

Labo 3 – GET met parameters en Selenium

In dit labo leer je parameters tussen de client en de server door te geven met GET methode en leer je Selenium testen gebruiken.

Voorbereiding

- Installeer een recente versie van Chrome
- Voeg in pom.xml een dependency naar Junit 4 toe (GEEN junit.jupiter!)

```
<dependency>
  <groupId>junit</groupId>
  <artifactId>junit</artifactId>
  <version>4.13.1</version>
  <scope>test</scope>
</dependency>
```

- Voeg in pom.xml een dependency naar Selenium toe

```
<dependency>
  <groupId>org.seleniumhq.selenium</groupId>
  <artifactId>selenium-java</artifactId>
  <version>3.141.59</version>
</dependency>
```

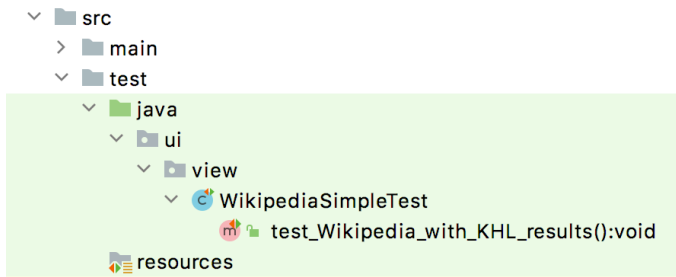
- Download Chromedriver
 - o De versie van Chromedriver moet overeenstemmen met de versie van Chrome. Als Chrome automatisch updates uitvoert, zal je binnen enkele weken een nieuwe versie van Chromedriver moeten downloaden.
 - o <https://sites.google.com/a/chromium.org/chromedriver/downloads>
 - o Onder het kopje "Current Releases"
 - o Bewaar in een map die je later gemakkelijk terugvindt

Opdracht 1: Seleniumtest

Op Toledo vind je een zip met daarin enkele Seleniumtesten. Run de test "ValidHtmlTest" in een IntelliJ-project naar keuze.

Voer hiervoor onderstaande stappen uit:

- Maak in de directory "test" een nieuw package aan dat je ui.view noemt.



- Kopieer naar deze map de gegeven Selenium test (in bovenstaande screenshot was het WikipediaTest)
- Open de testklasse. Importeer JUnit (alt+Enter).
- Pas het pad naar Chromedriver aan.
- Run de test.

Opdracht 2: StudentenInformatiesysteem

Deze opdracht bouwt verder op labo 2b. Maak een kopie van je oplossing en werk verder op de kopie.

We willen een webapplicatie maken waar we op basis van de naam en de voornaam, de leeftijd en de studierichting van een student kunnen *opvragen*.

Je hebt de volgende dataset gemaakt in labo 2b:

Naam	Voornaam	Leeftijd	Studierichting
Jongen	Greetje	23	Toegepaste Informatica
Melaerts	Kristien	21	Chemie
Steegmans	Elke	16	Vroedkunde
Van Hee	Jan	18	Verpleegkunde

In tegenstelling tot labo 2b gaan we de lijst met studenten nu niet hardcoderen in de jsp pagina, maar maken we een db klasse aan. De studenten worden vervolgens hardgecodeerd toegevoegd als arraylist in de constructor van deze klasse.

1. Maak het klassendiagram (klassen Student en StudentDB) voor de studenten op papier.
2. Maak de nodige klassen aan in Java (package `domain.model` en `domain.db`).
3. Maak een servlet (package `ui.controller`) zodat je met de URL <http://localhost:8080/StudentInfo?naam=steegmans&voornaam=elke> leeftijd en studierichting kan opvragen.
4. Maak een JSP-pagina (gevonden.jsp) die een webpagina met het resultaat van het GET-request genereert.
Het resultaat van het voorbeeld in puntje 3 moet een paragraaf tonen met daarin de volgende tekst: "Je vroeg naar volgende gegevens: Steegmans Elke (16 jaar): Vroedkunde"
5. Indien je een student ingeeft die niet voorkomt in de tabel, dan kom je op een andere jsp pagina uit (nietgevonden.jsp) waarop de volgende boodschap verschijnt: "Helaas, de student waarnaar je vraagt is niet gevonden."

6. Maak een JSP-pagina (zoek.jsp) met een formulier met velden “naam” en “voornaam”. Als gebruiker op submit klikt, stuurt de browser een http-GET-request naar de server zoals omschreven in puntje 3. De invoer van de gegevens wordt afgehandeld zoals omschreven in puntjes 4 en 5.
7. Run de test “StudentInformatieTest”. Pas je code aan totdat hij groen kleurt.
8. Pas de overzichtspagina aan zodat ze de db-klasse gebruikt om de reeds ingevoerde studenten te tonen. De overzichtspagina spreekt de servlet NIET aan (leerstof voor volgende week).

Opdracht 3: Doorgeven parameters met GET

Deze opdracht is een extra oefening.

We willen een webapplicatie maken waar we kunnen laten tellen hoeveel keer een bepaalde letter in een gegeven woord voorkomt. Woord en letter worden expliciet meegegeven in de querystring van de URL, dus niet via een formulier.

De URL moet er als volgt uitzien

<http://localhost:8080/Occurences?woord=papa&letter=a>

- Maak een Servlet in het package `ui.controller`.
- Schrijf de java-code die nodig is om het aantal letters te tellen in de Servlet.
 - o Tip: gebruik de methode `charAt(i)` van de klasse `String` en gebruik ook `Character.toString` om een char om te zetten naar een String
- Geef de berekende waarde door aan een jsp-bestand (`result.jsp`) dat het resultaat toont. Zorg voor valide html-code.
- De berekende waarde moet in een paragraaf staan met id "antwoord" en leest als volgt: "Het antwoord is: 2 keer."

Gebruik de gegeven Selenium testen die je van ons krijgt om je opdracht te testen in plaats van alles handmatig zelf te testen.