class: inverse, center, middle

# Felületi tesztelés Seleniummal

class: inverse, center, middle

## Webes alkalmazás

## Webes alkalmazások

* Böngésző a kliens
  + Microsoft Internet Explorer, Microsoft Edge
  + Mozilla Firefox
  + Google Chrome
* Mivel böngésző mindenütt telepített, megszűnnek a telepítési/frissítési problémák
* Vékony kliensnek is nevezik, utalva arra, hogy a logika főleg szerver oldalon van
  + Elegendő csak ott frissíteni
* Háttérben a HTTP, HTML, CSS, JavaScript technológiák

## URL

* Uniform Resource Locator
* Interneten található erőforrások (szöveges tartalmak, képek, hangfájlok, videók) egyedi azonosítására
* Felépítése
  + Protokoll
  + Tartománynév
  + Port
  + Elérési út

## HTTP(S) protokoll

* 1999-ben kiadott RFC 2616 definiálja a HTTP/1.1-et (W3C szervezet)
* 2015-ben leváltott a HTTP/2.0-ás verzió, amit az RFC 7540 definiál
* Kliens-szerver kommunikáció
* Kliens tipikusan böngésző
* Kérés-válasz alapú protokoll
* Szöveges
* Fejléc, törzs
* Állapotmentes
* (S): secure - SSL alapú titkosítás
* Eszközök: Böngésző Fejlesztői eszköztár

## HTTP kérés

Accept: text/html,application/xhtml+xm…plication/xml;q=0.9,\*/\*;q=0.8  
Accept-Encoding: gzip, deflate  
Accept-Language: hu-HU,hu;q=0.8,en-US;q=0.5,en;q=0.3  
Cache-Control: max-age=0  
Connection: keep-alive  
Cookie: \_ga=GA1.2.1967894445.149906973…yis3b41advsb3cwc3b3rk; \_gat=1  
Host: training360.com  
Upgrade-Insecure-Requests: 1  
User-Agent: Mozilla/5.0 (Windows NT 10.0; …) Gecko/20100101 Firefox/57.0  
  
GET http://training360.com/ HTTP/1.1

## HTTP válasz

HTTP/1.1 200 OK  
Access-Control-Allow-Headers: range  
Access-Control-Allow-Origin: \*  
Cache-Control: private  
Content-Encoding: gzip  
Content-Length: 16217  
Content-Type: text/html; charset=utf-8  
Date: Thu, 14 Dec 2017 10:19:08 GMT  
Server: Microsoft-IIS/7.5  
Vary: Accept-Encoding  
X-AspNet-Version: 4.0.30319  
X-Powered-By: ASP.NET  
  
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN">  
<html xmlns="http://www.w3.org/1999/xhtml">  
<head>  
 <!-- ... -->  
</head>  
<body>  
 <!-- ... -->  
</body>  
</html>

## Hivatkozott erőforrások

* CSS
* JavaScript
* Képek (tipikusan gif, jpg, svg formátumban)
  + Formátumok közötti különbségek

## Státuszkódok

* 200 OK
* 302 Found, lsd. később
* 304 Not Modified, lsd. később
* 400 Bad Request
* 401 Unauthorized
* 403 Forbidden
* 404 Not Found
* 500 Internal Server Error

## HTTP paraméterek

* Kérdőjellel (?) elválasztva
* Egymástól & karakterrel elválasztva
* URL encoding

http://training360.com/search.aspx?filter=javax&date=2017.12.01.&chkIT=true&chkOffice=true

http://training360.com/search.aspx?filter=alma%20k%C3%B6rte&date=0&chkIT=true&chkOffice=true

## Cache

* Cache-Control
  + Cache-Control fejléc (mp-ben), az adott tartalmat meddig lehet tárolnia lokálisan a böngészőnek
* If-Modified-Since
  + Kliens küldi, kéri a szervert, hogy csak akkor adjon vissza tartalmat, ha módosult azóta
  + 304 Not Modified

## Mime-type

* Content-Type header
* Néhány: text/plain, text/html, text/javascript, text/CSS, image/gif, image/jpeg, application/vnd.ms-excel (vnd - vendor - nem szabványos)

http://www.training360.com/download.aspx?src=tanfolyam\_jelentkezes\_Training360.doc

Content-Type: application/msword

## Űrlapok

* POST metódus

POST /contact.aspx HTTP/1.1  
Host: http://training360.com  
User-Agent: Mozilla/5.0 (Windows NT 10.0; …) Gecko/20100101 Firefox/57.0  
  
UserName=demo@example.com&Password=password

## Cookie

* Szerver hozza létre
* Böngésző tárolja, és minden kéréskor visszaküldi a szervernek
* Csak ugyanannak a szervernek küldheti vissza
* Használata:
  + Pl. nyelv, pénznem (miben írja ki a webáruház az árakat)
  + Session követés (lsd. később)

## Állapotmentesség, session

* Két kérés között nincs kapcsolat
* Session (munkamenet): "felhasználónkénti" szerver oldali tárterület egyedi azonosítóval
* Azonosítót leküldi a szerver cookie-ban, melyet a kliens minden kéréssel visszaküld
* Timeout

## Átirányítás

* GET paraméterek, hossz limit
* POST kezelése, felmerülő problémák:
  + Refresh
  + Back gomb használata
  + Könyvjelző
* Redirect after post tervezési minta

## HTML formátum

* HyperText Markup Language
* W3C szabvány
* Szöveges formátum
* Szöveges tartalom reprezentálására
* Struktúra és tartalom (hisztorikusan megjelenés is)
* Hiperhivatkozások
* Fej, törzs
* Fa hierarchia
* Tagek

## HTML karakterkódolás

* HTML5-ben az alapértelmezett karakterkódolás az UTF-8

## Egyszerű HTML dokumentum

<!DOCTYPE html>  
<html>  
 <head>  
 <title>Example</title>  
 </head>  
 <body>  
 <h1>Example Page</h1>  
 <p>This is an example page.</p>  
 </body>  
</html>

## DOM ábrázolás

DOM

DOM

## CSS

* Cascading Style Sheets
* W3C szabvány
* Színek, méretek, elhelyezkedések, betűtípusok
* Elválik a tartalom és a forma
* Könnyebben változtatható a megjelenés
* Szelektorokkal kiválaszható elemeknek adható meg formátum
* Megadható a HTML-be ágyazva, fejlécben és külön állományban is

## CSS példa

<h1>A CSS használata</h1>

h1 {  
 text-align: center;  
 color: red;  
 font-size: large;  
 font-family: "Times New Roman", serif;  
 font-style: italic;  
}

## Beágyazott CSS

<h1 style="color: red;">A CSS használata</h1>

## CSS a fejlécben

<html>  
 <head>  
 <style>  
 h1 {  
 color: red;  
 }  
 </style>  
 </head>  
 <body>  
 <h1>A CSS használata</h1>  
 </body>  
</html>

## Hivatkozás külső CSS állományra

<html>  
 <head>  
 <link rel="stylesheet" href="styles.css" />  
 </head>  
 <body>  
 <h1>A CSS használata</h1>  
 </body>  
</html>

## Selenium

* Böngésző automatizálás, tipikusan webes alkalmazások tesztelésére
* Eszközkészlet, több eszközből áll

Selenium logo

Selenium logo

## Selenium tulajdonságai

* Platformfüggetlen (Windows, Apple OS X, Linux, pl. Ubuntu)
* Képes meghajtani a különböző böngészőket is (Firefox, Internet Explorer, Safari, Opera, Chrome)
* Nyílt forráskódú, ingyenes

Selenium logo

Selenium logo

## Felhasználási területek

* Funkcionális tesztelés
* GUI tesztelés
* Regressziós tesztelés
* Böngészőfüggetlenség tesztelése

Selenium felhasználási területek

Selenium felhasználási területek

## Eszközök

* Selenium IDE: felvétel és visszajátszás grafikus felületen
* Selenium WebDriver: böngészővezérlés (pl. programozási nyelvekből API-n keresztül)
* Selenium Grid: automatizált tesztelés több gépről, összehangoltan

Selenium tools

Selenium tools

## Selenium IDE

* Felvétel és visszajátszás (tesztesetek lépésekkel)
* Chrome vagy FireFox Add-On
* Programozási nyelv funkciók: parancskészlet, paraméterezés, változók, vezérlési elemek
* IDE funkciók, pl. projektkezelés, autocomplete, debug
* Parancssori futtatási lehetőség

## Selenium telepítése (gyakorlat)

* https://addons.mozilla.org/en-US/firefox/addon/selenium-ide/
* https://chrome.google.com/webstore/detail/selenium-ide/mooikfkahbdckldjjndioackbalphokd

## Selenium IDE indítása

* Új projekt létrehozása, és felvétel
* Létező projekt megnyitása
* Új projekt létrehozása
* Selenium IDE bezárása

## Welcome screen

Welcome screen

Welcome screen

## Első teszt felvétele és visszajátszása (gyakorlat)

* http://tadev.training360.com/testapp-web
* Egy teszt végigvitele: Answer questions
* Eredmény ellenőrzése: Jobb klikk/Selenium IDE/Assert text

## Selenium IDE képernyő

Selenium IDE

Selenium IDE

## Futtatás

* Base URL
* Run all tests
* Run current test
* Execution speed
* Log ablak

## Debug

* Breakpoints (Toggle breakpoint - B)
* Disable all breakpoints
* Stop test execution
* Resume test execution
* Play to this point
* Play from here

## Fájlkezelés

* Project név
* Create new project
* Open project
* Save project

## Tesztek szervezése

* Tests (test cases)
* Rename, Duplicate, Delete, Export
* Test suites
* Add tests, Rename, Delete, Settings, Export
* Executing

## Új teszt felvétele (gyakorlat)

* New test (név: List results)
* Record
* Eredmény ellenőrzése: Jobb klikk/Selenium IDE/Assert text

## Teszt lépések

* Navigator
  + Command (Enable/Disable, Add new window configuration)
  + Target
  + Value
  + Description

## Gyakori parancsok

* open
* type
* click
* assert, pl. assert title, assert text - ha hamis, leáll a futás
* verify, pl. verify title, verify text - ha hamis, warning
* https://www.seleniumhq.org/selenium-ide/docs/en/api/commands/
* Reference ablak

## Lépések szerkesztése

* Cut, Copy, Paste, Delete
* Sorrendezés drag n' droppal
* Insert new command
* Clear all command
* Record from here

## Komponensek kijelölése

* Formátuma: locatorType=location
* https://www.guru99.com/locators-in-selenium-ide.html
* Id alapján
* Name alapján (filterelhető)
* Link
* CSS
  + Developer tools/Inspector/Copy CSS selector
* XPath
  + Developer tools/Inspector/Copy XPath

## Locator Selenium IDE támogatás

* Selenium IDE: Select target in page, Find target in page
* Egyszerre több locatort is megjegyez, amelyik sikerül

## CSS selector

* id attribútum alapján: css=tag#id, pl. css=input#name-input
* CSS class alapján: css=tag.class, pl. css=input.btn, vagy tag nélkül .btn
* Attribútum érték alapján css=tag[attribute=value], pl. css=input[type=submit]

## XPath

* W3C szabvány
* Egy XML dokumentum elemei és attribútumai közötti navigációt biztosítja
* XPath szintaktika segítségével definiálhatjuk az XML dokumentum részeit
* Kifejezések segítségével mozoghatunk az XML dokumentumban
* https://codebeautify.org/Xpath-Tester

## XPath példák

* /html
* /html/body
* //input
* /html/@lang

## XPath predicates 1.

* /html/table/tbody/tr[1] - első tr
* /html/table/tbody/tr[last()] - utolsó tr
* /html/table/tbody/tr[last() - 1] - utolsó előtti tr
* /html/table/tbody/tr[position() < 3] - első két tr

## XPath predicates 2.

* //input[@id"] - van id attribútuma
* //input[@id="name-input"] - id attribútumának értéke name-input

## XPath ismeretlen csomópontok

* /html/body/\* - összes gyerek
* //\* - összes elem
* //\*[@id="name-input"] - összes tag, megadott attribútummal

## Legjobb gyakorlatok

* Körültekintően nevezzük el a teszteket
* Selectornál támaszkodjunk az id értékekre, fejlesztők támogatása szükséges
* id elnevezés legyen konzekvens (elnevezési konvenció)
* Idempotens és izolált
* Tesztesetek legyenek egyszerűek
* DRY - don't repeat yourself
* Használjuk Continuous Integration rendszeren belül

## Idempotencia és izoláltság

* Tesztesetek egymásra hatással vannak
  + Állapot: pl. adatbázis
* Ugyanazon tesztkörnyezeten több tesztelő vagy harness dolgozik
* Megoldás:
  + Teszteset "rendet tesz" maga előtt, un. set-up
  + "Rendet tesz" maga után, un. tear down
  + Test fixture
    - Legszélsőségesebb megoldás: adatbázistörlés

## Új komplex teszt (gyakorlat)

* Egy tesztkitöltés rögzítése, valamint az eredmények megtekinítése

## Másik teszteset futtatása

* run parancs

## DRY megszüntetése (gyakorlat)

* Answer questions hívása run parancs segítségével

## Változók használata

* store [érték] [változónév]
* echo [érték] parancs, változóra: ${változónév}

## Változó újrafelhasználása (gyakorlat)

* A név értéke legyen a name változóban

## XPath kifejezés értéke változóba (gyakorlat)

* A results oldalon a táblázat sorainak számát kell ellenőrizni

## Saját alkalmazáspéldány elindítása (gyakorlat)

* séma inicializálás

create schema if not exists testdb default character set utf8 collate utf8\_hungarian\_ci;  
  
create user 'testuser'@'localhost' identified by 'testpwd';  
grant all on \*.\* to 'testuser'@'localhost';

* testapp-web indítása (java -jar testapp-web.jar)
* Base URL módosítása

## Control flow

* Elágazás és ciklus
* https://www.seleniumhq.org/selenium-ide/docs/en/introduction/control-flow/

## Ciklus (gyakorlat)

* Módosítsd úgy a Answer questions teszt esetet, hogy kétszer válassza ki az első választ!
* Tipp: módosítsd úgy a locatort (XPath-ra), hogy mindig az első gombra klikkeljen

## Command-line Runner

* Node.js (https://nodejs.org/en/download/)
* selenium-side-runner: npm install -g selenium-side-runner
* geckodriver: npm install -g geckodriver

(Windowson letöltendő és PATH-ba helyezendő: https://github.com/mozilla/geckodriver/releases)

Futtatás

* selenium-side-runner -c "browserName=firefox" testapp.side

## Selenium WebDriver

* Böngészővezérlés programozottan
* Különböző programozási nyelvekhez illesztés: C#, Groovy, Java, Perl, PHP, Python, Ruby and Scala
* Driver böngészőnként (Firefoxhoz geckodriver, https://github.com/mozilla/geckodriver)

## WebDriver

WebDriver

WebDriver

## Selenium Javaval

* Selenium IDE képes exportálni

## Selenium Pythonnal

* https://selenium-python.readthedocs.io/

## Új projekt

* requirements.txt állományba selenium
  + Install requirements
* Új könyvtár létrehozása
  + .py Ptyhon állományok létrehozása

## Első teszteset

from selenium import webdriver  
  
driver = webdriver.Firefox(executable\_path="C:\Java\geckodriver-v0.20.1-win64\geckodriver.exe")  
driver.get("http://localhost:8080/")  
assert "Test Application" in driver.title  
  
# driver.close()

## Input

nameInput = driver.find\_element\_by\_id("txtUserName")  
nameInput.clear()  
nameInput.send\_keys("vicziani")  
  
button = driver.find\_element\_by\_xpath("/html/body/form/button")  
button.click()

## Elem szövegének vizsgálata

secondAnswerLabel = driver.find\_element\_by\_xpath("/html/body/form/label[2]")  
print (secondAnswerLabel.text)  
assert "1b válasz" in secondAnswerLabel.text

## Újrafelhasználás, page object

## Adatvezérelt tesztelés

## Ismétlő kérdések

* Milyen komponensekből áll a Selenium? Melyiknek mi a feladata?
* Mi a Selenium fő célja?
* Milyen területeken használható a Selenium?
* Mire való a Selenium IDE?
* Milyen szoftverként jelenik meg, hogy kell telepíteni?
* Hogyan lehet a legegyszerűbben egy tesztet elkészíteni?
* Milyen debugging lehetőségek vannak benne?
* Milyen gyakori tesztlépések vannak?
* Mi az a lokátor? Hogyan lehet megadni?

## Ismétlő kérdések 2.

* Mik a teszteléssel kapcsolatos legjobb gyakorlatok?
* Hogy lehet a kódismétlést minimalizálni?
* Hogyan lehet változót deklarálni és használni?
* Milyen vezérlési szerkezetek vannak?
* Hogyan lehet parancssorban futtatni teszteseteket?
* Mire való a WebDriver?
* Milyen programozási nyelveket támogat?
* Milyen lépéseket ismersz?
* Milyen tervezési mintát ismersz?