



# Agenda

---

- Vorstellungsrunde
- Administratives
- Einstieg in Software-Testing
- Ursachenkette für Fehler
- Hands on «Unit Test»
- Unit-Testing Frameworks

# Mein Name ist...

---

- Michael Keiser
  - Dipl. Techniker HF Softwareentwicklung
  - Dipl. Betriebswirtschafter HF NDS
  - [michi.keiser@gmail.com](mailto:michi.keiser@gmail.com)
- Hobbies
  - Meine zwei Katzen
  - Systemtechnik
  - Schrauben in der Werkstatt
  - Autos



# Ich arbeite bei...

---



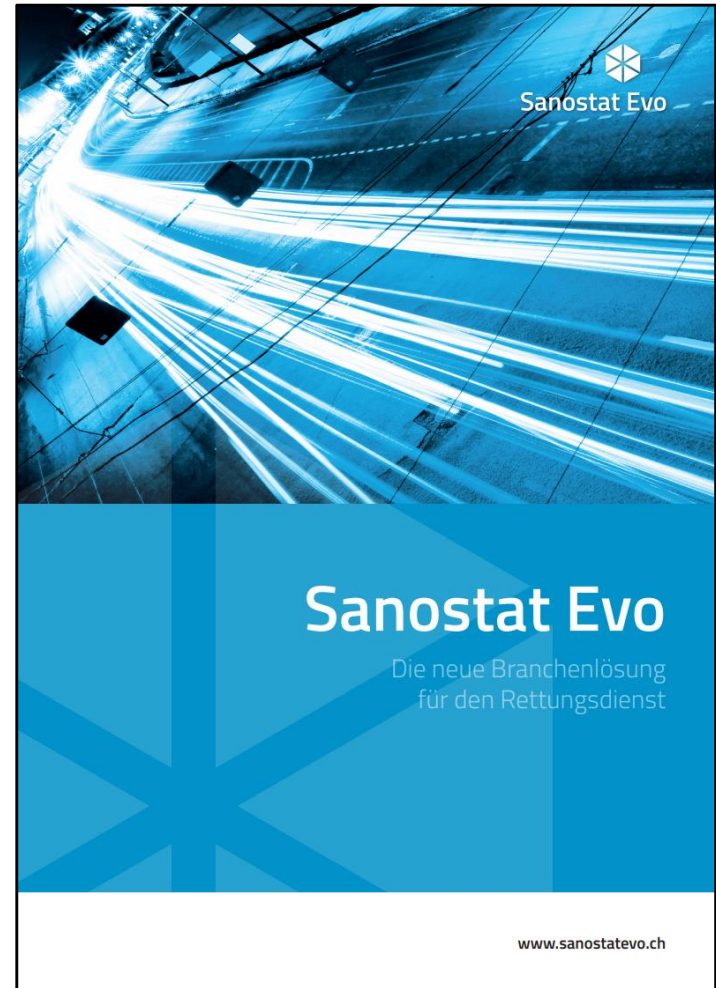
# BST

## Informatik GmbH

## Meine Funktionen sind...

- Stv. Geschäftsführer
- Product Owner
- CTO

# Wir entwickeln...



# Meine Erwartungen an die Studierenden...

---

- aktive Mitarbeit im Unterricht
- Stellen Sie Fragen!
- Seid nicht voreingenommen vom Thema «Testing»
- Ich erwarte, dass Sie die Hausaufgaben erledigen

# Nun seid ihr am Zug...

---

- Vorname / Nachname
- Firma
- Aufgabe in der Firma
- Hobbies
- Erwartungen an diesen Kurs
- Berührungspunkte mit «Softwaretesting»?

# Administratives

---

- Sie oder du?
- Pausen
- Datenaustausch
  - GitHub
  - Moodle
- Testat



# Was lernen wir hier? (1)

---



Bildquelle: <http://www.bz-mg.de/politik-verwaltung-parteien/die-ampel/neubau-zentralbibliothek-lernen-ist-ein-prozess-der-relativ-stabilen-veraenderung-des-verhaltens-fehlanzeige-bei-der-ampel.html>

# Was lernen wir hier? (2)

---

- Grundverständnis
- Die gleiche Sprache sprechen
- Qualität messen
- Code Metriken
- Prozesse zur Qualitätssicherung
- Testautomatisierung

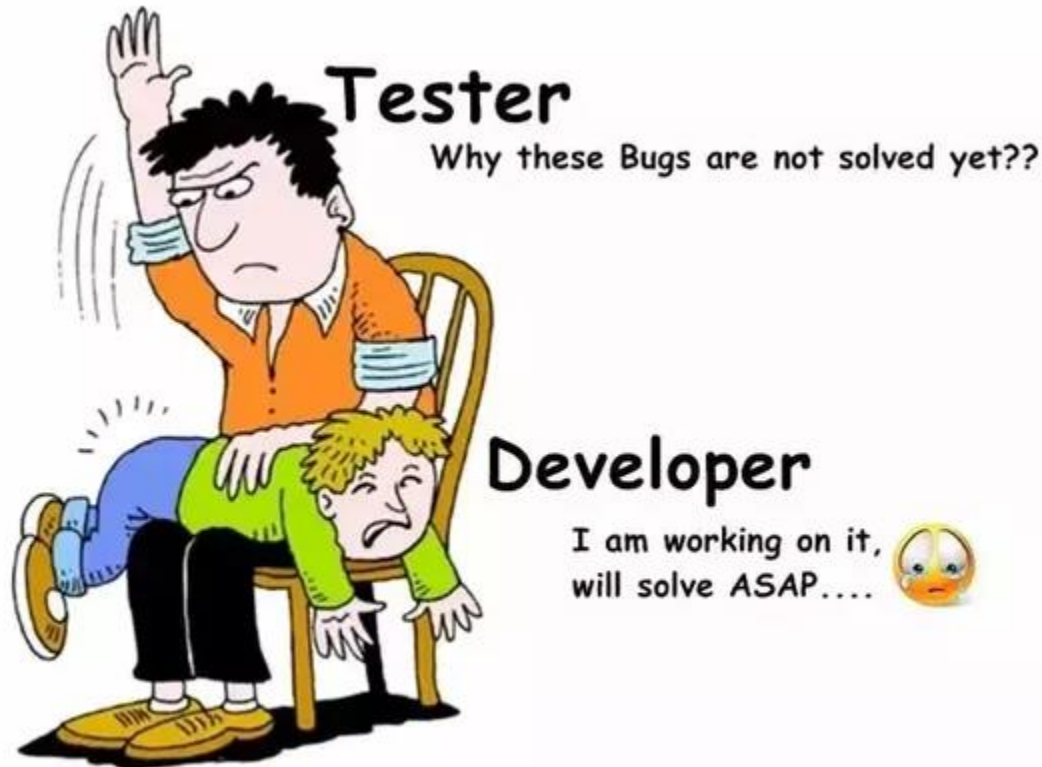
# Grundverständnis (1)

---



Bildquelle: <http://www.dselva.co.in/independent-software-testing/>  
<https://www.edx.org/course/software-testing-management-usmx-umuc-stv1-2x-1>

# Grundverständnis (2)



Bildquelle: <https://www.quora.com/How-can-I-get-into-software-testing-in-3-months>

# Die gleiche Sprache sprechen...



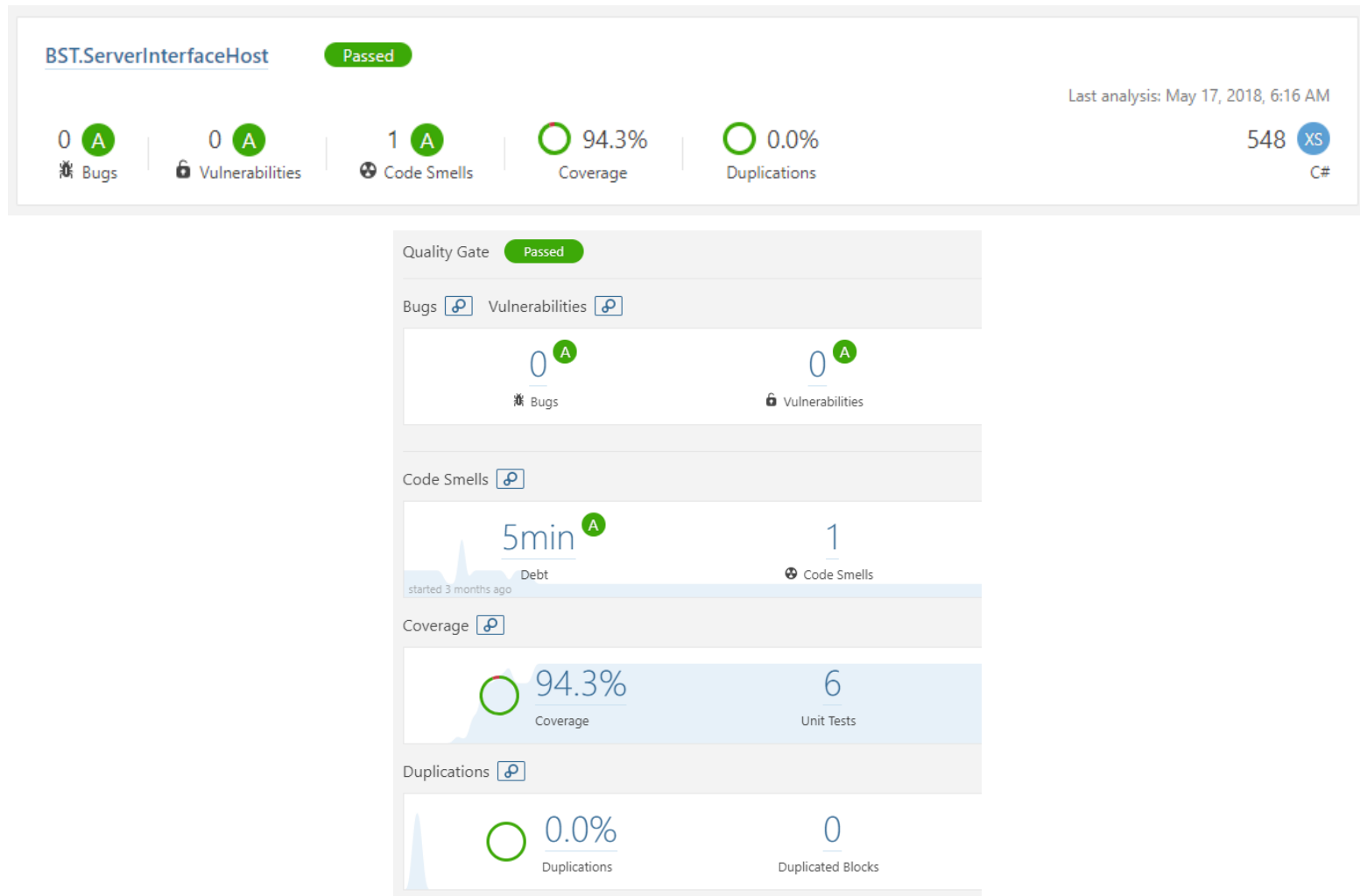
Bildquelle: <https://statistik.tg.ch/themen-und-daten/bevoelkerung-und-haushalte/bevoelkerungsstand-und-struktur/sprachen.html/7035>

# Was ist Softwaretesting?




Bildquelle: <http://tip-web.de/de/about/qualitaet>

# Code Metriken






# Prozesse zur Qualitätssicherung

▼  BST.ServerInterfaceHost | ▼

☰ Start (Manual Productive) | ▼


No builds to display

▼  Start (Manual Test) | ▼

develop

#0.0.0.112 (Alpha)


✔ Build chain finished (success: 4); Tests passed: 6 | ▼

▼  Start (Nightly) | ▼

develop

#0.0.0.114 (Alpha)

✔ Build chain finished (success: 4); Tests passed: 6 | ▼

▼  Start (VCS-Trigger) | ▼

develop

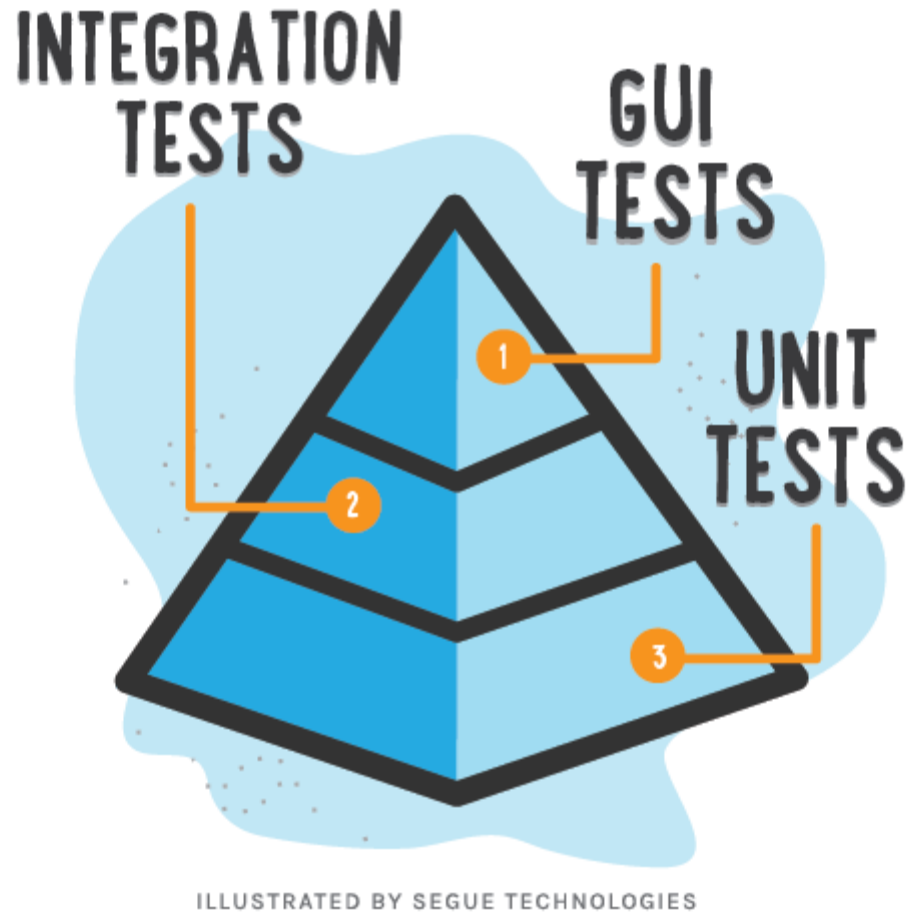
#0.0.0.113 (Alpha)

✔ Build chain finished (success: 4); Tests passed: 6 | ▼



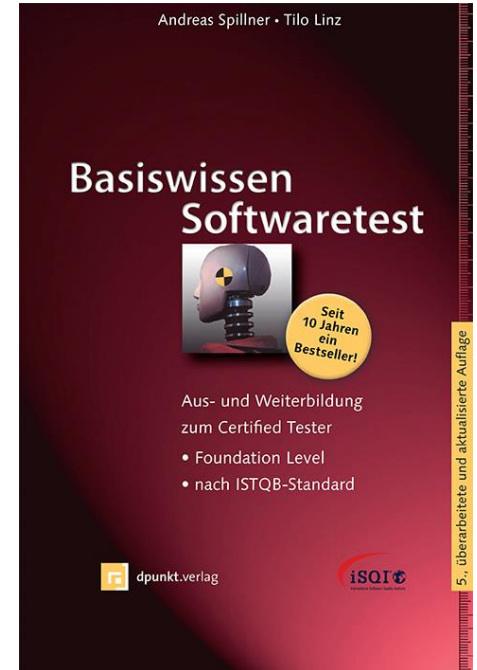
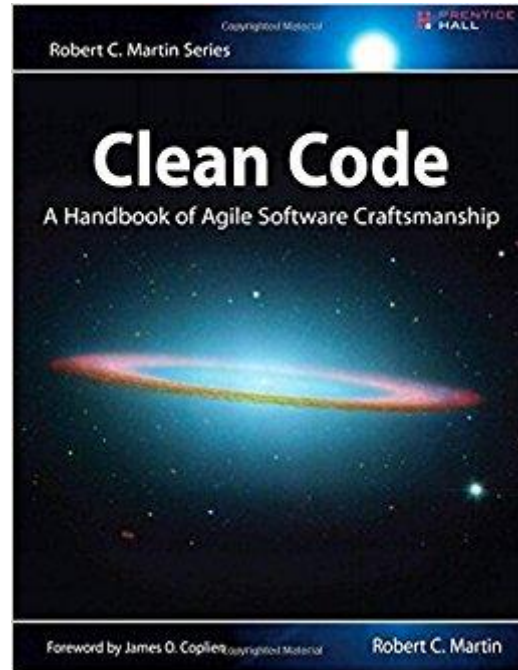
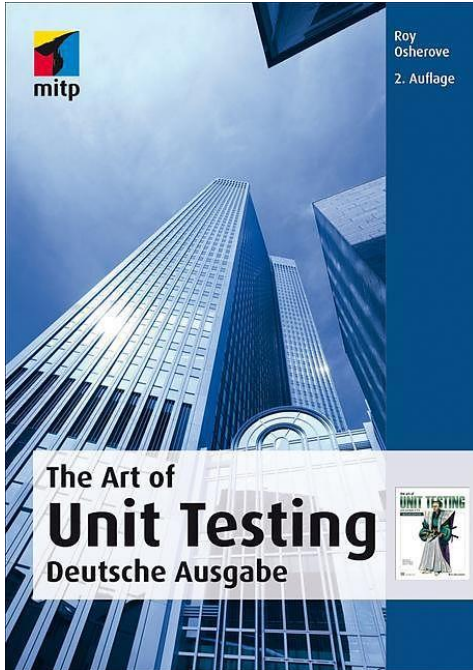
# Testautomatisierung

---



Bildquelle: <https://www.seguetech.com/the-benefits-of-unit-testing/>

# Lehrmittel



# Einstieg in Software-Testing (1)

---



Bildquelle: <https://sii.pl/blog/wie-wird-man-tester-ein-kurzer-ratgeber-zur-softwaretesting-thematik-2/>

# Einstieg in Software-Testing (2)

---

- Typische Begriffe aus dem SW-Testing (Auswahl)
  - Komponententests
  - Integrationstests
  - Systemtests
  - Abnahmetests
  - Testspezifikation
  - Testorakel
  - White- und Blackbox-Testing
  - usw.

# Softwarefehler und ihre Folgen... (1)

---



<https://www.unrealengine.com/marketplace/the-explosions-mega-pack>

# Softwarefehler und ihre Folgen... (2)



Bildquelle: <http://www.hdbilder.eu/bild/201969/stuxnet-iranische-atom-computer-politische-anarchie-windows-microsoft-cyber-%E2%80%8B%E2%80%8Bhacker>  
 Weitere Infos: <https://de.wikipedia.org/wiki/Stuxnet>

# Softwarefehler und ihre Folgen... (3)

---



Bildquelle: <http://www.amatteroflifeandtech.com/blog/tech-life/the-heartbleed-bug-explained/>  
Weitere Infos: [https://de.wikipedia.org/wiki/Liste\\_von\\_Programmfehlerbeispielen](https://de.wikipedia.org/wiki/Liste_von_Programmfehlerbeispielen)



# Wie entsteht ein Bug?




Photo # NH 96566-KN First Computer "Bug", 1945

9/2

9/9

0800 Antam started  
1000 " stopped - antam ✓  
1300 (032) MP-MC 1.58264000 9.037 847 025  
(033) PRO 2 2.130476415 9.037 846 995 correct  
2.130476415 4.615925059(-2)  
2.130476415  
2.130676415  
Relays 6-2 in 033 failed special speed test  
in relay "on test".  
Relays changed

1100 Started Cosine Tape (Sine check)  
1525 Started Multi-Adder Test.

1545  Relay #70 Panel F  
(moth) in relay.

First actual case of bug being found.

1650 Antam started.  
1700 closed down.

Relay 3145  
Relay 3371

Bildquelle: <https://online-marketing-news.ch/bloggen/so-finden-sie-ideen-fur-neuen-content/> [https://www.aref.de/kalenderblatt/2005/36\\_computer-bug.php](https://www.aref.de/kalenderblatt/2005/36_computer-bug.php)



# Ursachenkette für Fehler (1)

---

## Schwärzen

«Die Teilnehmer lesen Texte und haben die Aufgabe, alle Informationen, die ihnen nicht wichtig erscheinen, mit einem schwarzen Filzstift wegzustreichen. Anschliessend Austauschen und Diskussion.»

**Arbeitsform:** Einzelarbeit

**Methodik:** Schwärzen

**Text:** Wird ausgeteilt

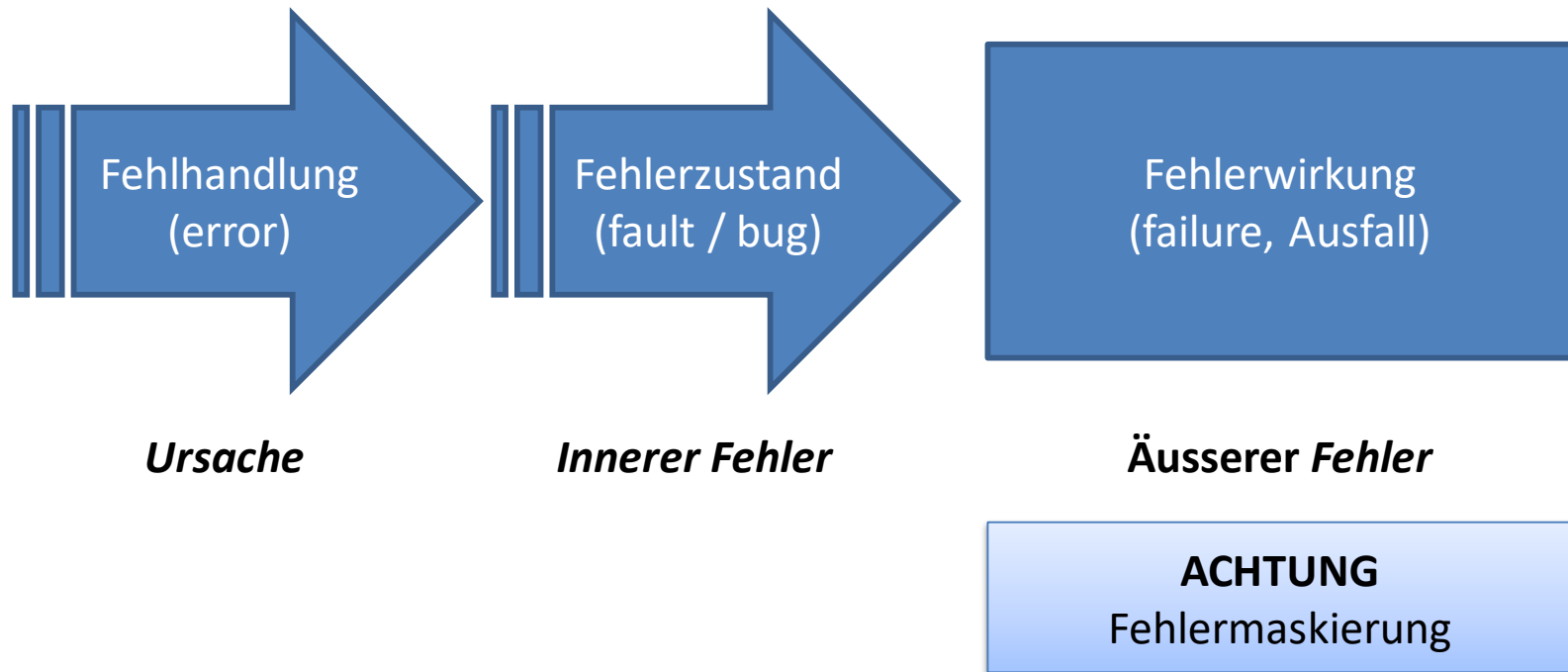
**Hilfsmittel:** Schwarzer Stift

**Zeit:** 15 Minuten

**Besprechung / Feedback:** In der Klasse

Textquelle: Munterrichtsmethoden von Schilling Verlag (Gross, Boden, Boden), 2012

# Ursachenkette für Fehler (2)



Buch / S. 7

# Let's code!

---

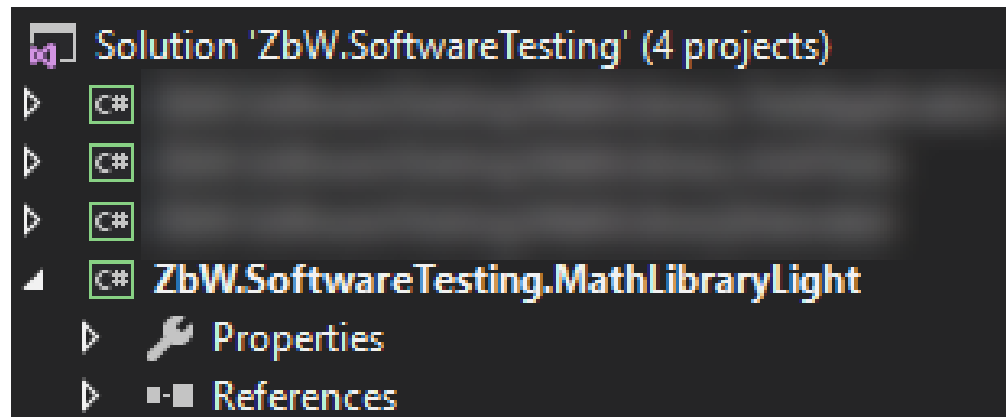


Bildquelle: <http://www.loslegen.net/los-gehts/>

# Hands on: Math-Library «light» (1)

---

Erstellen Sie eine neue Solution und eine neue Class Library



# Hands on: Math-Library «light» (2)

---

Erstellen Sie eine «Checked Exception»

```
using System;  
2 references  
public class ZeroDivisorNotAllowedException : Exception  
{  
}
```

# Hands on: Math-Library «light» (3)

```
public class BasicOperation
{
    1 reference
    public int Addition(int firstAddend, int secondAddend)
    {
        var sum = firstAddend + secondAddend;
        return sum;
    }

    1 reference
    public int Subtraction(int minuend, int subtrahend)
    {
        var difference = minuend - subtrahend;
        return difference;
    }

    2 references
    public double Division(int dividend, int divisor)
    {
        if (divisor == 0)
        {
            throw new ZeroDivisorNotAllowedException();
        }

        var quotient = (double)dividend / divisor;
        return quotient;
    }

    1 reference
    public int Multiplication(int multiplier, int multiplicand)
    {
        var product = multiplier * multiplicand;
        return product;
    }
}
```

Erstellen Sie die Klasse «BasicOperation» und programmieren Sie die Methoden.

Feuern Sie die ZeroDivisorNotAllowedException wenn eine Division durch «0» ausgeführt werden soll

**Arbeitsform:** Einzelarbeit  
**Zeit (Vorbereitung):** 10 Minuten  
**Besprechung / Feedback:** In der Klasse

# Bugs?

---



Bildquelle: <http://www.mokacoding.com/blog/fixing-bugs-driven-by-tests-in-swift/>



# Blindflug oder?

---



Bildquelle: <http://www.fotocommunity.de/photo/blindflug-klaus-gierden/24119292>



# Besser wäre doch...

---

```
#####  
# Hi, I'm your Test Application for MathLibrary Light #  
#####  
  
Try to test Method 'Addition'  
Result: '2 + 3 = 5' is correct  
--> Passed  
  
Try to test Method 'Subtraction'  
Result: '20 - 5 = 15' is correct  
--> Passed  
  
Try to test Method 'Division'  
Result: '100 / 5 = 20' is correct  
--> Passed  
  
Try to test Method 'Division' by zero  
--> Passed  
  
Try to test Method 'Multiplication'  
Result: '5 * 10 = 50' is correct  
--> Passed  
  
Press any key to close...
```

# Oder eben...

---

```
#####  
# Hi, I'm your Test Application for MathLibrary Light #  
#####  
  
Try to test Method 'Addition'  
Result: '2 + 3 = 6' is wrong  
--> Failed!!!  
  
Try to test Method 'Subtraction'  
Result: '20 - 5 = 16' is wrong  
--> Failed!!!  
  
Try to test Method 'Division'  
Result: '100 / 5 = 21' is wrong  
--> Failed!!!  
  
Try to test Method 'Division' by zero  
--> Failed!!!  
  
Try to test Method 'Multiplication'  
Result: '5 * 10 = 51' is wrong  
--> Failed!!!  
  
Press any key to close...
```

# Mit oder ohne... Wie schläft ihr besser?

---



Bildquelle: <https://www.bodyteam.ch/gesundheit-winterthur/sieben-tipps-die-sie-besser-schlafen-lassen/>

# Unit Testing Framework – do it yourself!

---

- Erstellen Sie eine Konsolenapplikation mit welcher Sie Ihre Math-Library «light» testen.

**ZbW.SoftwareTesting.MathLibraryLight.ConsoleTests**

**Arbeitsform:** Hands on

**Zeit (Vorbereitung):** 40 Minuten

**Besprechung / Feedback:** In der Klasse

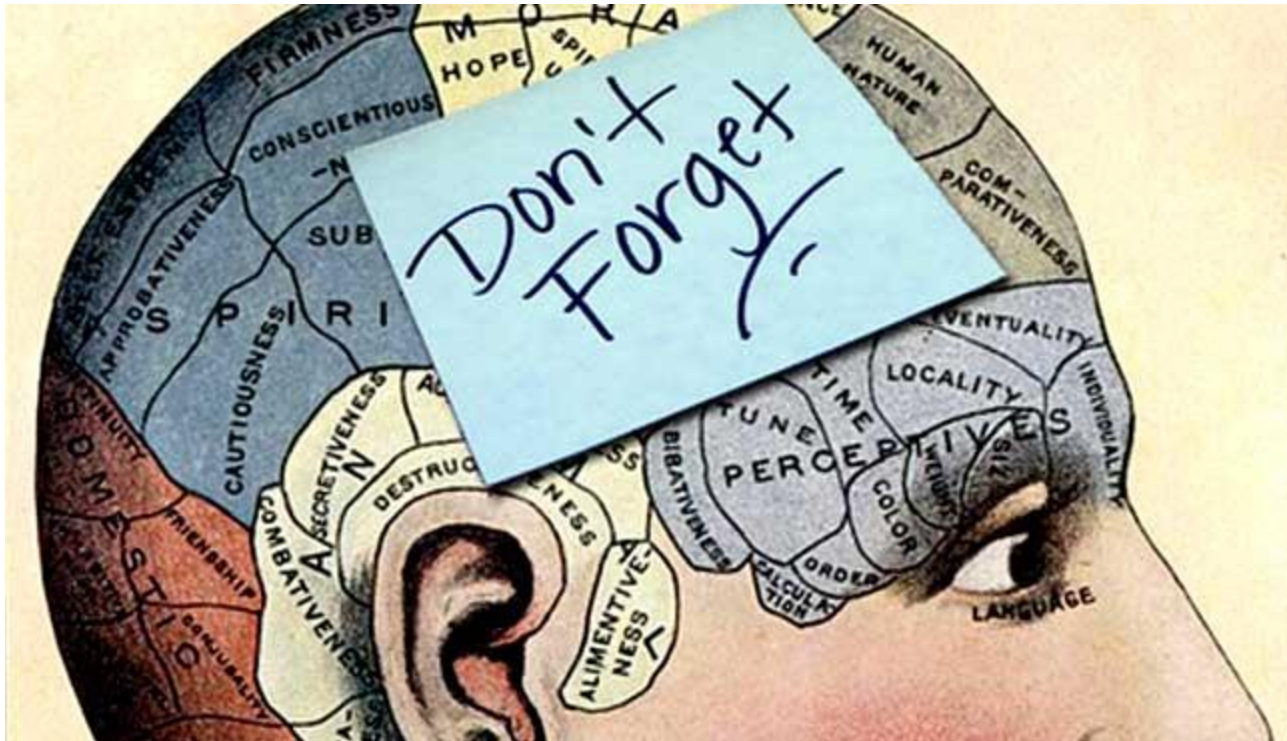
# Geht das nicht einfacher?

---



Bildquelle: [https://de.wikipedia.org/wiki/Ja!\\_Nat%C3%BCrlich](https://de.wikipedia.org/wiki/Ja!_Nat%C3%BCrlich)

# Was nehmen wir vom Unterricht mit?



Bildquelle: <http://floraremedia.com/keep-mind-memory-sharp/>

# Hausaufgaben

---

- Kapitel 2 (S. 5 - 40) durchlesen
- JetBrains ReSharper installieren
- JetBrains dotCover installieren
- (optional) Buch «The Art of UnitTesting» kaufen