Software Testing

ONELITY

Περιεχόμενα



- Software Development Lifecycle Models
- Λόγοι διεξαγωγής των Test
- Reviews
- Test Activities
- Το επάγγελμα του Software Tester / QA
- Χαρακτηριστικά των Testers
- Test Levels / Types / Techniques
- Οι 7 αρχές του Test
- Η εταιρεία ONELITY

Ο συναρπαστικός κόσμος του Software Testing



ONELITY

Ομιλητής: Ζηκίδης Δημήτρης

Εμπειρία: 20 χρόνια ως Software Tester & QA

Σπουδές: Ηλεκτρονικός Μηχανικός

Εργασία: Ηλεκτρονικός Μηχανικός, Developer, Software Tester, Software Quality Assurance Engineer

Γλώσσες προγρ.: C / C++, Python, Java, Visual Basic



01

Foundation Level

ISTQB®
International Software Testing Qualifications Board

02

Test Automation Engineer

03

Advanced Level Technical Test Analyst





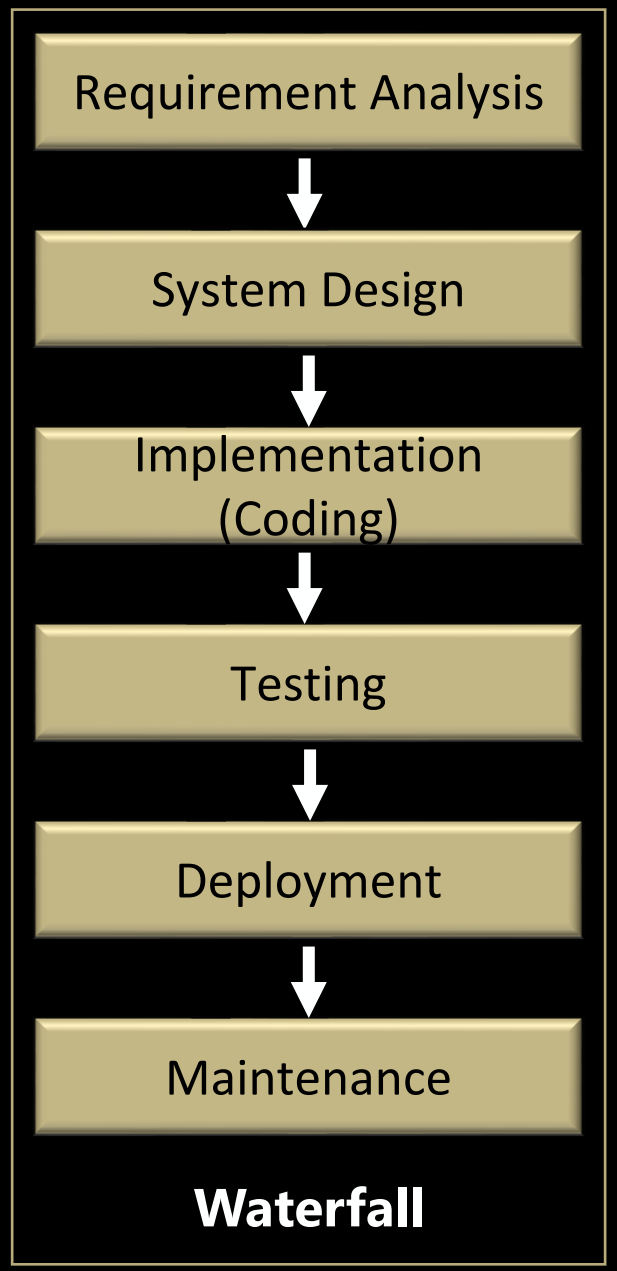
Γιατί διεξάγουμε τα Test;

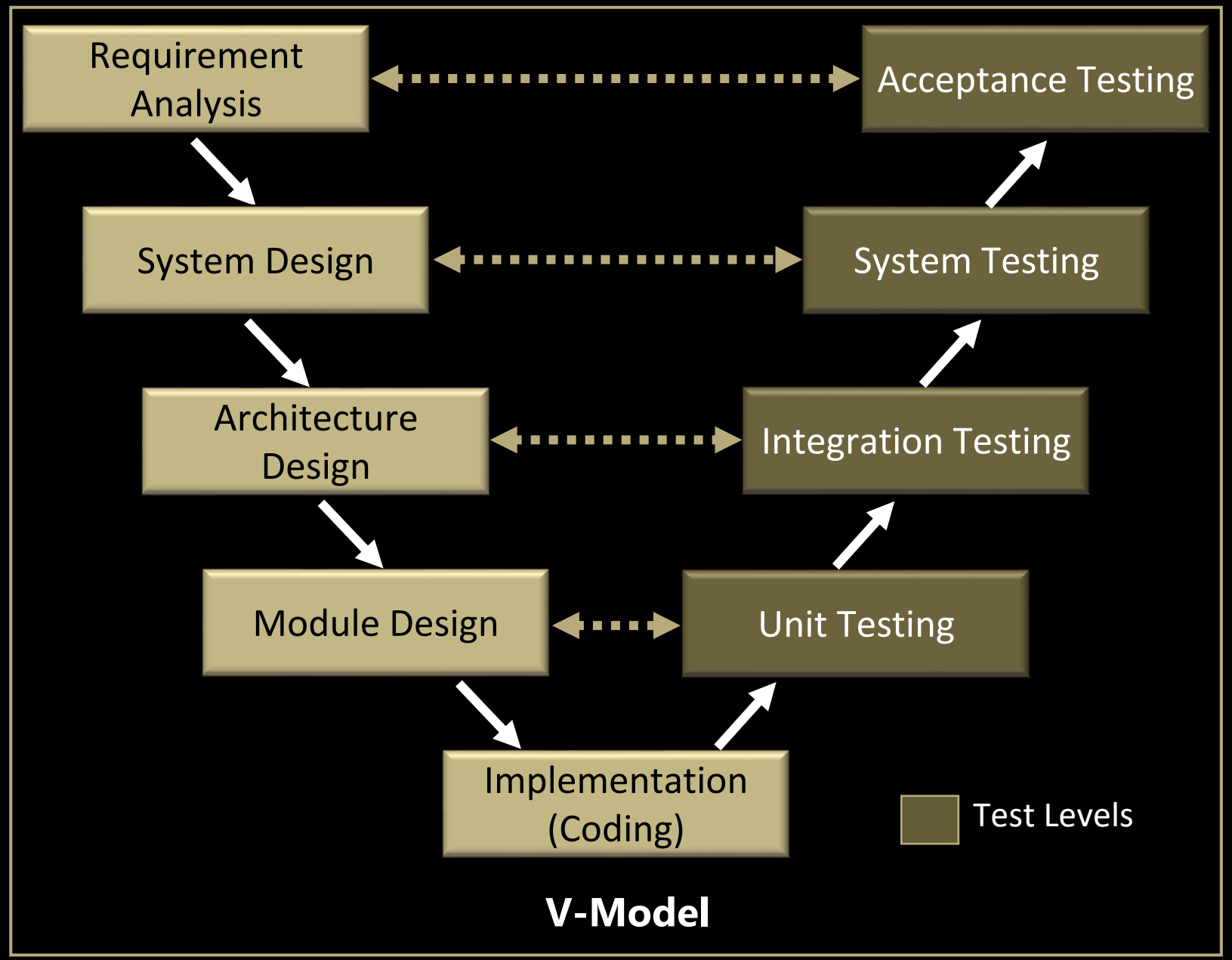
SDLC Models

Software Development Life Cycle Models

1	**Sequential**
2	**Iterative - Incremental**

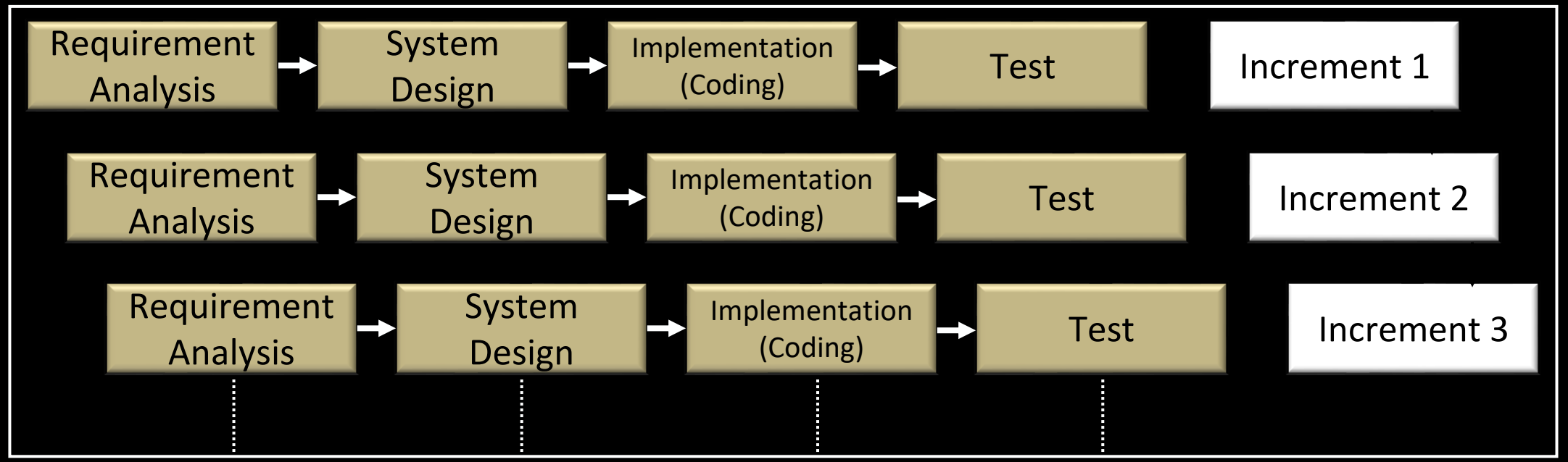
Sequential


Requirement Analysis
System Design
Implementation (Coding)
Testing
Deployment
Maintenance
Waterfall


Requirement Analysis
Acceptance Testing
System Design
System Testing
Architecture Design
Integration Testing
Module Design
Unit Testing
Implementation (Coding)
Test Levels
V-Model

Iterative - Incremental

Agile	• **Scrum** • **Kanban**
	• **Rational Unified Process (RUP)**



Reviews

- Informal
- Walkthrough
- Technical
- Inspection

Types

- Ad-Hoc
- Checklist Based
- Scenarios & Dry Runs
- Perspective Based
- Role Based

Techniques

Test Activities

Activity	Actions
Test planning	test objectives, test techniques, test schedule
Test monitoring and control	actual progress against planned progress, actions to meet objectives
Test analysis	what to test
Test design	high level test cases, priorities, identification of test data, test environment
Test implementation	final test cases with test data, arrange in schedule
Test execution	executing, comparing, analyzing, reporting, logging
Test completion	collect and share data



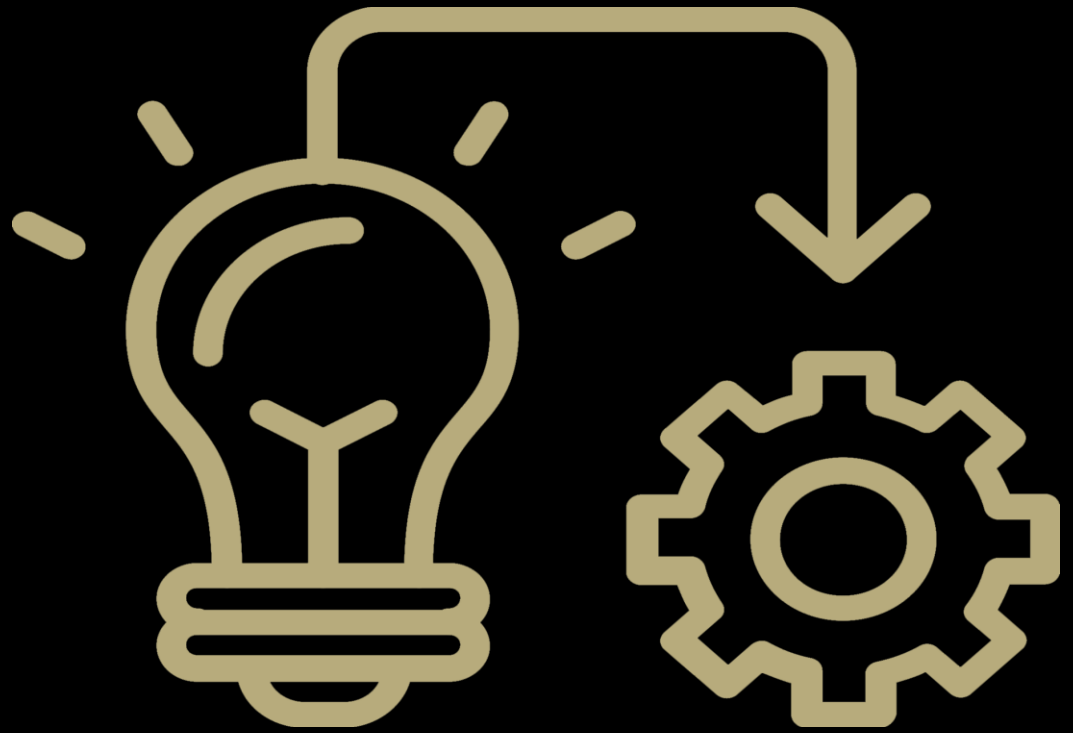


Γιατί διεξάγουμε τα Test;

1. Πρόληψη λαθών



2. Επιβεβαίωση ότι όλα τα requirements έχουν υλοποιηθεί



3. Επιβεβαίωση ότι η εφαρμογή είναι πλήρης και λειτουργεί όπως απαιτούν οι χρήστες και οι ενδιαφερόμενοι



4. Το χτίσιμο της εμπιστοσύνης ως προς το επίπεδο της ποιότητας της εφαρμογής



5. Να βρούμε λάθη ώστε να μειώσουμε το ρίσκο όπου το λογισμικό θα είναι ανεπαρκές



6. Να παρέχουμε επαρκείς πληροφορίες στα ενδιαφερόμενα μέρη (stakeholders)



7. Για τη συμμόρφωση στα συμβόλαια, νομικές ή ρυθμιστικές απαιτήσεις ή πρότυπα (standards)



Πόσο σταθερό είναι το επάγγελμα του Software Tester / Quality Assurance Engineer;





Μπορούμε να απαλλαγούμε από τους Testers; (Μέρος 1ο)

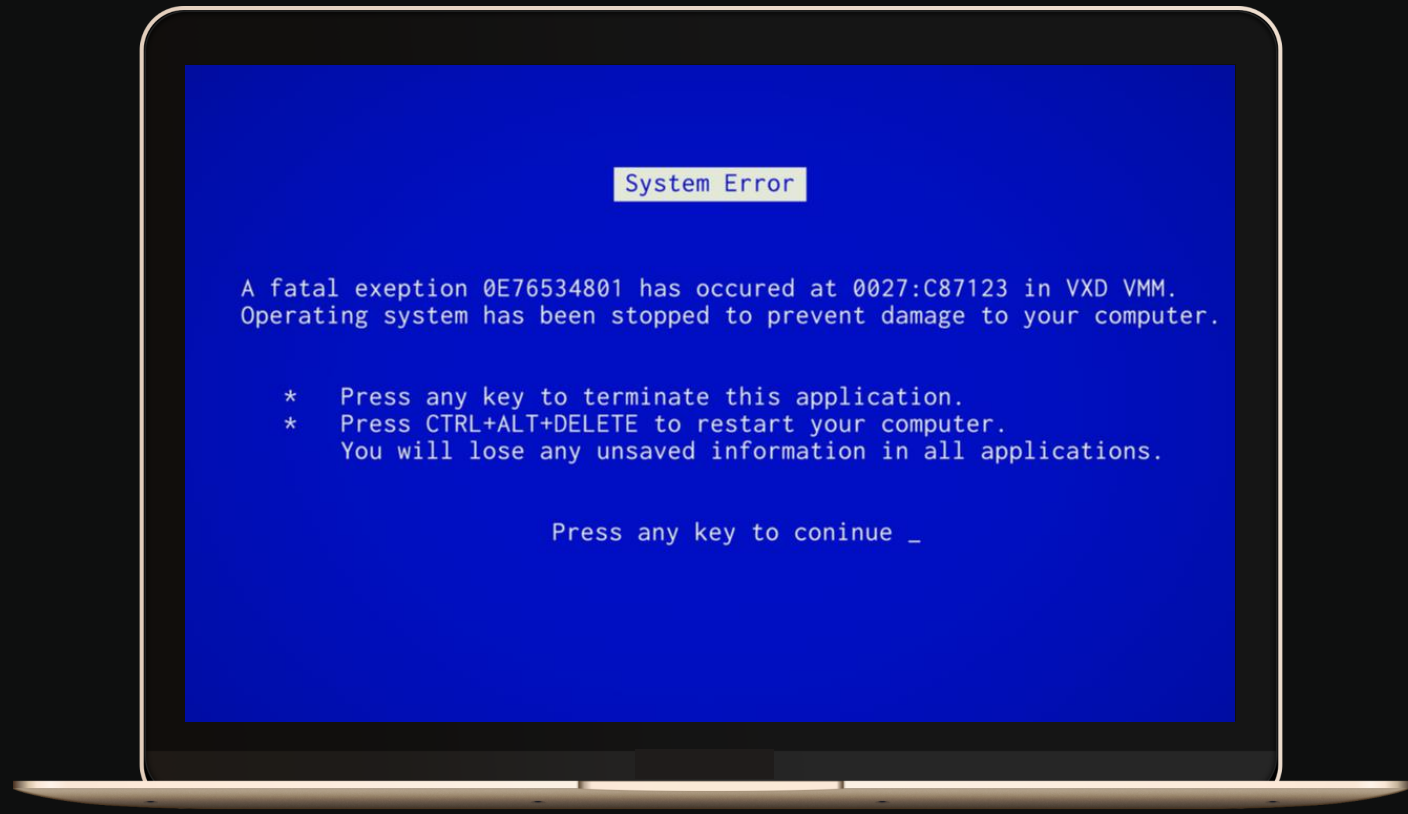
Μπορούμε να απαλλαγούμε από τους Testers;



1. Να δημιουργήσουμε λογισμικό χωρίς λάθη

Παρουσίαση του USB (1998)





Mars Climate Orbiter (1999)





Λάθη των Testers

Παραλείπουν σημαντικά Test
απροσεξία, πίεση χρόνου, έλλειψη γνώσεων, έλλειψη εμπειρίας, έλλειψη εξοπλισμού, φόβος λάθους

Μη κατανόηση των requirements
κάνουν υποθέσεις, έλλειψη επικοινωνίας, φόβος επικοινωνίας, κακή επικοινωνία

Λάθος αποτελέσματα
δεν τα αντιλαμβάνονται σωστά ή δεν τα εξακριβώνουν - διασταυρώνουν

Πλεονάζοντα ανόμοια συστήματα
(Redundant Dissimilar Systems)

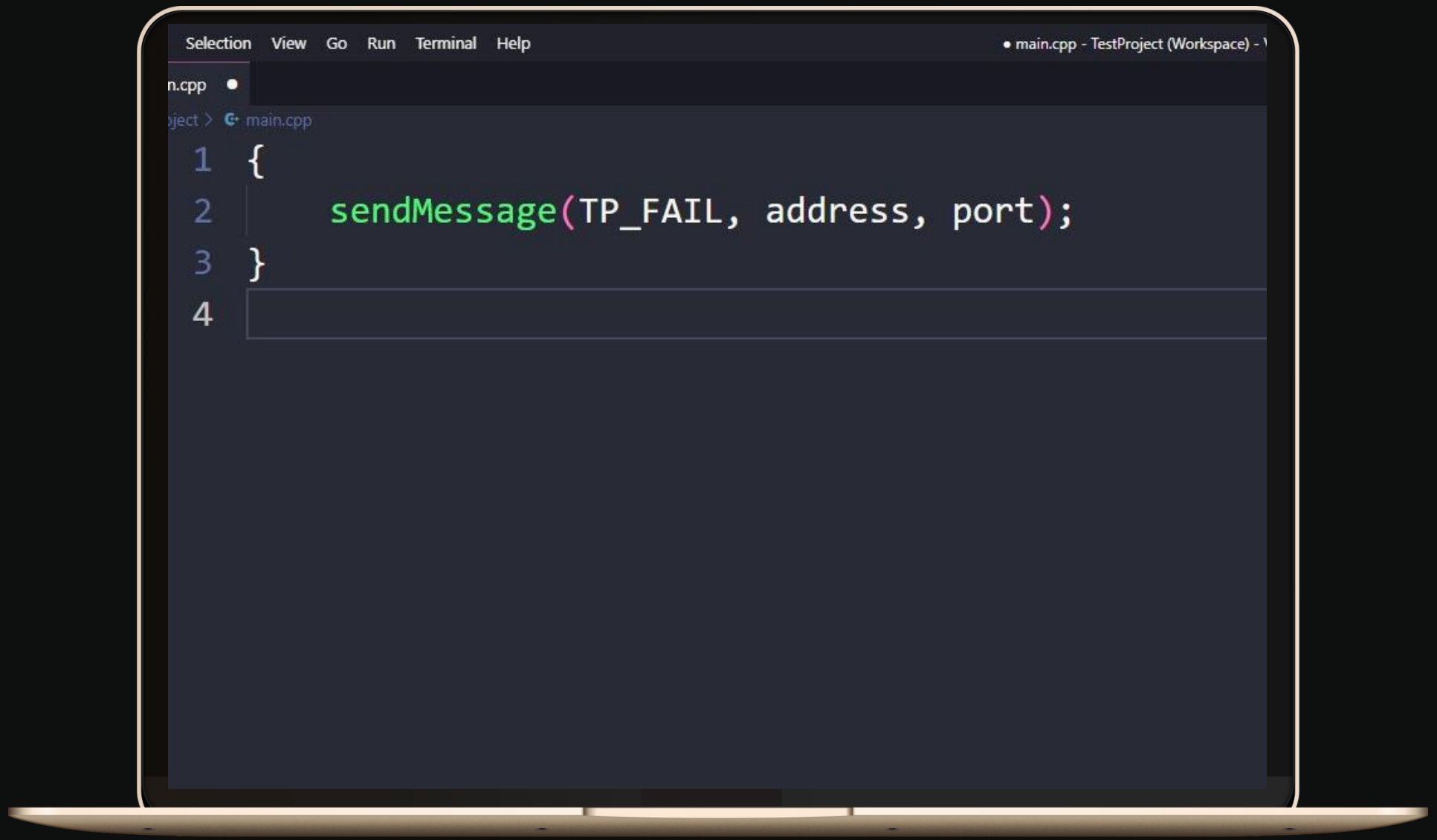
1

Τα πλεονάζοντα συστήματα είναι συνήθως ένας συνδυασμός λογισμικού και υλικού και μπορούν να ονομαστούν διπλά, τριπλά ή τετραπλά συστήματα, ανάλογα με τον αριθμό των ανεξάρτητων παρουσιών (δύο, τρία ή τέσσερα αντίστοιχα).

2

Τα Test που διεξάγουμε γι' αυτά ονομάζονται 'Failover Tests' και ανήκουν στην κατηγορία του Recoverability testing

Ελάχιστος δυνατός κώδικας





Μπορούμε να απαλλαγούμε από τους Testers; (Μέρος 2ο)

Μπορούμε να απαλλαγούμε από τους Testers;



2. Να μην τεστάρουν οι Testers αλλά κάποιοι άλλοι στη θέση τους

Γιατί δεν τεστάρουν οι developers;



1. Δεν είναι η δουλειά τους

Επαγγέλματα σχετικά με το Software Testing

Quality Assurance Engineer	Performance Tester
Team Coordinator	Security Tester
Test Manager	Acceptance Tester
Test Engineer	AI Tester
Test Analyst	Automotive Software Tester
Technical Test Analyst	Gambling Industry Tester
Test Automation Engineer	Mobile Application Tester
Agile Technical Tester	Model Based Tester
Operational Test Manager	Usability Tester

Αρμοδιότητες

Test Analyst



Functional suitability
Functional correctness
Functional appropriateness
Functional completeness



Usability
Appropriateness recognizability
Learnability
Operability
User interface aesthetics
User error protection
Accessibility



Compatibility
Interoperability

Αρμοδιότητες

Technical Test Analyst



Reliability
Maturity
Fault-tolerance
Recoverability
Availability



Performance efficiency
Time behavior
Resource utilization
Capacity



Maintainability
Analyzability
Modifiability
Testability
Modularity
Reusability



Portability
Adaptability
Installability
Replaceability



Security
Confidentiality
Integrity
Non-repudiation
Accountability
Authenticity



Compatibility
Co-existence

Γιατί δεν τεστάρουν οι developers;



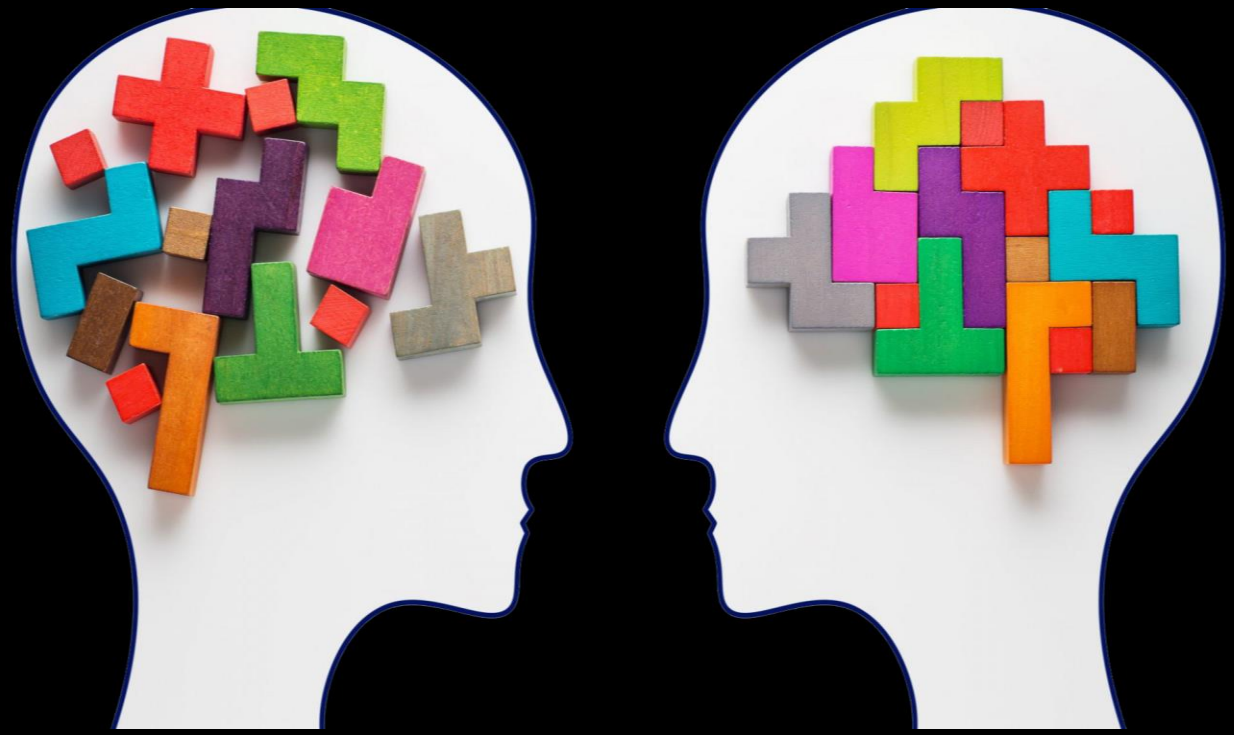
2. Δεν έχουν το χρόνο να το κάνουν

Γιατί δεν τεστάρουν οι developers;

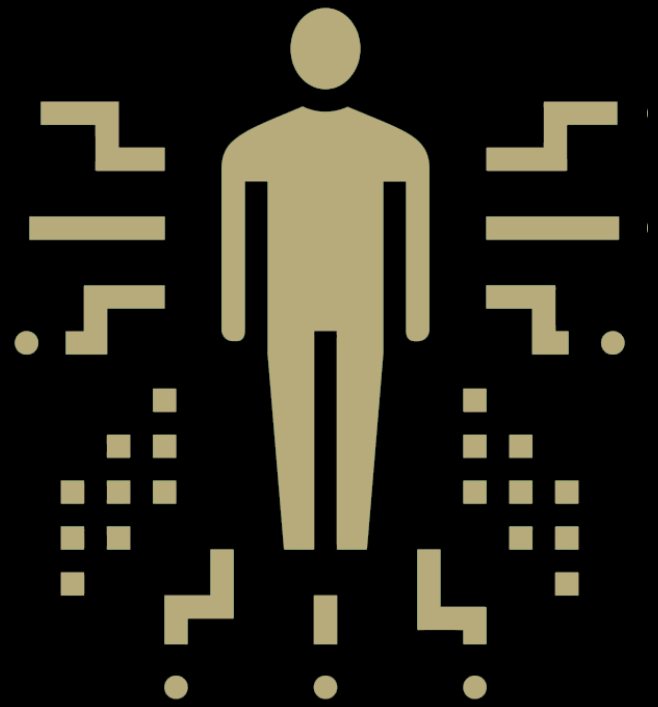


3. Το Test που θα διεξάγουν δεν θα είναι αντικειμενικό και ουδέτερο αλλά προκατειλημμένο (biased)

Γιατί δεν τεστάρουν οι developers;



4. Έχουν διαφορετική νοοτροπία (mindset)



Χαρακτηριστικά ενός Tester

Χαρακτηριστικά ενός Tester



1. Προσοχή στη λεπτομέρεια

Χαρακτηριστικά ενός Tester



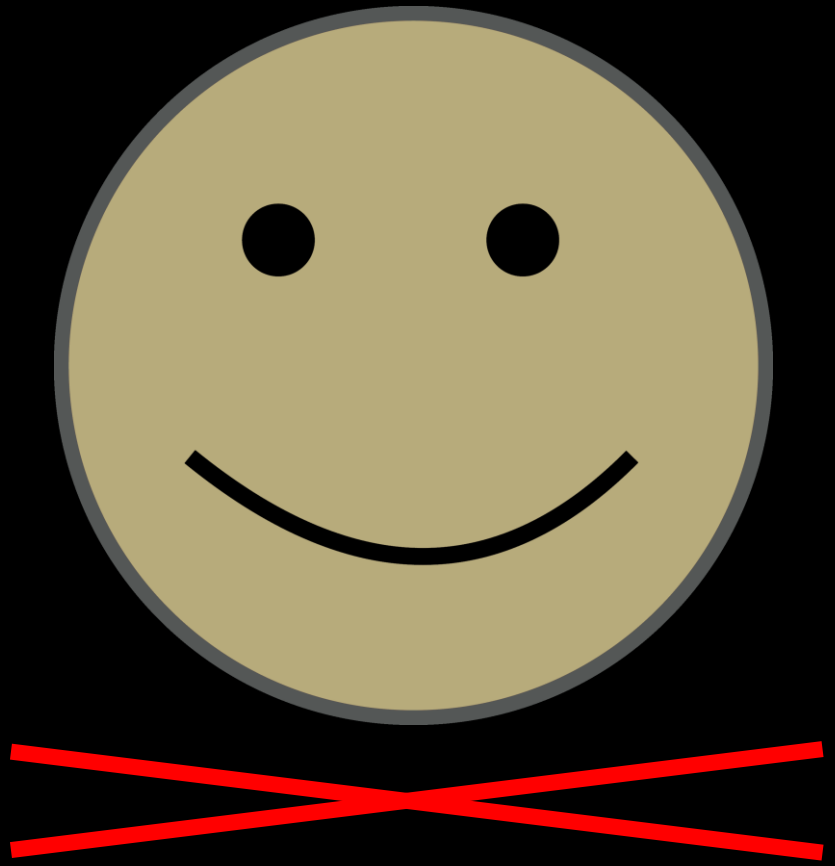
2. Κριτικό μάτι

Χαρακτηριστικά ενός Tester



3. Συγκέντρωση

Χαρακτηριστικά ενός Tester



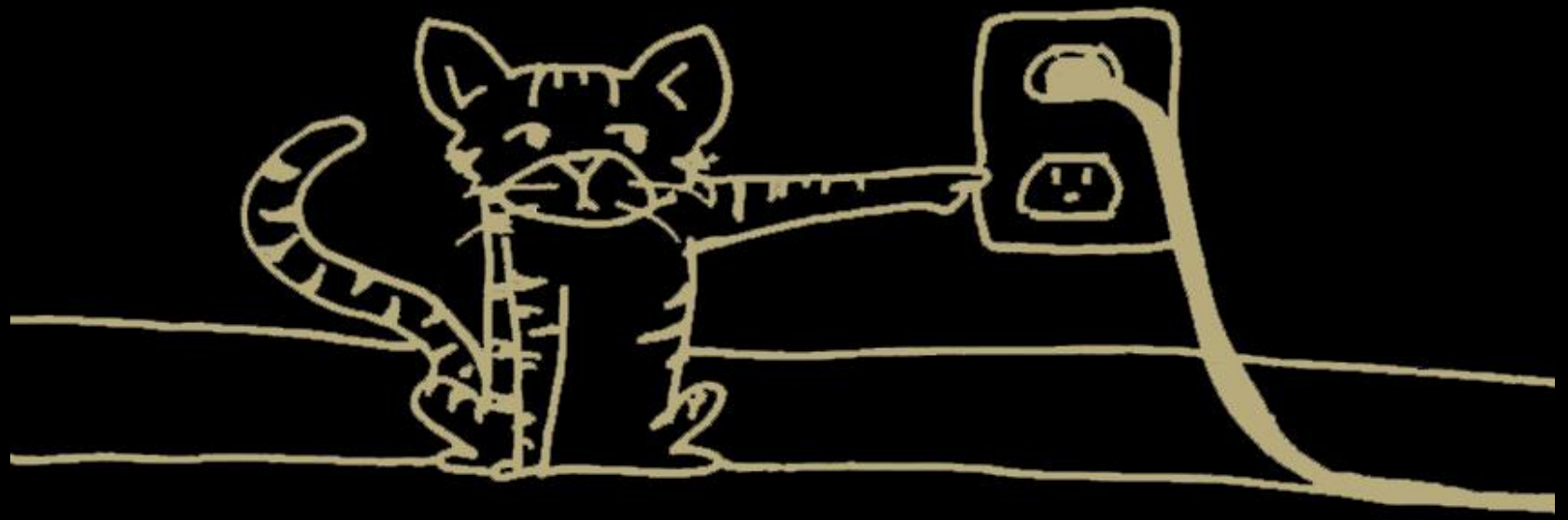
4. Επαγγελματική απαισιοδοξία
(professional pessimism)

Χαρακτηριστικά ενός Tester



5. Περιέργεια

Χαρακτηριστικά ενός Tester



5. Περιέργεια

"Curiosity killed the cat" is a proverb used to warn of the dangers of unnecessary investigation or experimentation

(WIKIPEDIA)

Χαρακτηριστικά ενός Tester

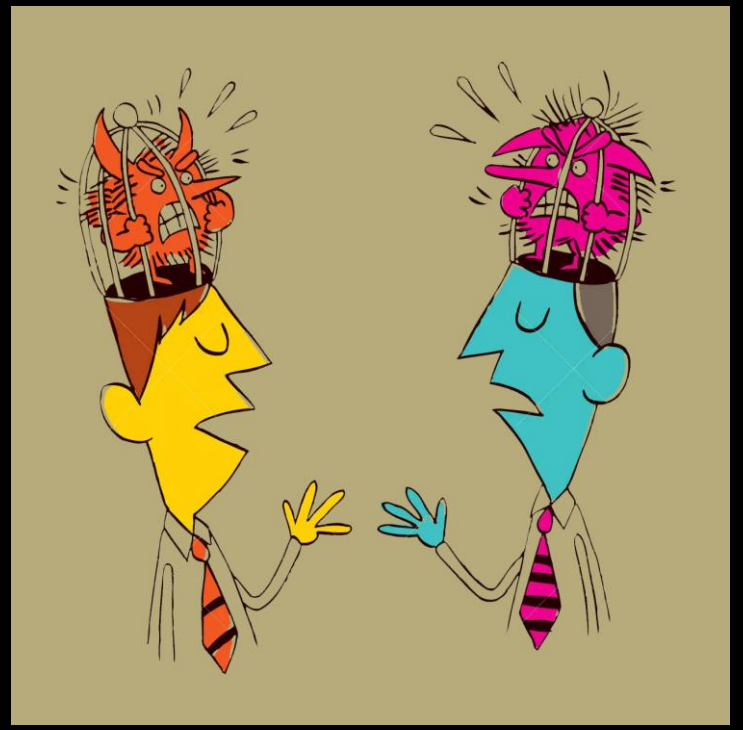


5. Περιέργεια

”Curiosity crashed the software” is a phrase used to denote the necessity of investigation and/or experimentation in software testing

(ZIKIDIS)

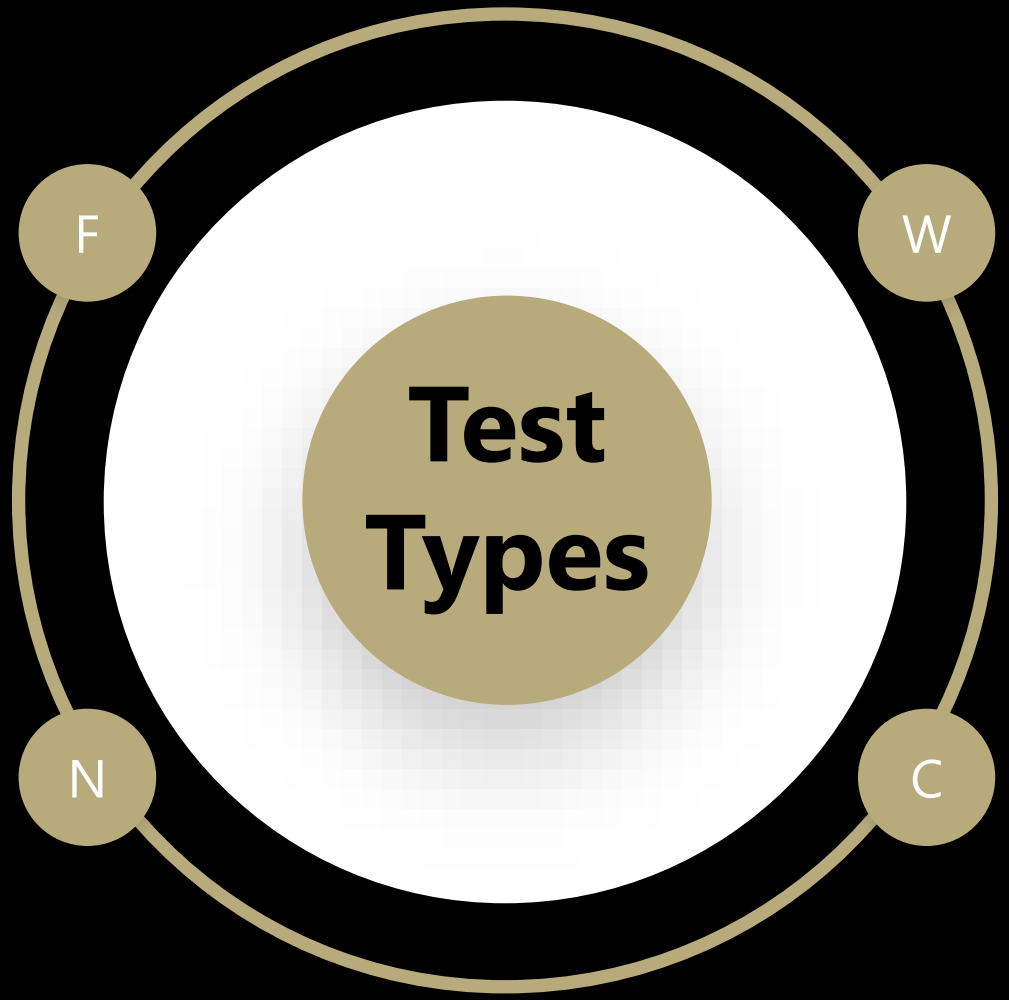
Χαρακτηριστικά ενός Tester



6. Τακτ και διπλωματία στις σχέσεις του

Κώδικας


File Edit Selection View Go Run Terminal Help
• main.cpp
main.cpp •
TestProject > main.cpp
1 for (int x = 0; x < 10; x++)
2 {
3 data[x] = getData(x);
4
5 sendData(data, 10);
6 }
7



Functional

White Box

Non Functional

Performance efficiency
Reliability
Usability
Security
Compatibility
Portability

Change Related

Confirmation
Regression

Test Techniques

White Box

- statement testing
- decision testing
- modified condition/decision coverage (MC/DC)
- multiple condition testing
- basis path testing
- API testing

Black Box

- equivalence partitioning
- boundary value analysis
- decision table testing
- state transition testing
- use case testing

Experience Based

- error guessing
- exploratory testing
- checklist based testing

Statement Testing (White Box)

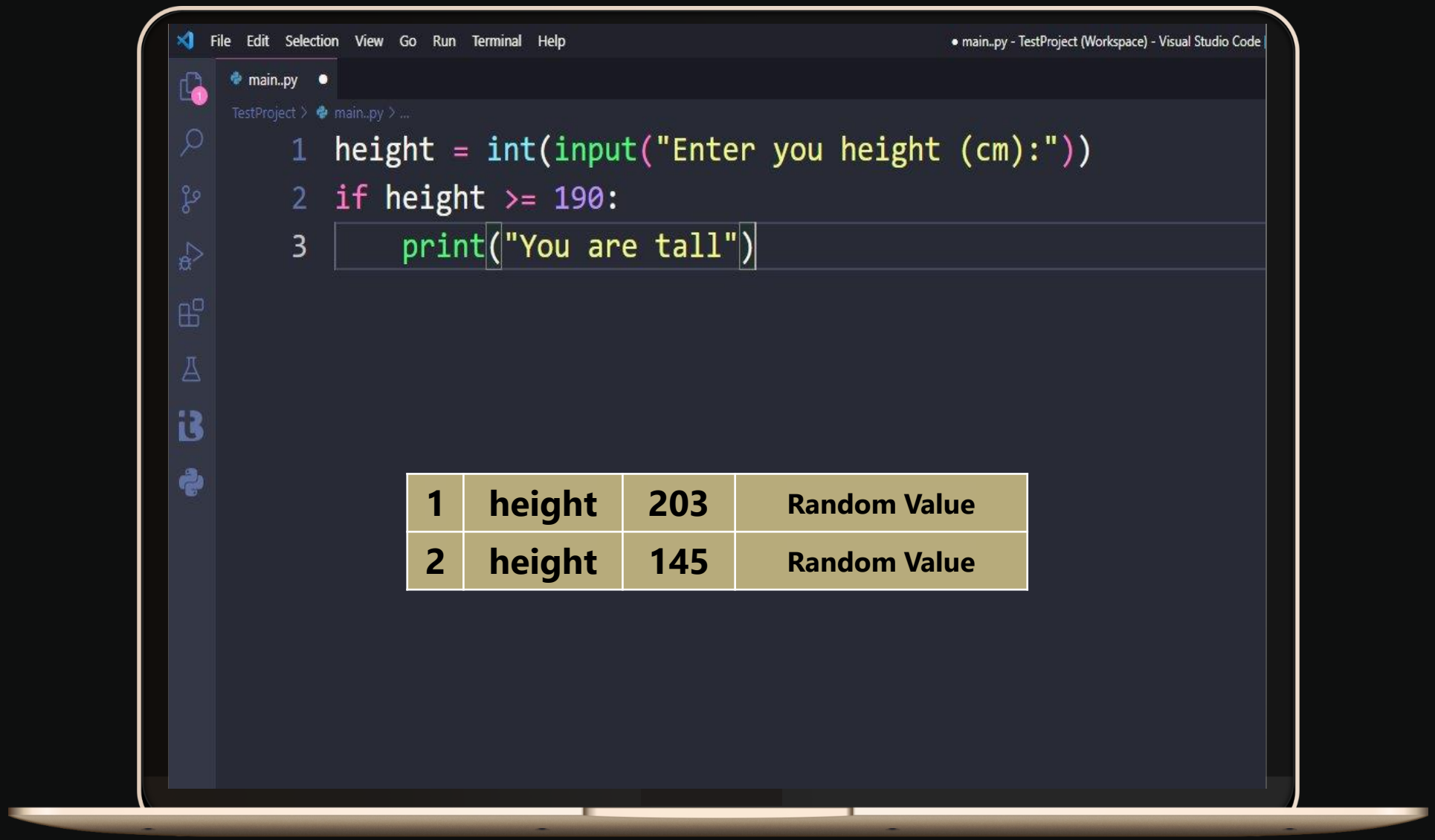
100% coverage με 1 test



1	height	197	Random Value

Decision Testing (White Box)

100% coverage με 2 test

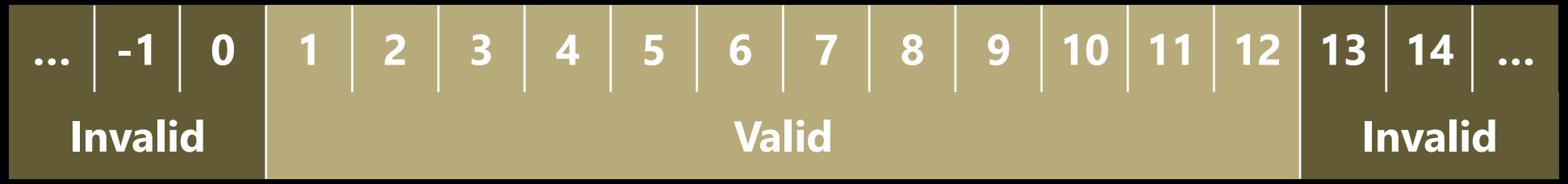


1	height	203	Random Value
2	height	145	Random Value

Equivalence Partitioning (Black Box)

100% coverage με 3 test

Month: [ ]

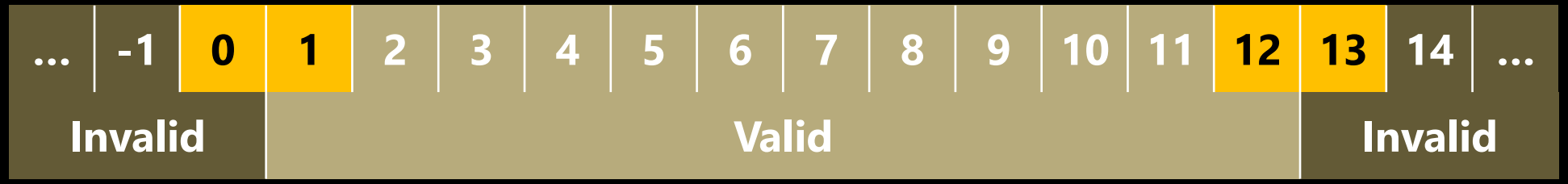


1	Month	-1.123.378	Random Value
2	Month	9	Random Value
3	Month	114.001	Random Value

Boundary Value Analysis (Black Box)

100% coverage με 4 test

Month: [ ]



1	Month	0	Boundary Value
2	Month	1	Boundary Value
3	Month	12	Boundary Value
4	Month	13	Boundary Value

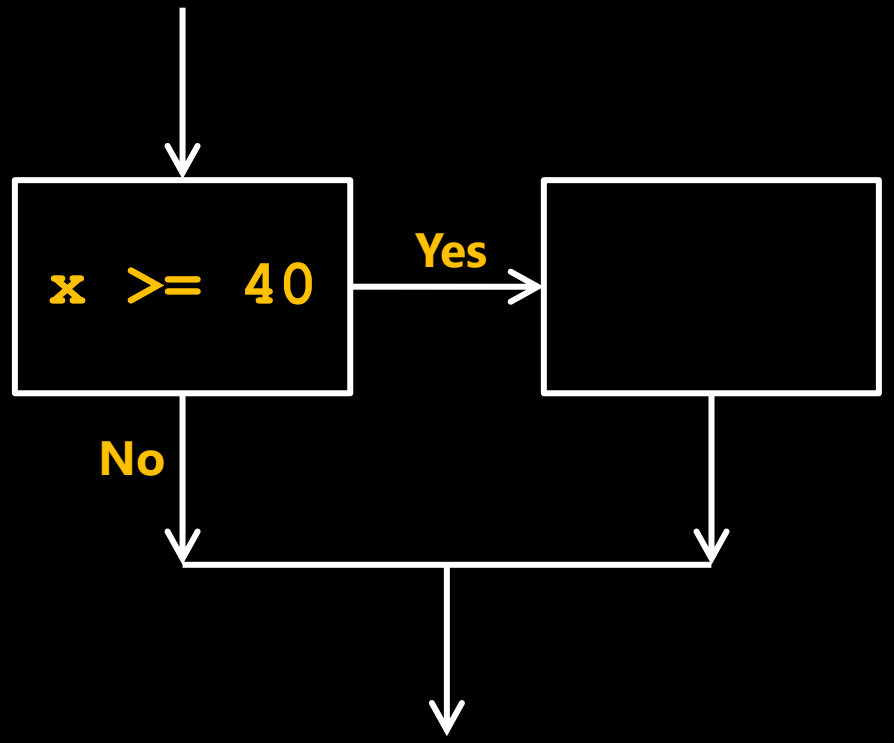


Άσκηση

Πόσα test χρειάζονται;

```
If (x > 39)
{
     // do some calculation
}

If (x >= 39)
{
     // do some calculation
}
```




```
switch (x) {

     case 35:
          calc();
          // break;

     case 40:
          anotherCalc();
          break;

     default:
          anotherCalc();
}
```

Πόσα test χρειάζονται;

Quality Characteristics:

- Modularity
- Reusability
- Analyzability
- Modifiability
- Fault Tolerance
- Recoverability
- Time Behavior
- Resource Utilization
- Non Repudiation
- Accountability
- Co-Existence

Άλλα Test:

- Control Flow Analysis
- Data Flow Analysis
- Call Graphs
- Fault Seeding
- Fault Injection
- Memory Leaks & Wild Pointers

- **Testability**

- Unit test using MC/DC
- MISRA for C or C++
- JSF coding standard for C++

- DO-178C
- RTCA/DO-254
- EUROCAE ED-12C

- ARP-4754A
- ISO/IEC 12207
- ISO/IEC 15504

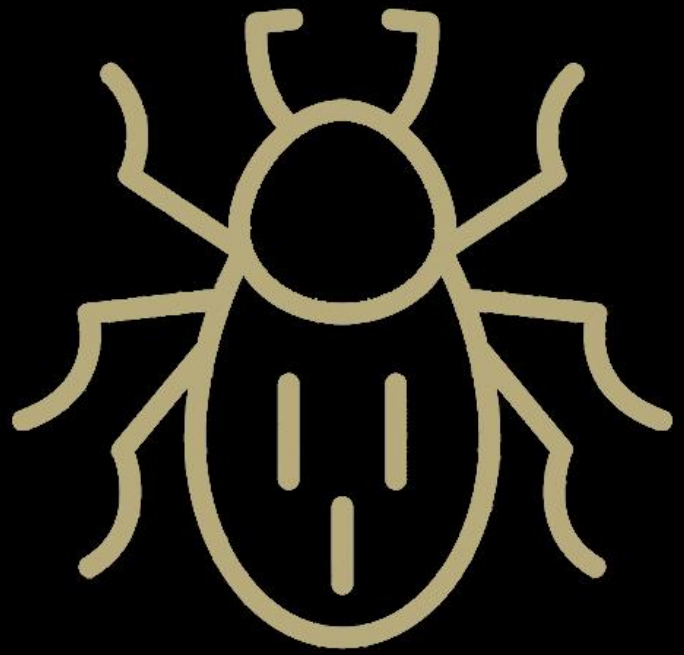
Οι 7 αρχές του Test
(7 testing principles)



1. Το Test υποδεικνύει τα λάθη
2. Το εξαντλητικό Test είναι ανέφικτο
3. Συγκέντρωση λαθών
4. Το παράδοξο του εντομοκτόνου
5. Το Test εξαρτάται από το περιεχόμενο
6. Το πρόωρο Test εξοικονομεί χρόνο και χρήμα
7. Η πλάνη της απουσίας λαθών

Οι 7 αρχές του Test
(7 testing principles)

1. Το Test υποδεικνύει τα λάθη
2. Το εξαντλητικό Test είναι ανέφικτο
3. Συγκέντρωση λαθών
4. Το παράδοξο του εντομοκτόνου
5. Το Test εξαρτάται από το περιεχόμενο
6. Το πρόωρο Test εξοικονομεί χρόνο και χρήμα
7. Η πλάνη της απουσίας λαθών



1. Το Test υποδεικνύει τα λάθη δεν αποδεικνύει την απουσία αυτών

Οι 7 αρχές του Test
(7 testing principles)

1. Το Test υποδεικνύει τα λάθη
2. Το εξαντλητικό Test είναι ανέφικτο
3. Συγκέντρωση λαθών
4. Το παράδοξο του εντομοκτόνου
5. Το Test εξαρτάται από το περιεχόμενο
6. Το πρόωρο Test εξοικονομεί χρόνο και χρήμα
7. Η πλάνη της απουσίας λαθών

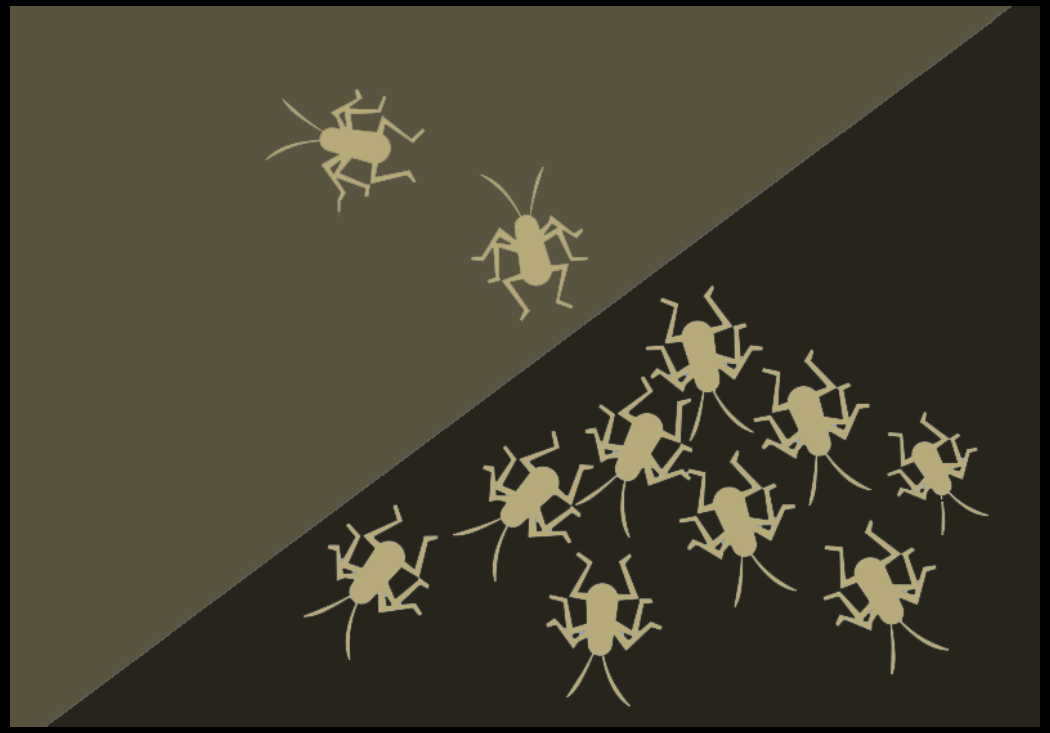




2. Το εξαντλητικό Test (όλοι οι πιθανοί συνδυασμοί) είναι ανέφικτο να εφαρμοστεί
(exhaustive testing is impossible)

Οι 7 αρχές του Test
(7 testing principles)

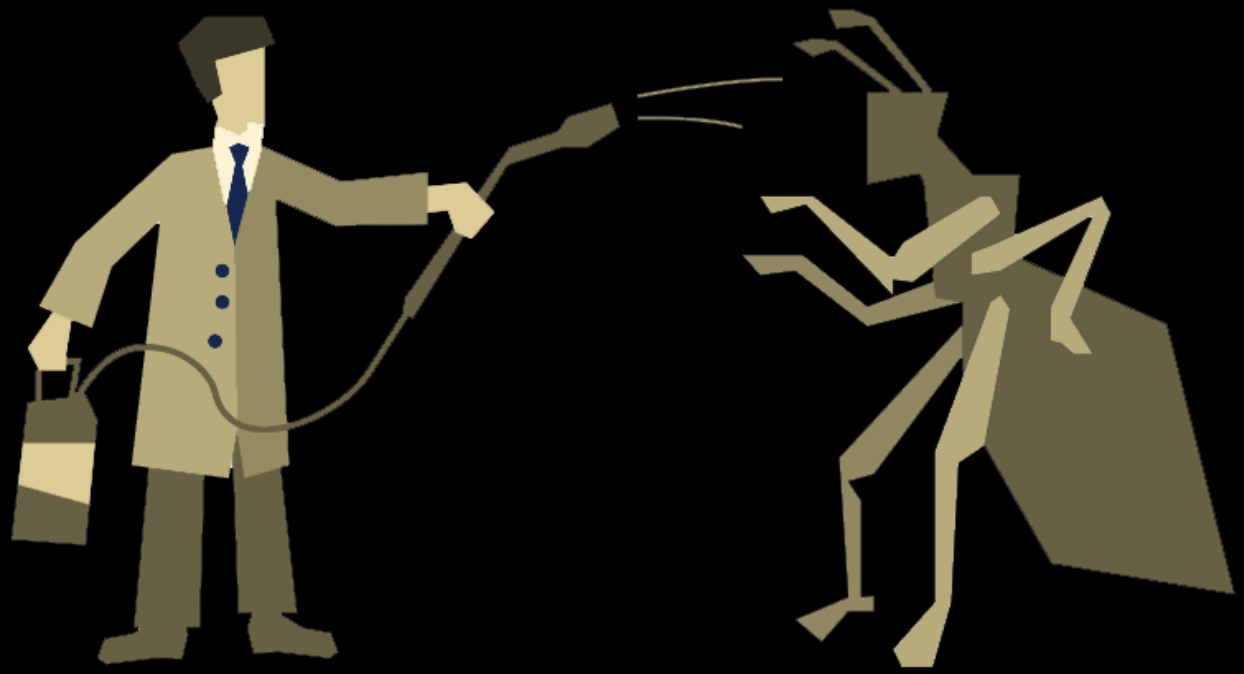
1. Το Test υποδεικνύει τα λάθη
2. Το εξαντλητικό Test είναι ανέφικτο
3. Συγκέντρωση λαθών
4. Το παράδοξο του εντομοκτόνου
5. Το Test εξαρτάται από το περιεχόμενο
6. Το πρόωρο Test εξοικονομεί χρόνο και χρήμα
7. Η πλάνη της απουσίας λαθών



3. Συγκέντρωση λαθών
(defect clustering)

Οι 7 αρχές του Test
(7 testing principles)

1. Το Test υποδεικνύει τα λάθη
2. Το εξαντλητικό Test είναι ανέφικτο
3. Συγκέντρωση λαθών
4. Το παράδοξο του εντομοκτόνου
5. Το Test εξαρτάται από το περιεχόμενο
6. Το πρόωρο Test εξοικονομεί χρόνο και χρήμα
7. Η πλάνη της απουσίας λαθών



4. Το παράδοξο του εντομοκτόνου
(pesticide paradox)

Οι 7 αρχές του Test
(7 testing principles)

1. Το Test υποδεικνύει τα λάθη
2. Το εξαντλητικό Test είναι ανέφικτο
3. Συγκέντρωση λαθών
4. Το παράδοξο του εντομοκτόνου
5. Το Test εξαρτάται από το περιεχόμενο
6. Το πρόωρο Test εξοικονομεί χρόνο και χρήμα
7. Η πλάνη της απουσίας λαθών



5. Το Test εξαρτάται από το περιεχόμενο
(context dependent)

Οι 7 αρχές του Test
(7 testing principles)

1. Το Test υποδεικνύει τα λάθη
2. Το εξαντλητικό Test είναι ανέφικτο
3. Συγκέντρωση λαθών
4. Το παράδοξο του εντομοκτόνου
5. Το Test εξαρτάται από το περιεχόμενο
6. Το πρόωρο Test εξοικονομεί χρόνο και χρήμα
7. Η πλάνη της απουσίας λαθών



6. Το πρόωρο Test (πρόληψη) μας εξοικονομεί χρόνο και χρήμα
(shift left)

Οι 7 αρχές του Test
(7 testing principles)

1. Το Test υποδεικνύει τα λάθη
2. Το εξαντλητικό Test είναι ανέφικτο
3. Συγκέντρωση λαθών
4. Το παράδοξο του εντομοκτόνου
5. Το Test εξαρτάται από το περιεχόμενο
6. Το πρόωρο Test εξοικονομεί χρόνο και χρήμα
7. Η πλάνη της απουσίας λαθών



7. Η πλάνη της απουσίας λαθών

Τι χρειάζεται να γνωρίζει ο Tester ή Quality Assurance Engineer (QA);



Προγράμματα διαχείρισης Test & Bugs
(**Jira**, **Confluence**, Xray, TestRail, Bugzilla, κλπ.)

SQL, Linux shell, Windows shell

Browser Dev tools

Προγράμματα διαχείρισης βάσεων δεδομένων
(Oracle SQL Developer, DBeaver, κλπ.)

Προγράμματα ελέγχου REST APIs και Web Services
(**Postman**, SoapUI, JMeter, LoadRunner, κλπ.)

Γλώσσες προγραμματισμού
(**JavaScript**, Groovy, **Java** κλπ.)

Version control systems
(**Git**, GitHub, GitLab, Bitbucket, SVN κλπ.)



Γιατί το επάγγελμα του Software Tester είναι συναρπαστικό;

Bug Report

Summary: Application is not checking for a valid 'Zip/Postal Code'

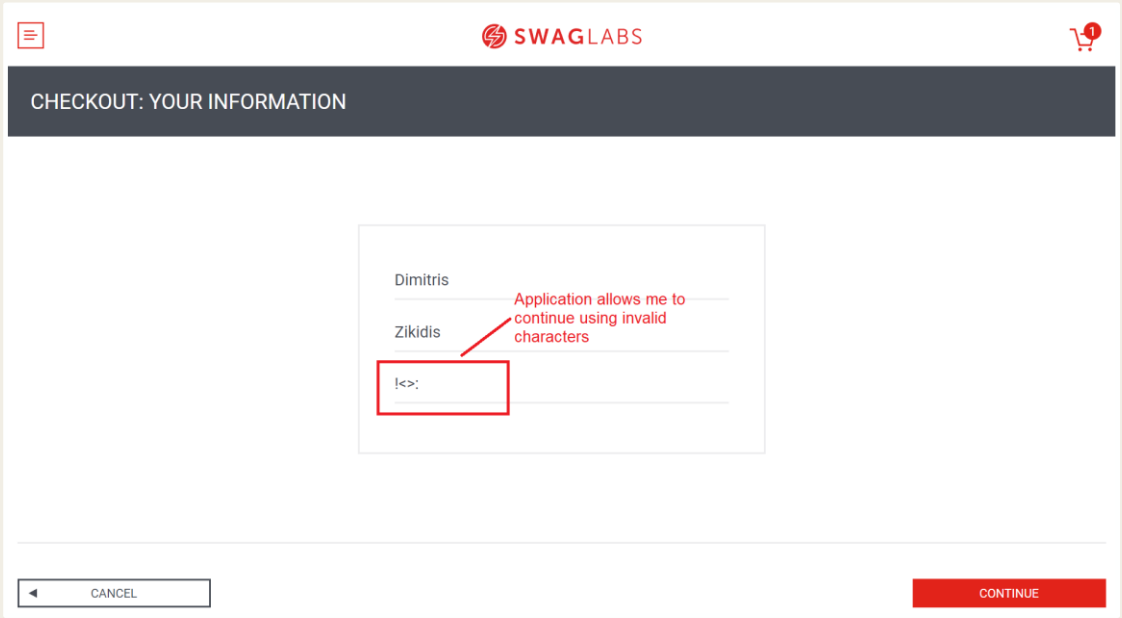
Description: When you make an order, 'Zip/Postal code' is mandatory. You are allowed to complete the order by providing an invalid 'Zip/Postal Code'

Actual Result: Application allows me to continue and complete order

Expected Result: Application should apply a validation rule for Zip/Postal code and should stop me from advancing on next screen and completing order

Steps to reproduce:

1. Navigate to https://www.saucedemo.com/ website
2. Login using username: **standard_user** and password: **secret_sauce**
3. Add in your cart any product of the provided list by clicking its 'ADD TO CART' button
4. Navigate to cart page by clicking on the cart icon which is located on the top right corner of the screen
5. On the cart screen layout click 'CHECKOUT' button
6. Fill in first and last name fields
7. Fill in 'Zip/Postal Code' field by providing an invalid Zip/Postal code for example: add some characters '!<>:'
8. Click 'CONTINUE' button




ONELITY

Τι προσφέρει;

Πρακτική άσκηση

- Developer
- Software Tester

Εργασία μετά την πρακτική

Σεμινάρια - Εκπαιδεύσεις

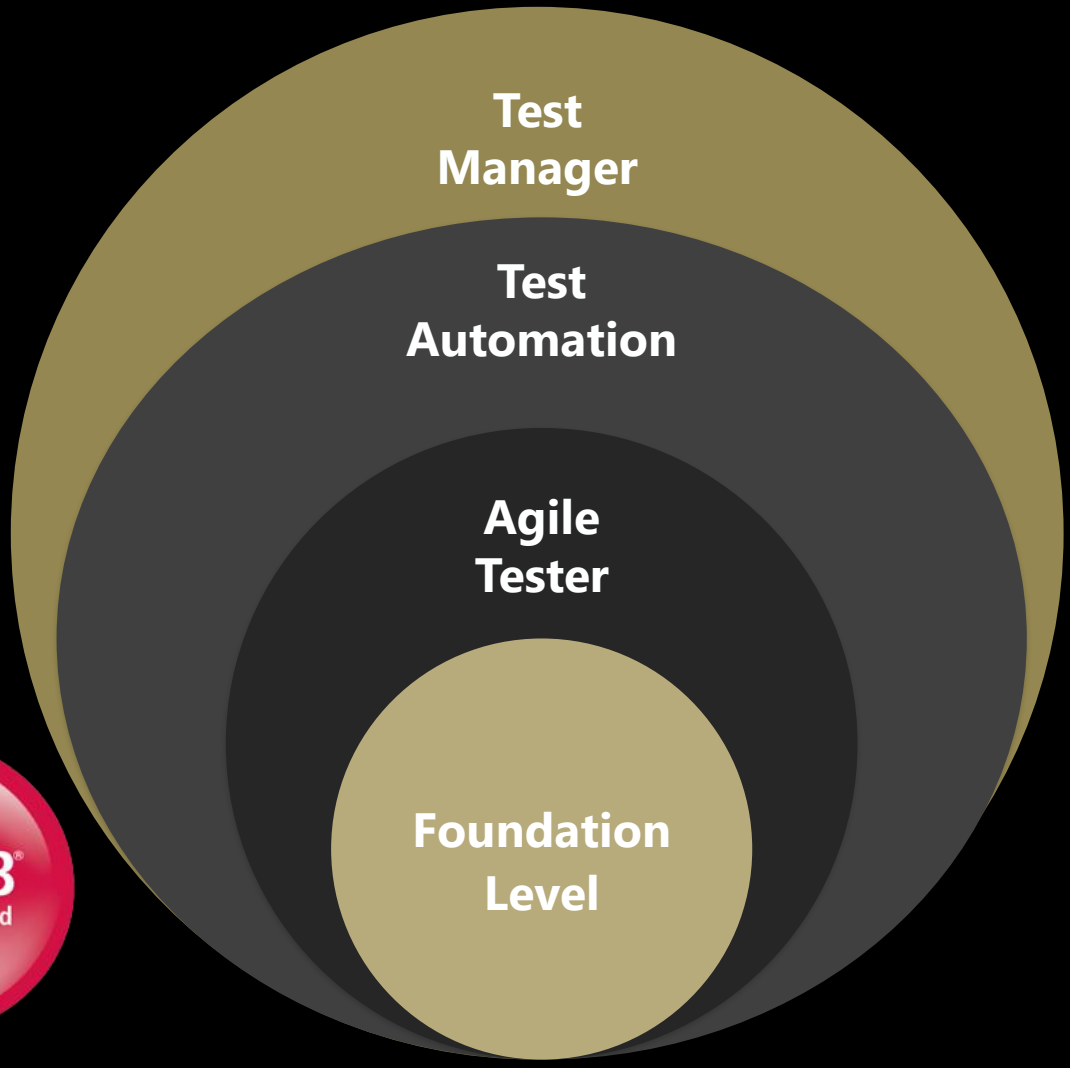
10% έκπτωση στους φοιτητές για τα σεμινάρια

-

Δωρεάν εξετάσεις για τις πιστοποιήσεις

Εκπαιδεύσεις & Πιστοποιήσεις ISTQB®

Πιστοποιήσεις αναγνωρισμένες σε περισσότερες από 120 χώρες



Εκπαιδεύσεις & Πιστοποιήσεις
PSM-I



Who is this program for?

Practitioners, Scrum Masters, Agile/Scrum Coaches and anyone involved in product delivery using Scrum.

Course Learning Objectives: Understand the theory and principles behind Scrum and empiricism, deliver value to your organization and learn the skills, traits & behavior shifts required to be a Scrum Master.

Study Load: The assessment is difficult: We strive to ensure that the assessments remain valuable and relevant through frequent inspection and adaptation of the assessment content.

2. Professional Scrum Master PSM-I:

Become the Scrum Master you always wanted.

Εκπαιδεύσεις & Πιστοποιήσεις Kubernetes

3. Kubernetes: Bridging the gap between software developers and operations teams.

Application Developer CKAD	**Certifies that users can design, build, configure, and expose cloud native applications for Kubernetes.**
Administrator CKA	**Allows users to demonstrate their competence in a hands-on, command-line environment.**
Security Specialist CKS	**Provides assurance that a CKS has the skills & the knowledge for securing container-based applications and Kubernetes platforms during build, deployment and runtime.**



Software Tester ή Developer;



Καλώς ήρθατε στον συναρπαστικό κόσμο του Software Testing!!!

Dimitris Zikidis