

Testscripts automatiseren

Echte werk gaat beginnen!

On the job

- Wat zijn jullie ervaringen?
- Wat zijn jullie verwachtingen?
- Wat lijkt jullie de ideale situatie?



On the job -> aanpak

- Kant en klare regressieset (yeah!)
 - Test cases uitgewerkt door analist en gechecked
- Kant en klare regressie set voor functioneel testen
 - Test cases uitgewerkt, wordt ook gebruikt
- Nog geen compleet uitgewerkte regressie set, wordt wel al getest
 - Tester leeft een beetje in zijn eigen wereld (docs)
- Biz/FB weet wat ze willen, moet alleen nog worden uitgewerkt
 - Hebben een doc voor overzicht wanneer ze zelf testen

3



On the job

- NIKS.....
 - Willen gaan automatiseren
 -

En Nu?

4



Skills, niet alleen programmeren

- Functioneel testen
- Test ontwerpen
- Test coverage
- Test management
- Communicatie
- Verwachtingsmanagement (communiceren)

Afhankelijk van de opdracht,
goed om van te voren onderzoek te doen

5



Vragen of opmerkingen?

Back to Selenium!

Road so far

- Browser opgestart
- Genavigeerd naar een url
- Element benaderd
- Geklikt
- Tekst vak ingevuld

Zijn dat alle handelingen die wij dagelijks doen?

7



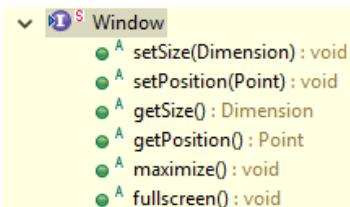
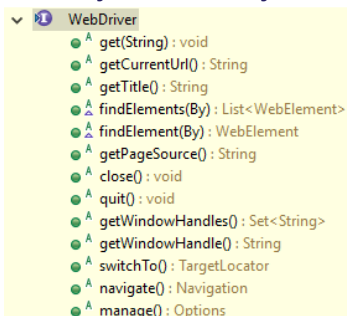
Selenium Interfaces (leer ze alsjeblieft niet uit je hoofd!)

- Main interface
 - Pagina openen
 - Elementen benaderen

```
WebDriver driver = BrowserFactory.  
createBrowser(Browser.CHROME);
```

```
driver.get("http://www.nu.nl");
```

```
driver.manage().window().maximize();
```



8

http://www.seleniumhq.org/docs/03_webdriver.jsp#introducing-the-selenium-webdriver-api-by-example



Selenium Interfaces (leer ze alsjeblieft niet uit je hoofd!)

- `driver.findElement(By.id("passwd")).sendKeys("1qazxsw2");`

WebElement

- `click() : void`
- `submit() : void`
- `sendKeys(CharSequence...) : void`
- `clear() : void`
- `getTagName() : String`
- `getAttribute(String) : String`
- `isSelected() : boolean`
- `isEnabled() : boolean`
- `getText() : String`
- `findElements(By) : List<WebElement>`
- `findElement(By) : WebElement`
- `isDisplayed() : boolean`
- `getLocation() : Point`
- `getSize() : Dimension`
- `getRect() : Rectangle`
- `getCssValue(String) : String`

- `driver.navigate().refresh();`

Navigation

- `back() : void`
- `forward() : void`
- `to(String) : void`
- `to(URL) : void`
- `refresh() : void`

9



Selenium Interfaces (leer ze alsjeblieft niet uit je hoofd!)

- `driver.manage().timeouts().implicitlyWait(5, TimeUnit.SECONDS);`

Timeouts

- `implicitlyWait(long, TimeUnit) : Timeouts`
- `setScriptTimeout(long, TimeUnit) : Timeouts`
- `pageLoadTimeout(long, TimeUnit) : Timeouts`

- `driver.switchTo().alert();`

TargetLocator

- `frame(int) : WebDriver`
- `frame(String) : WebDriver`
- `frame(WebElement) : WebDriver`
- `parentFrame() : WebDriver`
- `window(String) : WebDriver`
- `defaultContent() : WebDriver`
- `activeElement() : WebElement`
- `alert() : Alert`

- Input methoden managen.
 - Right to Left
 - Ander tekens (Japans, Hebreeuws)
- `driver.manage().ime().getAvailableEngines()`

ImeHandler

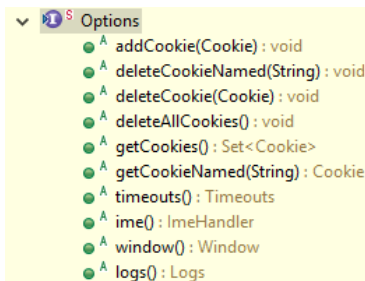
- `getAvailableEngines() : List<String>`
- `getActiveEngine() : String`
- `isActive() : boolean`
- `deactivate() : void`
- `activateEngine(String) : void`

10



Selenium Interfaces (leer ze alsjeblieft niet uit je hoofd!)

- Cookies managen.



- `driver.manage().deleteAllCookies();`

Kan een belangrijke optie zijn

11



Een element benaderen wat er niet meteen is

- WebDriver laten wachten en de DOM te pollen

Implicit Waits

An implicit wait is to tell WebDriver to poll the DOM for a certain amount of time when trying to find an element or elements if they are not immediately available. The default setting is 0. Once set, the implicit wait is set for the life of the WebDriver object instance.

java

```
WebDriver driver = new FirefoxDriver();
driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);
driver.get("http://somedomain/url_that_delays_loading");
WebElement myDynamicElement = driver.findElement(By.id("myDynamicElement"));
```

- WebDriver laten wachten op een specifiek attribuut
 - Explicit wait

http://www.seleniumhq.org/docs/04_webdriver_advanced.jsp#explicit-and-implicit-waits

12



Een element benaderen wat er niet meteen is

- We niet de hele test willen laten wachten (implicit) kunnen we een explicit gebruiken voor wanneer het element "er is"

```
public String validationMessage() {
    WebElement alertMessageElement = (new WebDriverWait(driver, 10))
        .until(ExpectedConditions.presenceOfElementLocated(By.cssSelector(".alert.alert-success")));

    return alertMessageElement.getText();
}
```

- In het ergste geval. Hard coded -> Java Sleep ☹️
- Wachten is een kunst!

13



Testcase 6.01 – Sign out testen

- Opdrachten bijlage -> 6.01
- Maak nieuwe git branch aan en checkout
 - "git branch feature/linear"
 - "git checkout feature/linear"
 - Code away.
- Denk aan validatie!

14



Testcase 6.01 – Sign out testen Extra

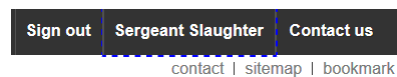
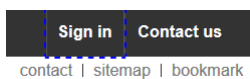
- Hoe hard coded maken wij onze tests
- Als Selenium een element niet kan vinden krijgen we een error.
- Manier om af te vangen natuurlijk maar we kunnen ook de code enig sinds "error proof" maken
- Vereist wat analyse....
- ... daar zijn we goed in! 😊

15



Testcase 6.01 – Sign out testen Extra

- Leg de situatie naast elkaar



16



Testcase 6.01 – Sign out testen Extra

- Leg de situatie naast elkaar met de code

```

class="authentication show-left-column show-right-column lang_en">
<div id="contact-link">
<div class="header_user_info">
  <a class="login" href="https://techblog.polteq.com/testshop/index.php?controller=my-account" rel="nofollow" title="Log in to your customer account"> Sign in </a>
</div>
</nav>

```

17



Testcase 6.01 – Sign out testen Extra

- Leg de situatie naast elkaar met de code

```

my-account show-left-column show-right-column lang_en">
<div id="contact-link" class="is_logged">
<div class="header_user_info">
  <a class="account" href="https://techblog.polteq.com/testshop/index.php?controller=my-account" rel="nofollow" title="Log in to your customer account"> Sign out </a>
</div>
<div class="header_user_info">

```

18



Testcase 6.01 – Sign out testen Extra

- Leg de situatie naast elkaar met de code
- Class header_user_info is beide gevallen aanwezig. (ingelogd 2x zelfs)
- Valt er 1 af als we uitgelogd zijn
- Als we een locator bouwen die het element benaderdt wat er altijd is...
- ...kunnen we daar een Assert op doen
- Zo valt onze code niet om als we niet uitgelogd zijn maar krijgen een mooie Assert warning

19



Testcase 6.01 – Sign out testen Extra

- Samen of eerst zelf even proberen?

20



Git actie

- Start GIT bash
- "git status"
- "git add ."
- "git commit -m "[message]"
- "git push origin [branch naam]"

21



Testcase 6.02 – Bestellen van accessoires

- Opdrachten bijlage -> 6.02
- Correcte GIT branch?
- Denk aan validatie!
- Let op: fun time is over. Deze is wat pittiger. Kijk goed naar de tips in de opdrachten
- Na de oefening duiken we verder in de wait mogelijkheden!

22



Testcase 6.02 – Bestellen van accessoires

Vragen?

23



Git actie

- Start GIT bash
- "git status"
- "git add ."
- "git commit -m "[message]"
- "git push origin [branch naam]"

24



Zijn er meer TestNG annotaties?



25



@BeforeMethod en @Aftermethod, handig! (verplicht?)

- Welke snippets code hebben nu meerdere keren gebruikt in elke test case?
- Hoe kunnen we duplicatie voorkomen door het gebruik van annotaties?

26



@BeforeMethod en @AfterMethod

```

TestShopScenario tearDown()
1 package testcases;
2
3 import io.github.bonigarcia.wdm.ChromeDriverManager;
4 import org.openqa.selenium.WebDriver;
5 import org.openqa.selenium.chrome.ChromeDriver;
6 import org.testng.annotations.AfterMethod;
7 import org.testng.annotations.BeforeMethod;
8
9 public class TestShopScenario {
10
11     @BeforeMethod
12     public void setUp() {
13         ChromeDriverManager.getInstance().setup();
14         WebDriver driver = new ChromeDriver();
15
16         // Open the website
17         driver.get("https://techblog.polteq.com/testshop/index.php");
18     }
19
20     @AfterMethod
21     public void tearDown() {
22         driver.quit();
23     }
24 }
25

```

Rood??

27



@BeforeMethod en @AfterMethod

```

public class TestShopScenario {

    protected WebDriver driver;

    @BeforeMethod
    public void setUp() {
        ChromeDriverManager.getInstance().setup();
        driver = new ChromeDriver();

        // Open the website
        driver.get("https://techblog.polteq.com/testshop/index.php");
    }

    @AfterMethod
    public void tearDown() {
        driver.quit();
    }
}

```

28



De annotaties gebruiken dmv overerving

- Eigenschappen/gedrag overerven
- "Is een" relatie
- In dit geval FirstSeleniumTest "is een" TestShopScenario

```
public class FirstSeleniumTest extends TestShopScenario {
```

- Elke (test) class die van **TestShopScenario** overerft, maakt gebruik van de setup en teardown methoden dankzij de annotaties.

29



Opdracht

1. Maak de class TestShopScenario
2. Pas testcase 6.01 en 6.02 aan zodat er gebruik wordt gemaakt van de TestShopScenario
3. Git add / commit / push

30



If / Else

- Kunnen de code laten kiezen welke weg te bewandelen door een conditie
- If
 - Heeft een of meerdere condities
- Else if
 - Komt na een **if** en heeft een of meerdere condities
- Else
 - Komt na een **if** of **else if** en heeft geen condities

31



Conditie mogelijkheden (relationele operators)

- Met de vergelijking van waarden kun je een conditie van een (else) if vormen. De uitkomst is true of false.
 - Groter dan: **>**
 - Kleiner dan: **<**
 - Gelijk aan: **==**
 - Groter dan of gelijk aan: **>=**
 - Kleiner dan of gelijk aan: **<=**
 - Ongelijk aan: **!=**

32



Voorbeeld met relationele operators

```
private String getWinner(int scorePlayerOne, int scorePlayerTwo) {
    if (scorePlayerOne > scorePlayerTwo)
    {
        return "Player one wins!";
    }
    else if (scorePlayerOne < scorePlayerTwo)
    {
        return "Player two wins!";
    }
    else
    {
        return "It's a tie!";
    }
}
```

33



AND en OR – als er meer condities zijn

- **&& (AND)**

- Beide condities moeten waar zijn:

```
int saldo = 150;
if (saldo > 64 && saldo < 256)
```

- **|| (OR)**

- Een logische OE, bijvoorbeeld:

```
double saldo = 150;
boolean crediteur = true;
if (credit > 256 || crediteur)
```

34



Oefeningen: If / Else

- Maak een class conditionExercises
- Oefening 1: maak methode bootcampAgeChecker. Bouw een if statement: Als de opgegeven leeftijd boven de 21 krijgt de deelnemer een console out met "deelnemer toegelaten". Onder de 21: "deelnemer is te jong". De test is een aparte methode de bovenstaande methode aanroept.
- Run de test met de leeftijd 19, 21 en 24
- Oefening 2: Breid de bootcampAgeChecker uit: als de leeftijd 65 of er boven is dan volgt er een print: Deelnemer niet toegelaten, check bij de manager.
- Run de test met de leeftijd 12, 21, 64, 65

35



Testcase 6.03 – Cart leegmaken

- Opdrachten bijlage -> 6.03
- Correcte GIT branch?
- Maak gebruik van If/Else
- Denk aan validatie!

36



Testcase 6.03 – Cart leegmaken

Vragen?

Git add / commit / push

37



Testcase 6.04 – My personal info aanpassen

- Opdrachten bijlage -> 6.04
- Correcte Git branch?
- Maak gebruik van If/Else
- De testcase moeten meerdere malen te runnen zijn waarbij de personal info ook echt veranderdt!
- Tip: Tekst in een textbox leeft ergens in het element als attribuut
- Tip: `string.Equals()`
- Denk aan validatie!

38



Testcase 6.04 – My personal info aanpassen

Vragen?

Git add / commit / push

39



Debugging

- Debuggen is handig als we fouten tegenkomen die we niet 1,2,3 begrijpen
 - Analyse waarde van variabelen
 - Stap voor stap de code laten uitvoeren
- Door een breakpoint te zetten kunnen we bepalen waarde de uitvoering van de code moet stoppen
- Dubbelklik in de linkerkant van de editor, naast de regelaanduiding. In ons geval naast regel waarin we een attribute opvragen

```

32 //Retrieve the current mail adres
33 String currentFirstName = driver.findElement(By.id("firstname")).getAttribute( S: "value");

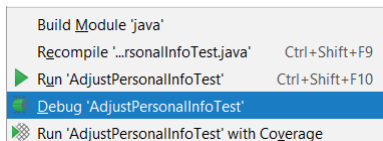
```

40



Debuggen

- Rechtermuisknop op de class -> Debug

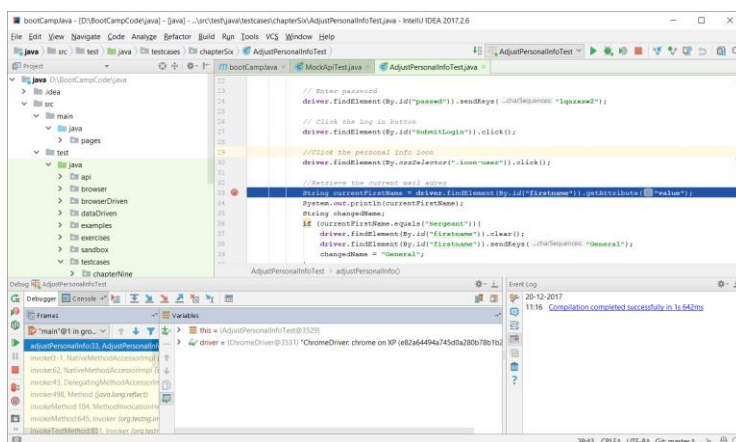


41



Debuggen

- De run stop bij het bolletje, dus voor de regel.



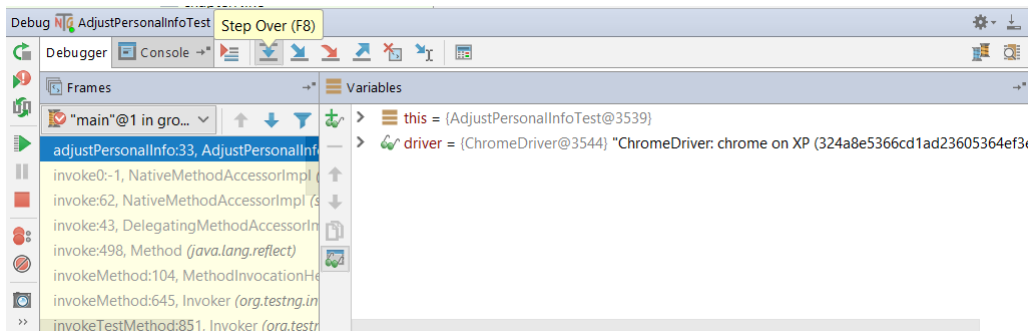
- Verschil tussen step in en step over?

42



Debuggen

- Step over (F8)
- Voor de statement uit
- Stop bij de volgende regel weer / statement

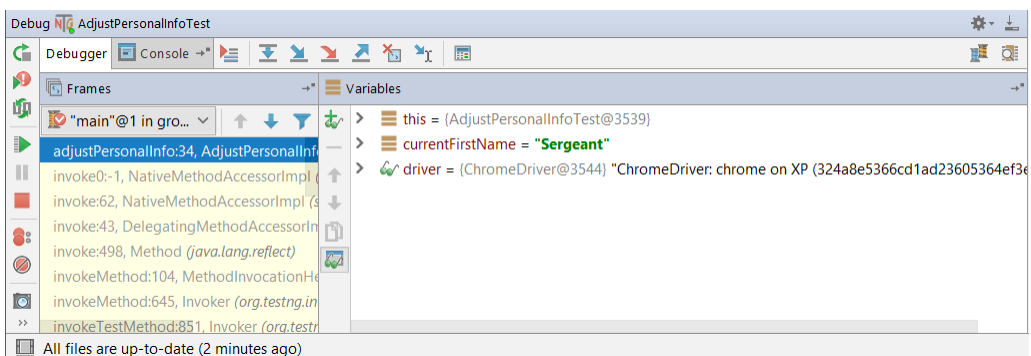


43



Debuggen

- We krijgen nu de waarde terug van currentFirstName in realtime



- En we gaan door naar de volgende statement (maar wordt nog niet uitgevoerd!)

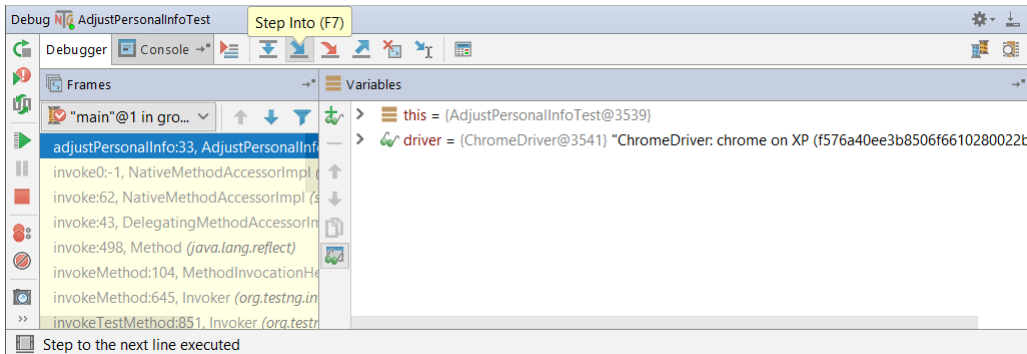
```
String currentFirstName = driver.findElement(By.id("firstname")).getAttribute("value");
System.out.println(currentFirstName); currentFirstName: "Sergeant"
```

44



Debuggen

- Step into (F7)
- "Duikt dan in de onderliggende code van de statement"

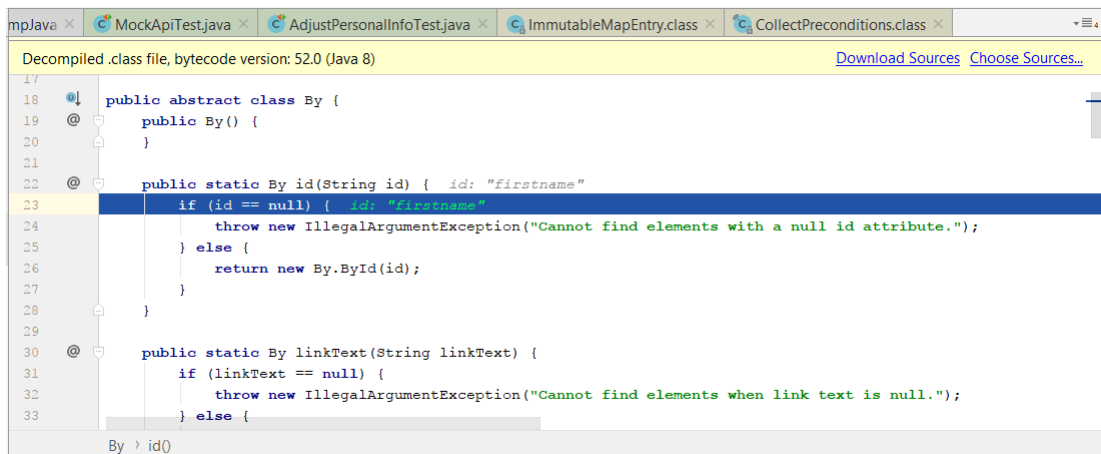


45



Debuggen

- In dit geval, Selenium code:



46



Slimme manier om met data om te gaan

- Array
 - Zelfde data type
 - Vooraf bepaalde de grootte
- List
 - Zelfde data type
 - Grootte niet gedefinieerd

To the code! – Collections.java

47



Betere manier om te itereren

- While en Do While loops
 - While: een loop die wordt uitgevoerd tot een conditie waar is
 - Do While: een loop die wordt uitgevoerd tot **EN MET** een conditie waar is
 - Conditie voorbeeld ($i < 5$)
- For loop
 - Loop wordt uitgevoerd zolang een conditie waar(true) is
 - 3 onderdelen: uitgangspunt, de conditie, increment of decrement

To the code! – Loops.java & Collections.java

48



Collections.java

```
@Test
public void workWithArrays(){
    //create the array - mandatory length
    String[] gamesArray = new String[5];

    //fill the array
    gamesArray[1] = "Zelda";
    gamesArray[3] = "Tekken";

    // print the array - is there a smarter way?
    System.out.println(gamesArray[0]);
    System.out.println(gamesArray[1]);
    System.out.println(gamesArray[2]);
    System.out.println(gamesArray[3]);
}
```

49



Collections.java

```
@Test
public void workWithLists(){
    //create the List - check the imports!
    List<String> gamesList = new ArrayList<>();

    //fill the List
    gamesList.add("Super Mario World");
    gamesList.add("Luigi's Mansion");

    //print the list - is there a smarter way? -> Loops.class
    System.out.println(gamesList.get(0));
    System.out.println(gamesList.get(1));
    //System.out.println(gamesList.get(2));
}
```

50



Loops.java

```

@Test
public void workWithWhileLoop(){
    //start value
    int i = 1;

    //create the while loop
    while (i < 5){
        System.out.println("current value: " + i);
        //increment to prevent infinite loop
        i++;
    }
}

@Test
public void workWithDoWhileLoop(){
    //start value
    int i = 1;

    //create the Do While loop
    do {
        System.out.println("current value: " + i);
        //increment to prevent infinite loop
        i++;
    }
    while (i < 5);
}

```

51



Loops.java

```

@Test
public void workWithForLoop(){
    //create the for loop
    for(int i = 1; i < 5; i++){
        System.out.println("current value: " + i);
    }
}

@Test
public void listIter(){
    //create the List - check the imports!
    List<String> gamesList = new ArrayList<>();

    //fill the List
    gamesList.add("Super Mario World");
    gamesList.add("Luigi's Mansion");
    gamesList.add("Zelda");
    gamesList.add("Street Fighter");
    gamesList.add("Zelda");

    for(int i = 0; i < gamesList.size(); i++){
        System.out.println(gamesList.get(i));
    }

    for(String game : gamesList){
        System.out.println(game);
    }
}

```

52



Webelements

- Selenium kan ook eerdere elementen in een lijst “zetten” voor ons
- `List<WebElement> allOptions = driver.findElements(By.xpath("./option"));`

53



Testcase 6.05 – Suppliers validatie, Apple store

- Opdrachten bijlage -> 6.05
- Correcte Git branch?
- Maak gebruik van **WebElements**
- Tip voor SelectBox:

```
Select dropdown = new Select(driver.findElement(By.id("identifier")));  
dropdown.selectByVisibleText("VisibleText");
```

54



Testcase 6.05 – Suppliers validatie, Apple store

Vragen?

Git add / commit / push

55



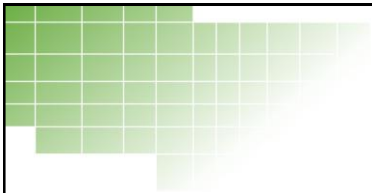
Merging time...

- Merge de branch met master (bijv pull request via de site)
- "git checkout master"
- "git pull"



56





Vragen?