to the options at the bottom of your Zoom Window:

- Q&A - type any questions
- Chat - ways to give feedback

Zoom



Who are we?

Introductions



Morgan Iverson
Software Engineer



Gregory Green
Solution Engineer



<https://via.vmw.com/bhm-tdd>

Agenda

Traditional Software Testing Methods

- Challenges

What is Test Driven Development (TDD)

- How does it help?

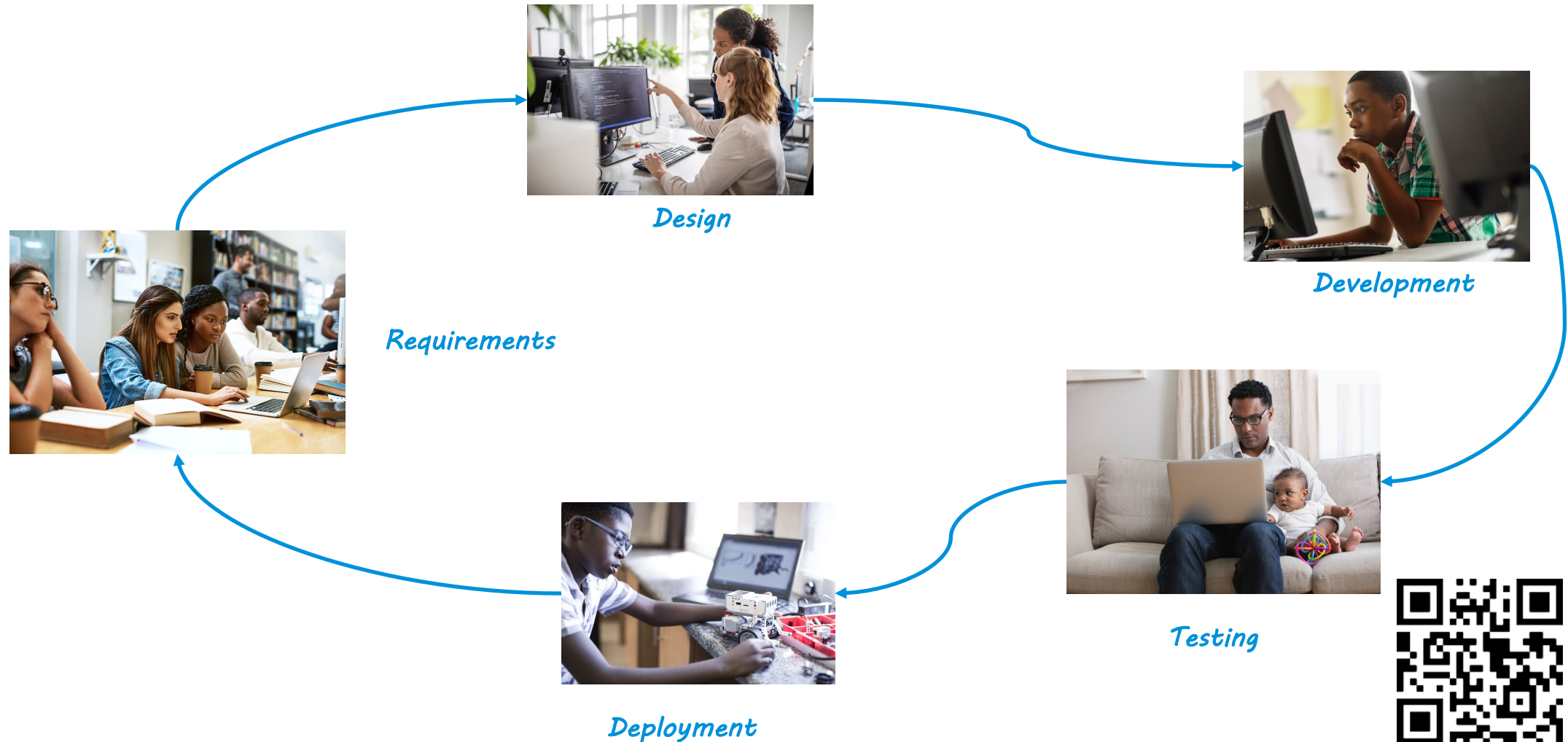
Example – Workshop

Q&A



<https://via.vmw.com/bhm-tdd>

Traditional Software Testing Methods



<https://via.vmw.com/bhm-tdd>

Traditional Software Testing Methods - challenges

- Lots of Manually testing
- Needed lots of testing resources
- Potentially low test-coverage (is each block of code checked)
- Takes more time to test with more and more code
- Time to market – long time between an idea (requirement) to deployment



Not instrumented →

Covered by tests →

Not covered by tests →

```
516 INSTANTIATE_TEST_SUITE_P(NativeSequenced,
517                             ThreadPoolWorkerPoolTest,
518                             ::testing::Values(PoolExecution
519                             test::PoolType::NATIVE,
520                             test::ExecutionMode::SEQU
521                             #endif
522
523 TEST(TaskSchedulerWorkerPoolTest, TestCodeCoverage) {
524     bool flag = true;
525     if (!flag) {
526         int value = 10;
527         EXPECT_EQ(10, value);
528     }
529     EXPECT_TRUE(flag);
530 }
```



<https://via.vmw.com/bhm-tdd>

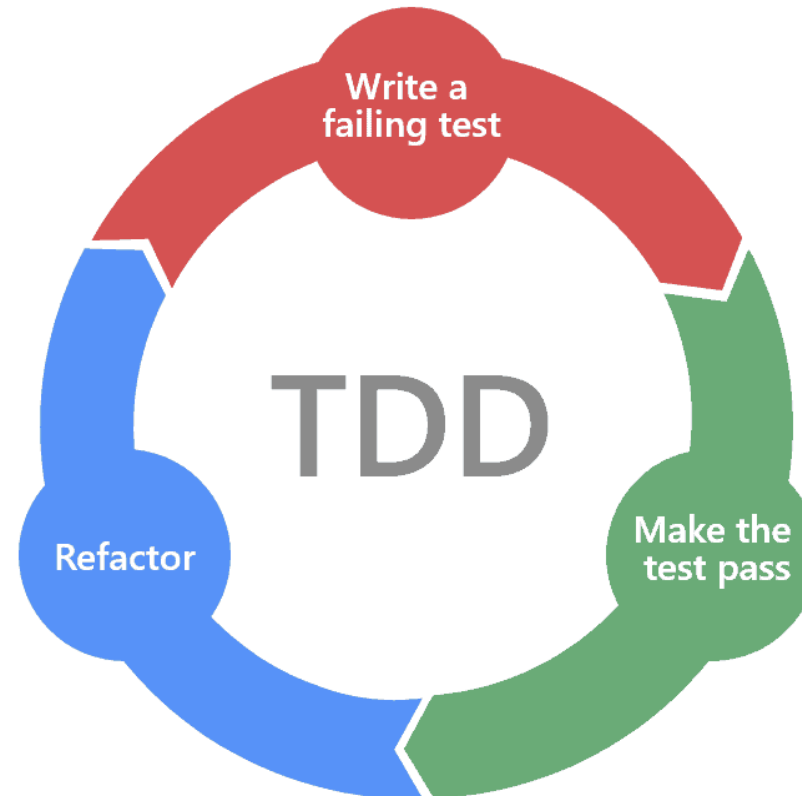
What is Test Driven Development

Test-driven development

From Wikipedia, the free encyclopedia

Test-driven development (TDD) is a [software development process](#) relying on software requirements being converted to [test cases](#) before software is fully developed, and tracking all software development by repeatedly testing the software against all test cases. This is opposed to software being developed first and test cases created later.

- Write code to test code first
- Start with writing code first
- The test code drives the requirements and design
- Paired programming
 - “Two are better than one”



<https://via.vmw.com/bhm-tdd>

How does TDD help?

- Automated testing
- Minimize manual people processing
- Improve software quality with test-coverage
- Its less to test with more and more code
- Reduces time to market from idea (requirement) to deployment



<https://via.vmw.com/bhm-tdd>

Basics - Computer Language is English

Programming Language

English

1. Noun
2. Verb

Computer Language: Kotlin

1. Class
2. Methods/Functions

Game starts
Player wins



<https://via.vmw.com/bhm-tdd>

Given-When-Then TDD Test

Essential idea is to break down writing a scenario (or test) into sections

Feature: description of a software feature

Scenario: Example or outline of test

GIVEN

- Describes the state of the **class** before you begin. You can think of it as the pre-conditions to the test. This part can be optional.

WHEN

- The **when** section can be **method/function** that your testing.

THEN

- Finally the **then** section describes the changes you expect due to the specified behavior.

Feature <description>

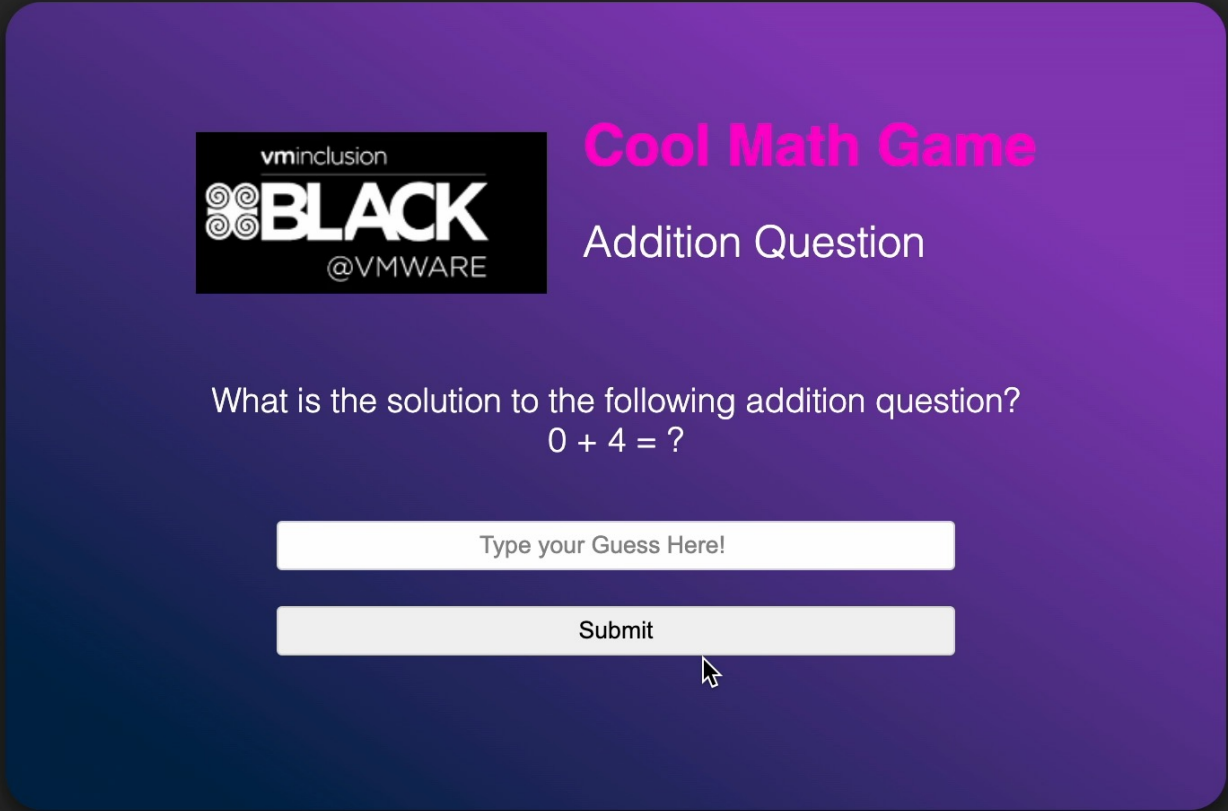
Scenario: <test>

GIVEN <Noun-Class>
description

WHEN <Verb-Method-
Function> description

THEN expected outcome

Our Cool Math Game



The image shows a web-based math game interface. It features a dark purple gradient background. In the top left corner, there is a logo for 'vm inclusion BLACK @VMWARE'. To the right of the logo, the title 'Cool Math Game' is displayed in a bright pink color, followed by 'Addition Question' in white. The main text asks, 'What is the solution to the following addition question?' followed by the equation '0 + 4 = ?'. Below this, there is a white input field with the placeholder text 'Type your Guess Here!'. Underneath the input field is a grey 'Submit' button, which is being pointed to by a mouse cursor.

vm inclusion
BLACK
@VMWARE

Cool Math Game

Addition Question

What is the solution to the following addition question?
 $0 + 4 = ?$

Submit

Our Development

Assurance Tools

Development



IntelliJ IDEA

Unit Testing

JUnit

Building Tools



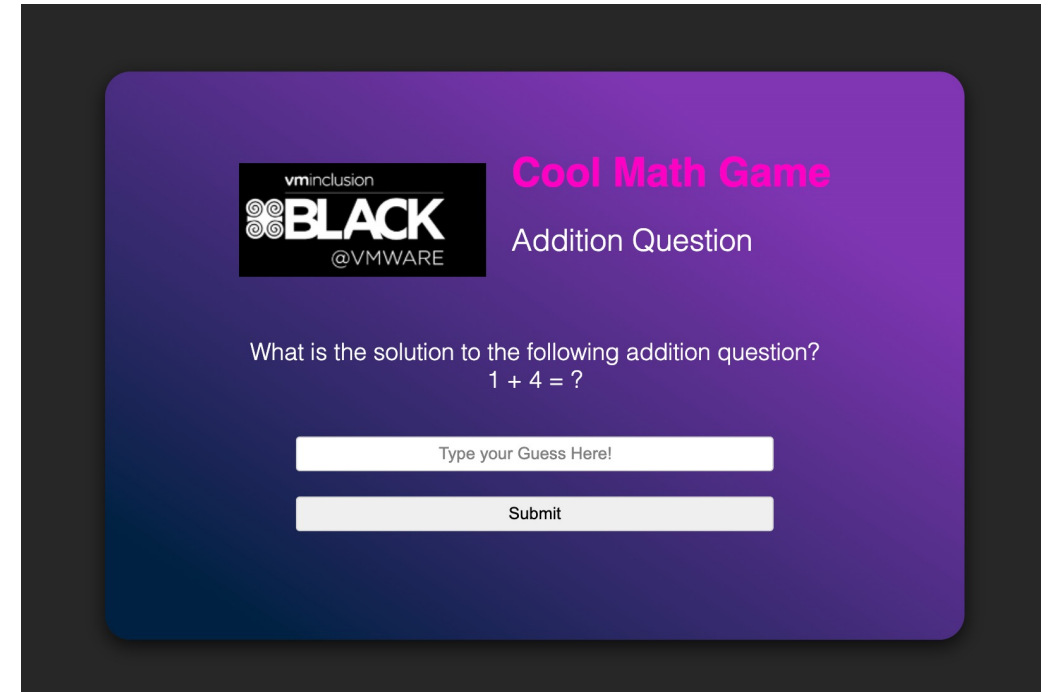
<https://via.vmw.com/bhm-tdd>

Test Driven Case 1

Feature: Addition Question

Scenario: Guess the correct answer

GIVEN
WHEN
THEN



The screenshot shows a web interface for a 'Cool Math Game'. At the top left is a logo for 'vminclusion BLACK @VMWARE'. To the right of the logo, the text 'Cool Math Game' is displayed in pink, and 'Addition Question' is displayed in white. Below this, a question is posed: 'What is the solution to the following addition question?' followed by the equation $1 + 4 = ?$. There are two input fields: the first is labeled 'Type your Guess Here!' and the second is labeled 'Submit'.



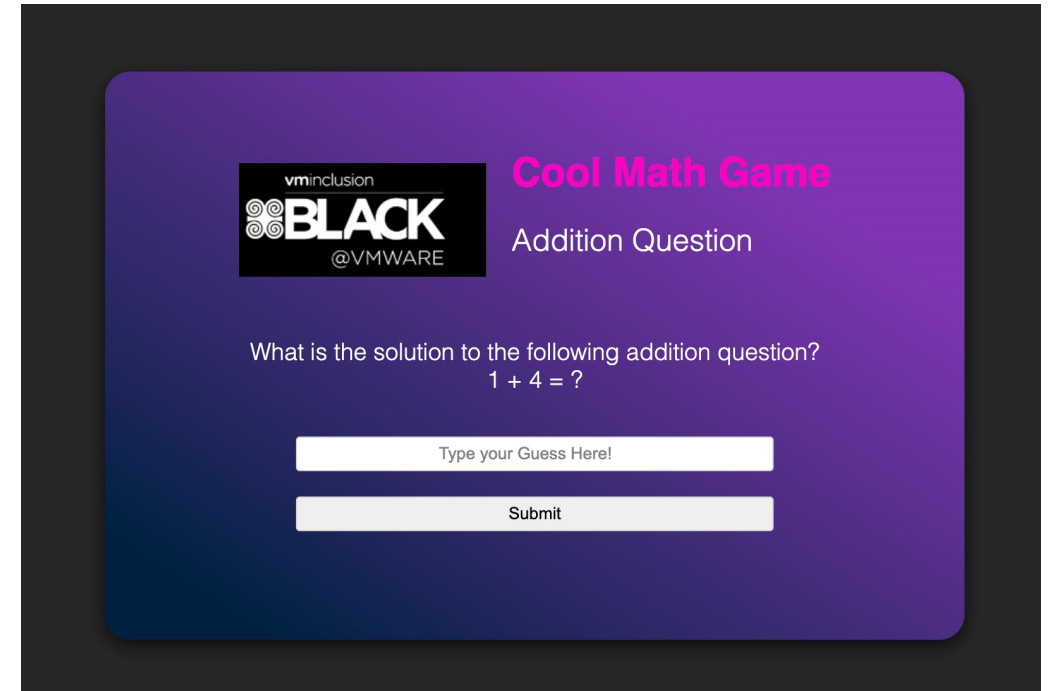
<https://via.vmw.com/bhm-tdd>

Test Driven Case 1

Feature: Addition Question

Scenario: Guess the correct answer

GIVEN Addition Question with 1 and 2
WHEN
THEN



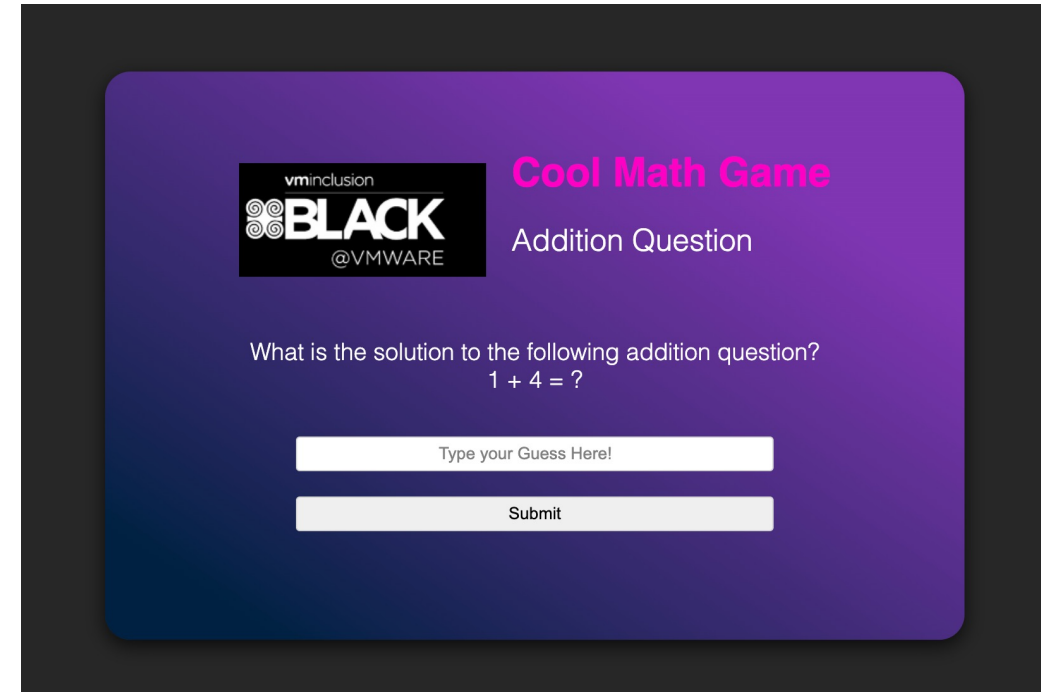
<https://via.vmw.com/bhm-tdd>

Test Driven Case 1

Feature: Addition Question

Scenario: Guess the correct answer

GIVEN Addition Question with 1 and 2
WHEN Is Guess Correct given 3
THEN



<https://via.vmw.com/bhm-tdd>

Test Driven Case 1

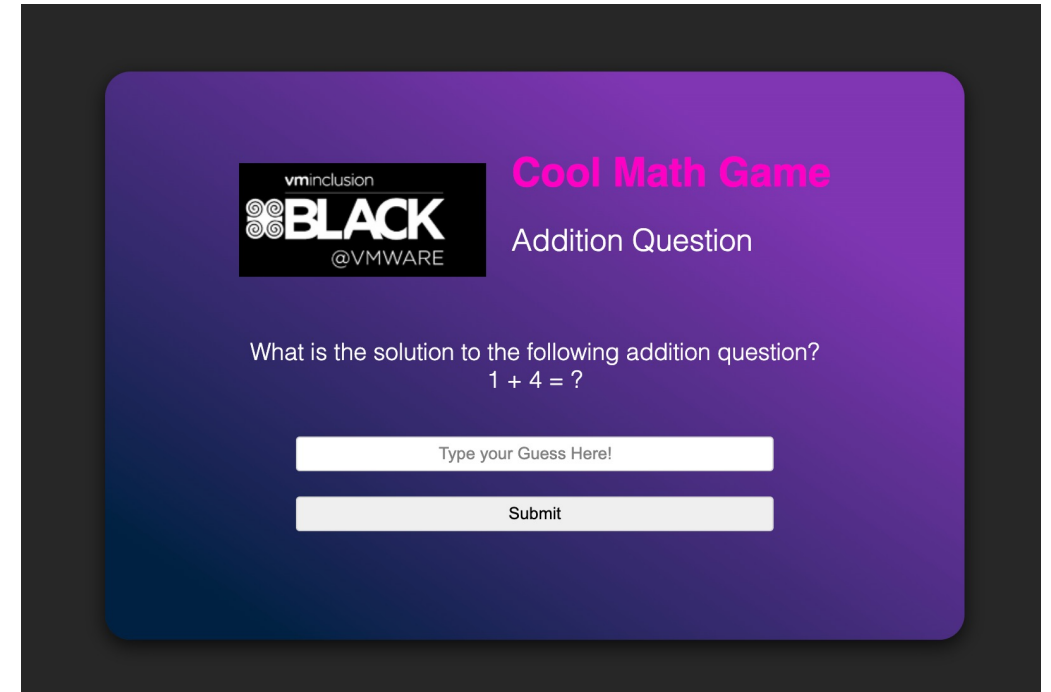
Feature: Addition Question

Scenario: Guess the correct answer

GIVEN Addition Question with 1 and 2

WHEN Is Guess Correct given 3

THEN Returns true



<https://via.vmw.com/bhm-tdd>

THANK YOU