



BATCH : ALUMNI
LESSON : INTERVIEW PREP
DATE : 21.09.2022
SUBJECT : SELENIUM



techproeducation



techproeducation



techproeducation



techproeducation



techproedu

SELENIUM INTERVIEW HAZIRLIĞI

1



3 ADIMDA

OFFER

INTERVIEW PREP

HER
PAZARTESİ Java

HER
ÇARŞAMBA Selenium

HER
CUMA SDLC & Softskill



16.00 EST / 22.00 CEST / 23.00 TSI





Bu hafta neler var?

1. **Selenium Nedir?**
2. **Selenium bileşenleri hakkında bilgi veriniz**
3. **Selenium'un avantajları ve dezavantajları nelerdir?**
4. **Selenium'da hangi testler otomate edilir, hangileri edilmez.**
5. **Locator nedir? Kaç farklı locator vardır?**
6. **Siz daha çok hangi locatorı tercih edersiniz?**
7. **Xpath ve CssSelector arasındaki farkları izah eder misiniz?**
8. **Absolute ve Relative xpath arasındaki farklar nelerdir?**
9. **Findelement ve Findelements arasındaki farklar nelerdir?**
10. **Selenium'da Page Object Model nedir?**



1. SELENIUM NEDİR?

Selenium, **web uygulamalarını test etmek için** kullanılan açık kaynaklı bir otomasyon suiti'dir.

Peki! Selenium neden bir **framework değil de suite** olarak adlandırılır.

Çünkü birden fazla framework kullanımı hatta bunların hibrit mantıkla bir arada kullanımına olanak sağladığı ve bu bağlamda farklı görevler üstlenen araçlara sahip olduğu için.

Mesela Cucumber framework'ü inşa ettiğimiz zaman hem JUnit hem de TestNg kütüphanelerinden istifade ederiz ve BDD yani davranış güdümlü geliştirme modelinin bir yansıması olan Gherkin Language'i kullanırız.



2. SELENIUM BİLEŞENLERİ HAKKINDA BİLGİ VERİNİZ

Dört tane bileşene sahiptir.

Selenium IDE (Selenium **Integrated Development Environment- entegre geliştirme ortamı):** Selenium IDE bir Firefox eklentisidir. Selenium Suite'teki en basit çerçevedir. Komut dosyasını kaydetmemizi ve oynatmamızı sağlar. (**İntelij) Selenium IDE'yi chrome browsera eklenti olarak ekleyebilir ve kullanabilirsiniz**

Selenium RC (Selenium Remote Control): Birden fazla testin sürekli olarak yapılmasını sağlar. Bir Selenium RC sunucusu ile çalışır. (**Selenium RC'nin karmaşık mimarisi ve Javascript komutları kullandığı için işleyişi oldukça yavaştır. Bu sebeple Selenium WebDriver geliştirilmiştir 2006'da**)



- **Selenium WebDriver:** Komutları kabul eden ve bunları bir tarayıcıya gönderen bir tarayıcı otomasyon çerçevesidir. Tarayıcıya özgü bir sürücü aracılığıyla uygulanır. Doğrudan onunla iletişim kurarak tarayıcıyı kontrol eder.
(Asıl kullandığımız tool)
- **Selenium Grid:** Selenium Grid, farklı makinelerde paralel olarak farklı tarayıcılarda testler yapmak için Selenium RC ile birlikte kullanılan bir araçtır.
(Jenkins)



3. SELENIUM'UN AVANTAJLARI

***Bu soru, “Neden Selenium’u tercih edersiniz” şeklinde de sorulabilir”**

Acik kaynakli olmasi, genis bir kullanici kitlesi ve bunlarin bilgi paylasim portallari mevcuttur. (ücretsizdir)

Birçok browser için destek sağlaması, (chrome, mozilla, edge vs.)

Farklı programlama dilleri ile çalışması, (Java, Python, C++, C#, PHP, Perl vs.)

Tüm işletim sistemleri ile uyumludur. (Microsoft, Linux, Mac)



4. SELENIUM'UN DEZAVANTAJLARI

Bu soru, **Selenium'da otomate edilemeyen nesneler** şeklinde sorulabilir.

Programlama dili bilmeniz gerekir.

Sadece web tabanlı uygulamalar ile çalışır. (Appium ile birlikte mobil testing yapılabilir)

Performans testleri yapmaz, daha çok fonksiyonel testlere odaklanır

Handle captcha mümkün değildir.



Açık kaynaklı olduğu için profesyonel bir müşteri hizmetleri yoktur.



5. SELENIUM İLE HANGİ TESTLER OTOMATE EDİLİR?

- Functional Tests (positive/negative, User Interface)
- Integration Tests
- Regression Tests
- Smoke/Sanity Tests
- End-to-end (System) Testing
- Data Driven (Excel tablolarından bilgi çekme gibi)



6. SELENIUM İLE OTOMATE EDİLMİYEN TEST TÜRLERİ

- Selenium sadece fonksiyonel test türlerini otomate ettiği için fonksiyonel olmayan testleri otomate edemez.
- Yani **performans**, **stres**, **load** test gibi testler kapsam dışıdır.
- Yine dinamik testlerin dışında kalan static testler de Selenium'un kapsamı dışındadır.
- Manuel ve Unit testler de Selenium'un dışındadır.
- Ayrıca şunu ilave edelim ki database testing yapılacaksa genellikle Selenium'a ihtiyaç olmaz.
- Örneğin çok yaygın olarak kullanılan API, Selenium'dan bağımsızdır.



7. LOCATOR NEDİR

Locator, (yer bulucu) web sayfası içinde unique bir web elementi tanımlayan bir adres olarak tarif edilebilir.

Bu sayede Selenium'da **farklı locator tiplerini** kullanarak doğru ve isabetli bir şekilde bir web elementi tanımlayabiliriz.

Web elementin doğru tanımlanması ilgili otomasyon case'nin başarılı bir şekilde execute edilmesini sağlayacaktır.

```
...  
::before  
▼<div class="row">  
  ::before  
  ▼<main id="main" class="site-main col-sm-12 full-width">  
    ▼<div id="lp-single-course" class="lp-single-course learn-press-4">  
      ▼<div id="learn-press-course" class="course-summary learn-press">  
        ▼<div class="course-info">  
          ...  
          <h1 class="entry-title" itemprop="name">Free Java</h1> == $0  
          ▶<div class="course-meta">...</div>  
          ▶<div class="course-navment">...</div>  
        ... anddrop.websockets.rgb.hsla.multiplebgs.backgroundsize.borderimage.borderradius.boxshadow.textshadow.op  
        //main[@id="main"]
```



8.LOCATOR ÇEŞİTLERİ

•Sekiz farklı locator vardır.

By.id() = driver.findElement(By.id(**"add-to-cart-button"**));

By.name() = driver.findElement(By.name(**"field-keywords"**));

By.tagName() = driver.findElements(By.tagName(**"a"**));

By.className() = driver.findElements(By.className(**"s-image"**));

By.linkText() = driver.findElement(By.linkText(**"Addresses"**));

By.partialLinkText() = driver.findElement(By.linkText(**"dresses"**));

By.xpath()= driver.findElement(By.xpath(**"//input[@id='twotabsearchtextbox']"**));

By.cssSelector()=driver.findElement(By.cssSelector(**"input[id='twotabsearchtextbox']"**));



9. Testlerinizi Otomate Ederken Genellikle Hangi Locator'ı Tercih Edersiniz

- Bir otomasyonda locate olarak ilk akla gelen **id**'dir.
- Çünkü **id** sistemlerin unique olarak atadığı benzersiz bir kullanıcı numarasıdır. Biz de locator alırken unique olmasına dikkat ederiz.
- Unique olmayınca aldığımız web elementin adresi net olmayacak ve otomasyon yaparken hangisinin handle edilmesi gerektiğine otomasyon tool'u karar veremeyecektir.
- Ancak her HTML kodu düzenli olmadığından **id** ile locate alma işlemi her zaman gerçekleşmez. **(Bu noktada projenizden bir örnek vermeniz iyi olacaktır)**
- O zaman diğer locator'lara bakarız. Burada web elementi bulma noktasında en garanti olduğunu düşündüğüm xpath ile locate almayı tercih ederim. Elbette duruma göre uygun olursa name, tagname veya diğer locator'lar ile de bu işlemi yapmak mümkündür.



10. XPATH VE CSS_SELECTOR ARASINDAKİ FARKLAR NELERDİR

CSS_SELECTOR	XPATH
Xpath'e göre hızlıdır	Css selector'e göre yavaştır.
Text () syntax'ini desteklemez	Text () syntax'ini destekler
Parent child ilişkisine sahip olmadığı için geriye dönük hareket edemez bu sebeple dinamik değildir.	Parent child ilişkisine sahip olduğu için geriye dönük hareket edebilir. Bu sebeple dinamiktir. Dinamik olduğu için CssSelector'e göre elementi bulma konusunda daha garantidir.
groupIndex'i desteklemez	groupIndex destekler



11. ABSOLUTE VE RELATİVE XPATH ARASINDAKİ FARKLAR NELERDİR

ABSOLUTE XPATH,

HTML kodlarındaki hiyerarşi kullanılarak parent/child silsilesi yazılıp istenen elemente ulaşmak için kullanılır.

HTML kodları dinamik olduğundan, developerlar sürekli ekleme veya çıkarmalar yapabilirler.

Absolute xpath yazıldıktan sonra hiyerarşideki ekleme veya çıkarmalar kodumuzun istenen webElementi bulamamasına sebep olur.

Bu sebeple kullanımı tavsiye edilmez.

For Example:

`html/body/div[1]/section/div[1]/div/div/div/div[1]/div/div/div/div/div[3]/div[1]/div/h4[1]/b`



RELATIVE XPATH

Yaygın kullanım olarak tag, attribute ve value bilgilerinin unique sonuc olusturacak sekilde birlesmesiyle olusur.

Relative Xpath için path (öge yolu), HTML DOM yapısının ortasından başlar.

Uzun xpath yazmanıza gerek yoktur.

Çift eğik çizgi (//) ile başlar, yani ögeyi web sayfasının herhangi bir yerinde arayabilir. O element degistirilmedikce relative xpath de degismez.

Relative XPath'in faydası daha çok kullanılabilir olması ve daha az bakım/maliyet gerektirmesidir.

Sadece, relative xpath için **index kullandığımızda** HTML kodlarında eklenecek veya cikarılacak webelementler xpath'i etkileyebilir.

Örnek:

```
By.xpath ( ) = driver.findElement(By.xpath("//input[@id='twotabsearchtextbox'] [1]"));
```



Select current
node

Selects
Attribute

Value of the
attribute

`Xpath=//tagname[@Attribute='Value']`

Tagname like Input,
Div, Img etc.

Attribute
Name



12. FİNDELEMENT VE FİNDELEMENTS ARASINDAKİ FARKLAR NELERDİR

- Her iki method da **WebDriver interface**'nin **abstract methodudur** ve bir web sayfasında WebElement bulmak için kullanılır.
- **findElement ()**; web sayfasında bir adet elementi bulmak için kullanılır ve sadece bir WebElement tipi döndürür. WebElement'i bulamazsa exception (**NoSuchElementException**) fırlatır.
- **findElements()**; birden fazla web element bulmak için kullanılır ve WebElements listesi döndürür. Şayet elementleri bulamazsa exception fırlatmaz boş liste döndürür.



13. SELENIUM'DA PAGE OBJECT MODEL (POM) NEDİR? TARİF EDER MİSİNİZ?

POM, Selenium'da bir çeşit dizayn pattern (tasarım modeli)'dir.

Bu tasarım modeli, yaygın olarak ortaya çıkan yazılım problemlerinin çözümü için bir standart ortaya koyar.

Bu standart ile clean kod yazımı, daha az kod, daha kullanışlı bir tasarım ve zaman ile paradan tasarruf sağlanabilir.

Naming Convention gereği doğru isimlendirmeler ile takım çalışmasında başka bir arkadaşımızın da kolaylıkla bu tasarımı kullanmasını sağlar ve daha sürdürülebilir bir framework meydana getirebiliriz.



Bunu daha da somutlaştırmak için POM'a uygun bir framework'te nasıl dizayn ettiğimizden de bahsetmek yerinde olacaktır.

POM temelli bir framework'te (örneğin TestNG ya da Cucumber) pages, tests ve utilities package'ları oluşturularak bu packageler içine ilgili classlar yerleştirilir.

Utilities package'ında driver'ımızı sürekli kullanacağımız için Driver classı, testlerimizin raporlarını almak için Test Base class'ı, **configuration properties dosyası ile test classlarımız arasında yine POM mantığı gereği key value alışverişi yapacak olan Config Reader classı** ve gerekirse bize kolaylık sağlaması için bir Reusable Method Classı koyarız.

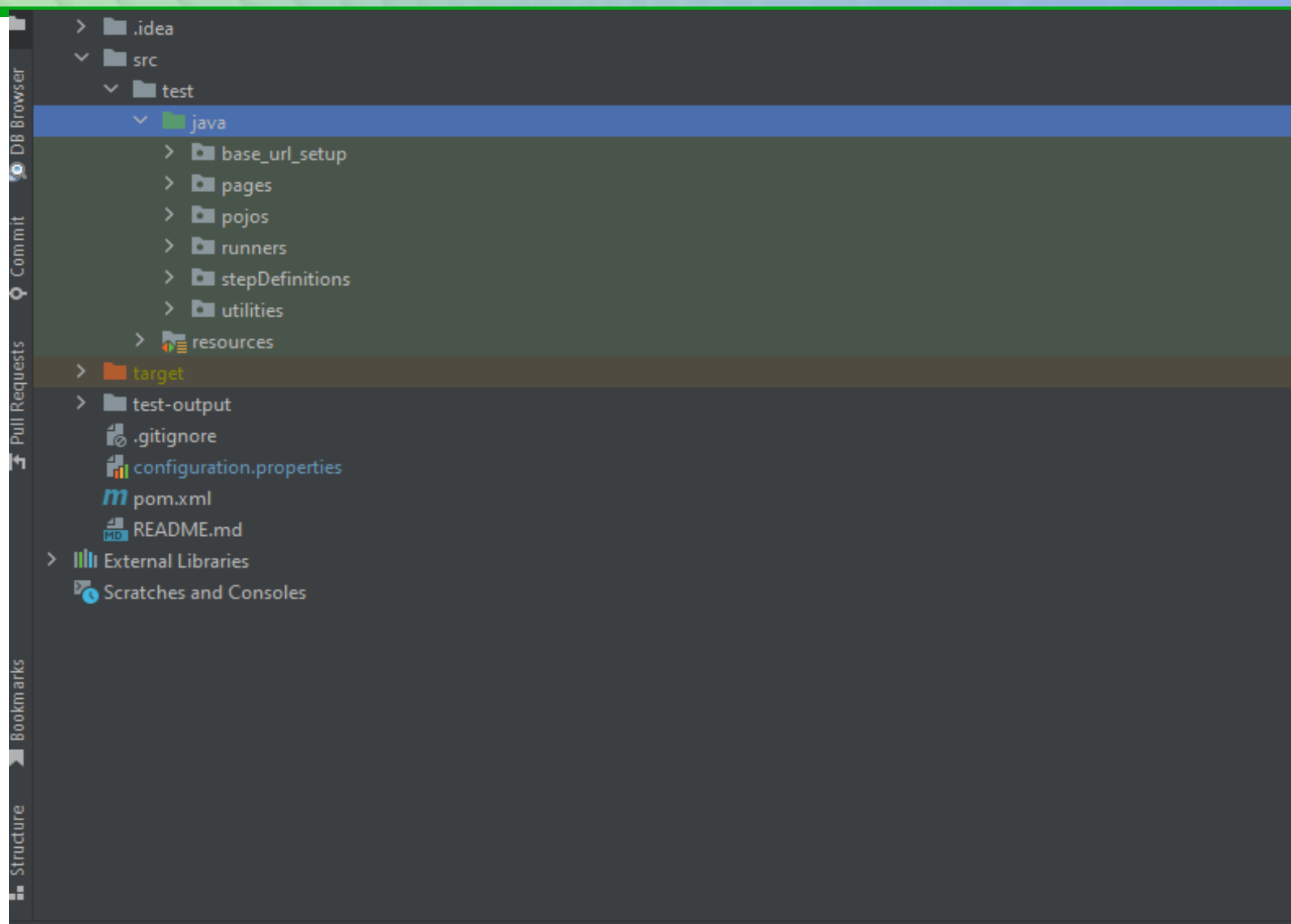


Pages package'ı içine çalışacağımız web uygulamasına ait web elementlerin yer aldığı classı ve test package'ı içine de testlerimizi koşacağımız classları koyarız.

Test classları içinde gereksinimlerimize göre Driver classı ve page classlarını obje yardımı ile çağırır ve daha reusable bir şekilde testlerimizi execute edebiliriz.

Özetle POM ile dizayn edilmiş framework, kodların daha reusable ve daha readable olmasını sağladığı gibi değişikliklerin daha kolay ve hızlı bir şekilde uygulanabilir olması dinamik bir framework ile çalışma imkânı da sağlar.

Böylece zamandan ve paradan da tasarruf edilebilir. (işverenin en sevdiği şey 😊)





EĞLENCE KÖŞESİ 😊

**DUYGULARINIZ SİZİ HER YERE
MANTIĞINIZ İSE SADECE
GEREKLİ OLAN
YERLERE GÖTÜRÜR!**

•TESTER ÖZLÜ SÖZÜ 😊