

```
// tek satırlık yöntemler
a = a ^ b ^ (b = a);
b = (a + b) - (a = b);
a += b - (b = a);

int temp = a; // geçici değişken
a = b; b = sıcaklık;
```

52. Yazmayı biliyor musunuz? Oyuncu seçimi nedir?

- **Otomatik kutulama** → ilkel bir değer alıp sarmalayıcı sınıf nesnesine `int i = 10` atadığımız bir süreçtir;

```
Tamsayı n = i;
Tamsayı num = 200;
Tamsayı num2 = new Integer (400); // BOXING YOK
```

- **Un-boxing** → Wrapper sınıf nesnesini alıp ilkele dönüştürdüğünüz bir süreçtir.

```
Tamsayı num2 = yeni Tamsayı (400);
Tamsayı num = 200;
int i = num2;
```

- Bir türden bir değerın başka türden bir değışkене atanması, Type Casting olarak bilinir.

53. Bu programın çıktısı nedir?

```
for (int i = 0; i < 3; i++) {
    for (int j = 3; j >= 0; j--) {
        eğer (i == j)
            devam et;
        System.out.println (i + " " + j);
    }
}
```

Çıkış: 1 0 2 3 2 1 2 0

54. Projenizde soyut bir sınıfı nasıl kullanıyorsunuz bana bir örnek verin?

- Bu kavramlar genellikle çerçeve geliştirmede kullanılır. Soyut sınıf, ortak bir süper sınıfı tanımlamada kullanılır
çerçevenin Sayfa Nesne Modeli katmanını yazarken. Genellikle hepsine sahip olmak için `BasePage` adında soyut bir sınıf oluştururuz.
Bu sınıf örneğinde yazılan her sayfa için ortak üyeler `getPageTitle ()` .
- Daha sonra her Sayfa sınıfı (`HomePage`, `LoginPage`, `DashboardPage` vb.) `BasePage`'den devralır. Bazen ihtiyaç duyulabilir
üst sınıfta uygulanan yöntemlerin davranışını değiştirir. Dolayısıyla, alt sınıf, bu yöntemi geçersiz kılma özgürlüğüne sahiptir.
polimorfizm kullanın. `Abstract` sınıfını gerçek projelerde böyle kullanıyoruz.

55. Değere göre geçirme ile referansla geçirme arasındaki fark nedir? değere göre geçmek ve referans ile geçmek?

- Değere göre geçiş, işlev parametresinin değerinin belleğinizin başka bir konumuna kopyalandığı anlamına gelir ve
İşlevinizdeki değışkene erişirken veya değıştirirken, yalnızca kopyaya erişilir / değıştirilir ve orijinal değër dokunulmadan bırakılır. Değere göre geçiş, değërlerin çoğu zaman nasıl aktarıldığıdır.
- Başvuru yoluyla geçiş, değışkenin bellek adresinin (bellek konumuna bir işaretçi),
işlevi. Bu, bir değışkenin değerinin aktarıldığı değere göre geçişten farklıdır. Örneklerde hafıza adresi
/ `myAge 106`'dır. `myAge` işlevini artırırken, işlev içinde kullanılan değışken olan `artışAgeByRef` (yaş
bu örnek) hala orijinal `myAge` değışkeni ile aynı bellek adresine işaret etmektedir (İpucu:
fonksiyon parametresi, bir değışkenin referansını / işaretçisini elde etmek için birçok programlama dilinde kullanılır).

- 57 -

Sayfa 58

SELENYUM

1. Selenyum nedir ve nelerden oluşur?

- Selenyum, otomatik web testi için bir araç paketidir. Tarafından bestelendi;
 - o Selenyum IDE (Entegre Geliştirme Ortamı); kayıt ve oynatma için çalışan bir Firefox eklentisi.
 - o Selenyum RC (Uzaktan Kumanda) (1.0); bir test aracıdır ve web uygulamasını otomatikleştirmek için JS üzerinde çalışmak için kullanılır. (2004)
 - o WebDriver (2.0); bir web otomasyon çerçevesidir ve testlerinizi farklı tarayıcılarda yürütmenize olanak tanır. (2011)
 - o Selenyum Izgara; testlerin birden çok makinede paralel olarak yürütülmesine izin verir.

2. Selenyum'un avantajları nelerdir?

- Selenyum açık kaynak kodludur ve herhangi bir lisans maliyeti olmaksızın kullanımı ücretsizdir
- Java, Ruby, Python, C # gibi birden çok dili destekler ...
- Çoklu tarayıcı testini destekler
- İyi miktarda kaynağı vardır ve topluma yardım eder
- Windows, Mac, Linux gibi birçok işletim sistemini destekler ...
- Web uygulamasıyla etkileşim kurun

3. Selenium'un dezavantajları nelerdir?

- Selenium yalnızca web tabanlı uygulamaları destekler, Windows tabanlı uygulamaları desteklemez
- Yerleşik raporlama aracı yoktur, rapor oluşturma etkinliği için üçüncü taraf araçlara ihtiyaç duyar
- Grafikler, captcha'lar, barkodlar, şekillerle çalışamaz
- Dosya yükleme özelliğini desteklemez.
- Ustalaşması zor, geliştirici düzeyinde bilgi gerektirir
- İyi konum belirleyicileri yazmak zor
- Senkronize etmek zor

4. Selenium'un sınırlamaları nelerdir?

- Masaüstü uygulamasını test edemiyoruz
- Web hizmetlerini test edemiyoruz
- Test çerçevesi (TestNG, JUnit),
harici dosyalar (excel için Apache POI)
- Captcha'yı Selenium kullanarak otomatikleştirmek mümkün değildir
- Dosya yükleme özelliğini desteklemez.

5. Selenium ile hangi tür testleri otomatikleştiriyorsunuz?

- fonksiyonel testler (pozitif / negatif, UI)
- duman testleri
- regresyon testleri
- entegrasyon testleri
- uçtan uca test
- veri tabanlı

6. Selenium ile ne yapmıyoruz?

- Performans, yük, stres testi, manuel geçici test (Bu testler, bu araçlarda eğitim almış uzmanlar tarafından yapılır)
- Saf veritabanı testi (yalnızca veritabanının kendisini test edersek),
- Birim testleri ..., görünüm ve his temelli testler (renk, şekiller, vb.),
- statik test

- 58 -

Sayfa 59**7. Selenium araç setinde neler var?**

- Selenium IDE → bir Chrome ve Firefox uzantısı olarak uygulanır ve testleri kaydetmenize, düzenlemenize ve hata ayıklamanıza olanak tanır.
- Selenium RC → herhangi bir programlama dilinde otomatik web uygulaması UI testleri yazmak için
- Selenium WebDriver → testlerinizi farklı tarayıcılarda gerçekleştirin
- Selenium GRID → testlerinizi farklı makinelerde paralel olarak farklı tarayıcılarda çalıştırın.

8. Şu anda Selenium'un hangi sürümünü kullanıyorsunuz?

- JDK (JAVA) - 1.8 → → Lambda exp nedeniyle beğendim. ve hata işlemeyi deneyin, birden çok yakalama ekleyebilirsiniz.
- IntelliJ - 2018.03.04
- Selenium - 3.141.59
- TestNG - 6.14.3
- Salatalık - 4.2.6
- Maven - 3.6.0
- GIT - 2.17.2

9. Örtülü Bekle mi Açık Bekle mi?

- **Örtülü bekleme** , fırlatmadan önce bir öğeyi bulurken belirli bir süre bekleyen bir beklemedir
"NoSuchElementException". Varsayılan olarak selenyum, öğeleri beklemeden hemen bulmaya çalışır. Bu yüzden iyi örtük bekleme kullanmak için. Bu bekleme, geçerli sürücü örneğinin tüm öğelerine uygulandı.
- **Açık bekleme** , belirtilen Beklenen Koşul karşılanıncaya kadar belirli bir web ögesine uygulanan beklemedir.
- Örtük bekleme basittir; koşul, zaman aşımından önce karşılanırsa, koşul içinde karşılanmazsa bir sonraki adıma geçer.
zaman aşımı "Böyle Bir Öğе Yok" istisnasını atar.
- Açık bekleme bazen ögenin görünür, tıklanabilir, etkin olması gibi belirli bir olayı / koşulu beklememiz gerekir ...

```
driver.manage().timeouts().örtük olarak Bekleme (5, TimeUnit.SECONDS);
webDriverWait wait = new WebDriverWait (sürücü, 5);
wait.until (ExpectedConditions.visibilityOf (element));
```

10. fluentWait nedir?

- Diyelim ki bazen sadece 1 saniyede görünen ve bir süre sonra ortaya çıkması dakikalar alan bir ögeniz var. Şöyle akıcı beklemeyi kullanmak daha iyidir, çünkü bu öğeyi bulana kadar veya son zamanlayıcı çalışana kadar tekrar tekrar bulmaya çalışacaktır. dışarı. Örnek AJAX veya JQuery'dir
- Açık beklemenin alt türü ancak koşulları geçersiz kılabilirsiniz

```
<WebDriver> bekleyin = yeni
FluentWait <WebDriver> (sürücü) .withTimeout (5, TimeUnit.SECONDS) .pollingEvery (100, TimeUnit.SECONDS) .ignoring (NoSuchElementException.class);
```

11. Selenium'da bir elementi bulmanın çeşitli yolları nelerdir?

- Selenium Konumlandırıcılar → Kimlik ve ad
- Selenium bulucuda, html'de bir öge bulmanın bir yolu:
- Id, name, className, xpath, css, linkText, partialLinkText, tagName

12. Elementi neden bulamıyorum?

- Konum belirleyici değiştirildi
- Bir iframe var
- Bekleme süresi :: sayfa yavaş yükleniyor veya Öge dinamik :: locator
- Sayfa tam olarak yüklenmemiş / açılmamış
- Sayfa değişir ve bu öge artık mevcut değildir

- 59 -

Sayfa 60

13. Bir öge nasıl vurgulanır?

- Selenium WebDriver'in vurgulama eylemi yoktur.
- Ancak bunu yapmak için JavaScript kullanabiliriz

```
JavaScriptExecutor js = ((JavaScriptExecutor) sürücüsü);
Dize bgcolor = element.getCssValue ("backgroundColor");
için (inti = 0; i < 10; i++) {
    changeColor ("rgb (0,200,0)",
        eleman, sürücü); // 1
    changeColor (bgcolor,
        eleman, sürücü); // 2
}
```

14. Xpath nedir?

- Xpath, html yapısını kullanarak bir web sayfasındaki herhangi bir ögenin konumunu bulmak için kullanılır.
- Metin kutusu gibi web Öğelerini bulmak için bir XML belgesindeki öğeler ve nitelikler arasında gezinebilirdik.
- buton. checkbox, Image ext ... in web Page

15. Mutlak (/) ve Göreli (//) Xpath?

- Sözdizimi → // etiketadı [@ özellik = "değer"]
- Mutlak xpath, kök öğeden başlayarak öğeye kadar tek eğik çizgiyle (/) başlar.
- Göreli xpath, belgenin herhangi bir yerinde seçimle eşleşen çift eğik çizgi (//) ile başlar.

16. Dinamik unsurları nasıl idare ediyorsunuz?

- id'nin statik kısmını bulun ve bir konum belirleyici yazın (xpath veya css) → Ve ardından Startswith, contains, EndsWith kullanın
- içerir () → // * [içerir (@ name = 'btn')]
- ile başla () → // etiket [startswith (@id, 'message')]
- Metin () → // td [text () = 'usedId']
- veya & ve → // girdi [@type = 'submit' AND @name = ' login']

17. Dinamik web sayfası nasıl test edilir?

- Bu soruna tüm çözümlere uyan tek bir çözüm yoktur. Uygulamayı çok iyi anlamalısınız
 - o Gerektiğinde açık beklemler kullanın.
 - o Özel xpath'leri ve css bulucularını kullanın
 - Xpath: içerir, şununla başlar, ile biter, metin içerir.
 - Ebeveyn, çocuk, kardeş ilişkilerini kullanarak başka bir kararlı öğeye ilişkin öğeyi bularak

18. Dinamik tablo nasıl test edilir?

- Özel xpath'ler ve css bulucuları kullanın
 - o Xpath: içerir, şununla başlar, ile biter, metin içerir.
 - o Ebeveyn, çocuk, kardeş ilişkilerini kullanarak öğeyi başka bir kararlı öğeye ilişkili olarak bularak
- Tablo ile çalışan yardımcı yöntemlerim var. Bir tablo web elemanını alan ve tüm sütunu döndüren bir yöntemim var isimler. Bir tabloyu, numarayı alan ve o satırdaki tüm verileri döndüren bir yöntemim var.

19. xpath kullanarak üst öğeye nasıl geçebiliriz?

- xpath'de (..) ifadesini kullanarak, üst öğeye geçebiliriz

20. close () ve quit () komutu arasındaki fark nedir?

- driver.close () → mevcut tarayıcıyı kapatmak için kullanılır
- driver.quit () → tüm tarayıcı örneklerini kapatmak için kullanılır

- 60 -

Sayfa 61

21. xpath kullanarak n'inci alt elemana nasıl geçebiliriz?

• İki yol vardır:

o dizin konumunda köşeli parantez kullanarak

Örneğin: div [2], ikinci div ögesini bulacaktır

o position () yöntemini kullanarak

Örneğin: div [position () = 2], ikinci div ögesini bulacaktır

22. xpath ve css seçici arasındaki fark nedir?

• xpath ile öğeleri geriye veya ileriye doğru arayabiliriz ...

css yalnızca ileri yönde çalışır

• Xpath metinle çalışabilir, css çalışamaz

• Xpath'in daha fazla kombinasyonu vardır ve dizine göre arama yapılabilir

css dizine göre arama yapamaz, ancak css xpath'den daha hızlı çalışır.

23. Çerçeve nedir?

• Test otomasyonunda çerçeve, test otomasyonunun taslağıdır.

• İşlev kitaplığınızı, test sonuçlarınızı, test verilerini, kaynakları nereye kaydedeceğinizi klasör yapılarınızı içerir.

• Bu çok önemlidir çünkü bir otomasyon projesi üzerinde çalışırken herkesin takip etmesi gereken bir kılavuz olacaktır ve bizim

komut dosyasının bakımı daha kolay olacaktır.

24. Görüşme sırasında HTML haberciliğinden mi bahsediyorsunuz?

• Çerçevemde birden fazla raporlama yöntemi kullanıyorum, sürücü betiği test durumları excel sayfasına geçti / kaldı,

• Reporter yardımcı programı nesnesi UFT raporuna yazıyor, ayrıca özel bir HTML raporlama motoru geliştirdim.

• Not Deferine HTML kodu gönderir ve teknik bilgisi olmayan kişilerin kolayca yapabileceği güzel bir HTML rapor belgesi oluşturur.

anlayın ve kullanın.

25. Bir web sayfası nasıl büyütülür?

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

26. Bazı durumlarda, maximize () çalışmaz> öyleyse yol nedir?

• İşlemler yapın veya sürümü değiştirin.

ChromeOptions seçenekleri = yeni ChromeOptions ();

options.addArguments ("startmaximized");

27. Selenium'daki anahtar sınıfı nedir?

• Bize klavyeden tuşlara basma seçeneği sunar

• Anahtar.ENTER

• SendKeys () yöntemine GEÇİLMELİDİR

• Örn; .sendKeys ("şarj cihazı" + keys.ENTER)

28. Ya rastgele açılan dinamik bir açılır pencere varsa

• Uyarı ile dene / yakala özelliğini kullanın

29. Thread.sleep () nedir?

• Yakalamak için selenyum yavaşlatır

• İstisna fırlatır, bu yüzden onu ele almalı veya fırlatmalıdır

- 61 -

Sayfa 62

30. Selenium Çerçeve nedir?

• Kod bakımını kolaylaştırmaya, kod okunabilirliğine ve kodun yeniden kullanılmasına yardımcı olan bir kod yapısıdır.

• Selenium WebDriver tarafından test durumlarını otomatikleştirmek için oluşturulmuş başlıca 3 tür çerçeve vardır:

Veriye Dayalı Çerçeve

• Piyasadaki en popüler otomasyon çerçevelerinden biridir

• Tüm test verilerimiz bazı harici dosyalardan oluşturulur;

o excel

o veya özellik dosyasında senaryo taslağı

o veya TestNG Veri Sağlayıcısı

• Selenium WebDriver, web tabanlı uygulamaları otomatikleştirmek için harika bir araçtır. Ancak okuma ve yazmayı desteklemiyor

excel dosyalarındaki işlemler. Bu nedenle, Apache POI gibi üçüncü taraf API'leri kullanıyoruz

Anahtar Kelime Odaklı Çerçeve

- Anahtar kelimeye dayalı test, uygulamayla ilgili anahtar kelimeleri içermek için veri dosyalarını kullanan bir komut dosyası oluşturma tekniğidir test edilmek.
- Anahtar sözcükler excel dosyası gibi bazı harici dosyalara yazılır ve Java kodu bu dosyayı çağırır ve test durumlarını yürütür.

HybridDriven Çerçeve

- DDF ve KDF'nin bir kombinasyonunun genellikle HDF olduğu söylenir.
- Hem test verileri hem de test eylemi harici dosyalarda saklanır.

31. Selenium'da aşırı yüklenmiş Yöntemleri nasıl kullandınız?

- İki değerin eşit olup olmadığını iddia ederken, → TestNG'den Assert.assertEquals (gerçek, Beklenen) kullanıyorum
- String, Objects, int, boolean değerleri parametrelerini girebilirsiniz

32. Neden NoSuchElementException alıyoruz?

- Konumlandırıcının doğru olup olmadığını kontrol edin
- Zamanlamanın doğru olup olmadığını kontrol edin
- Öğenin bir iframe içinde gizli olup olmadığını kontrol edin

33. js uyarılarını nasıl ele alıyorsunuz?

- Tarayıcıdaki uyarı JavaScript'ten geliyorsa, Alert sınıfını kullanınız.

```
Uyarı uyarısı = driver.switchTo.alert ();
alert.accept ();
alert.dismiss ();
alert.sendKeys ();
alert.getText ();
```

34. Birden çok çerçeve nasıl kullanılır?

- 4 kare varsa, belirli bir kareye ulaşmak için arka arkaya geçmeniz gerekir. 3. kareye atlayamıyorum
- 1. kareden.

35. driver.get () ve driver.navigateto () arasındaki fark nedir?

- driver.get () → Bir URL'yi açmak için ve tüm sayfa yüklenene kadar bekleyecektir
- sürücü. () 'E gidin → Bir URL'ye gitmek için ve tüm sayfanın yüklenmesini beklemeyecektir

- 62 -

Sayfa 63**36. Selenium'da çerçeveler nasıl kullanılır?**

- Bir html sayfasını başka bir sayfaya yerleştirmek için kullanılan çerçeveler
 - Adımlar
 - o iframe'i bulun
 - o driver.switchTo (). frame () ile başka bir iframe'e geçin;
 - .frame () → string, Integer, webElement, name veya id'yi doğrudan parametre olarak alır
- ```
driver.switchTo (). çerçeve (webElement);
driver.switchTo (). frame ();
```

- 2. karenin dışında (şu anda bulunduğunuz) bir öge bulmak istiyorsanız, şimdi 2. karedesiniz

- NosuchElementException oluşturur
- Önceki kareye geri dönmek gerekir
  - o driver.switchTo (). parentFrame () → Bir seviye yukarı gider
  - o driver.switchTo (). defaultcontent () → En üste gider
- Count kullanarak geçiş yapabilir
  - o driver.switchTo (0) → Varsayılan çerçeve olmayan her şeyi sayar

*Bu yöntemler, kullandığınız tarayıcıya bağlı olarak size farklı sonuçlar verebilir.*

**37. Tarayıcı açılır pencerelerini nasıl idare ediyorsunuz?**

- **void dismiss ()** → açılır pencere görünür görünmez “İptal” düğmesine tıklar.
- **geçersiz kabul et ()** → açılır pencere görünür görünmez “Tamam” düğmesine tıklar.
- **String getText ()** → uyarı kutusunda görüntülenen metni döndürür.
- **void sendKeys (String stringToSend)** → belirtilen dize modelini uyarı kutusuna girer.

**38. Windows / OS açılır pencerelerini nasıl kullanıyorsunuz?**

- Selenium, Windows tabanlı uygulamaları desteklemez, yalnızca web uygulaması testini destekleyen bir otomasyon test aracıdır.
- AutoIT, Robot sınıfı gibi bazı üçüncü taraf araçları kullanarak Selenium'da Windows tabanlı açılır pencereleri işleyebildik
- **driver.getWindowHandle ();** Bu, onu bu sürücü örneğinde benzersiz bir şekilde tanımlayan mevcut pencereyi işleyecektir.

- **driver.getWindowHandles ();** Tüm açık pencereleri işlemek için

### 39. Başsız tarayıcı nasıl kullanılır?

- Başsız tarayıcı: açılmayan tarayıcı, arka plan hizmeti / programı olarak çalışır.
- Örnek selenyumdan htmlunitdriver
  - o WebDriver = yeni htmlunitdriver ()
  - o Çok kararlı değil
- Phantomjsbrowser
  - o Daha kararlı
  - o tarayıcı = yeni phantomjsbrowser ()

### 40. findElement vs findElements?

- FindElement> bu yöntem ilk WebElement'i döndürür!
  - o öge bulunamazsa İstisna verir
- FindElements> List <WebElement> döndürür;
  - o Sonuç listesinde bulunmayan eleman boş değerlere sahipse o istisna vermez

- 63 -

## Sayfa 64

### 41. Birden çok pencere / sekme nasıl kullanılır?

- Selenyum tek pencerede kalır
  - Bir pencere açarsanız ve ardından 5 sekme açılırsa, selenyum ilk pencereye odaklanır
  - Yeni bir penceredeyseniz ve selenyum'a varsayılan pencerede bir öge yazdırmasını söylerseniz, yine de çalışacaktır.
    - o kullanıcının odağı yeni penceredir
  - Yeni pencereye geçmeli
    - o windowHandle () kullan
    - o Driver.getWindowHandle ()
      - Selenyum her tarayıcı açıldığında, Kollar adlı sayfa için bir dizin kimliği verecektir.
      - Geçerli sayfanın tutamacını / kimliğini döndürür (bir dize olarak)
    - o driver.switchTo (). window (dize tutamacı)
    - o Birden çok pencere için driver.getWindowHandles ()
      - Bir dizi pencere tutacağı döndürür
    - o Başlıkları kullanarak geçiş yapın
- ```
for (string handle: driver.getWindowHandles ()) {
    driver.switchTo (). Pencere (tanıtıcı)
    eğer (driver.getTitle (). equals (targetTitle) {
        kırmak;
    }
}
```

42. Sayfadaki tüm bağlantılar nasıl bulunur?

```
List <WebElement> list = driver.findElements (By.tagName ("a"));
```

43. isDisplayed (), isEnabled () arasındaki fark. Selenium WebDriver'da Selected () yöntemi mi?

- isDisplayed () → web sayfasındaki bir web öğesinin varlığını doğrulayın. Bulunursa → doğru, Bulunmazsa → yanlış
- isDisplayed () → mevcut her türlü web öğesinin varlığını kontrol edin
- isEnabled () → web öğesinin web sayfasında etkin mi yoksa devre dışı mı olduğunu doğrulayın.
- isEnabled () → öncelikli olarak düğmelerle kullanılır
- isSelected () → web öğesinin seçilip seçilmediğini doğrular
- isSelected () → radyo düğmeleri, açılır menüler ve onay kutuları ile kullanılır.

44. Nasıl Sürükleyip Bırakılır?

```
Eylemler eylem = yeni Eylemler (sürücü);
action.clickAndHold (driver.findElement (By.id ("öge")))
.moveToElement (driver.findElement (By.id ("hedef")))
.release (). build ()
.perform ();
```

45. Aşağı kaydırmak için:

```
WebDriver sürücüsü = yeni ChromeDriver (); JavascriptExecutor jse = (JavascriptExecutor) sürücüsü;
jse.executeScript ("window.scrollTo (0,250)", "");
```

- VEYA, aşağıdaki gibi yapabiliriz:


```
jse.executeScript ("kaydırma (0, 250);");
```

46. Yukarı Kaydırma için:

```
jse.executeScript ("window.scrollTo (0, -250)", ""); VEYA, jse.executeScript ("kaydırma (0, -250);");
```

- 64 -

Sayfa 65

47. Tanımlama bilgileri nasıl kullanılır?

- Selenium Webdriver'da, aşağıdaki yerleşik yöntemle çerezleri sorgulayabilir ve bunlarla etkileşim kurabiliriz:

```
driver.manage ().getCookies (); // Tüm Çerezlerin Listesini Döndür
driver.manage ().getCookieNamed (arg0); // İsme göre belirli bir çerez döndür
driver.manage ().addCookie (arg0); // Çerezi oluşturun ve ekleyin
driver.manage ().deleteCookie (arg0); // Belirli çerezleri silin
driver.manage ().deleteCookieNamed (arg0); // Belirli bir çerezi Ada göre silin
driver.manage ().deleteAllCookies (); // Tüm çerezleri sil
```

48. Tanımlama bilgilerini neden işlememiz gerekiyor?

- Her tanımlama bilgisi bir ad, değer, etki alanı, yol, son kullanma tarihi ve güvenli olup olmadığına ilişkin durumla ilişkilendirilir. Sırayla

Bir istemciyi doğrulamak için, bir sunucu tüm bu değerleri bir tanımlama bilgisinde ayrıştırır.

- Selenium web sürücüsünü kullanarak bir web uygulamasını test ederken, bir tanımlama bilgisi oluşturmamız, güncellememiz veya silmemiz gerekebilir.
- Örneğin, Çevrimiçi Alışveriş Uygulamasını otomatikleştirirken, birçoğumuz yer siparişi gibi test senaryolarını otomatikleştirmemiz gerekir, Sepeti Görüntüle, Ödeme Bilgileri, sipariş onayı vb.
- Çerezler depolanmazsa, yukarıda listelenen test senaryolarını yürütmeden önce her seferinde oturum açma eylemi gerçekleştirmemiz gerekecektir. Bu, kodlama çabanızı ve yürütme sürenizi artıracaktır.
- Çözüm, tanımlama bilgilerini bir Dosyada saklamaktır. Daha sonra, çerez değerlerini bu dosyadan alın ve ona mevcut tarayıcınızı ekleyin. oturum, toplantı, celse. Sonuç olarak, her Test Durumunda oturum açma adımlarını atlayabilirsiniz çünkü sürücü oturumunuzda bu bilgiler bulunur.
- Uygulama sunucusu artık tarayıcı oturumunuzu kimliği doğrulanmış olarak kabul eder ve sizi doğrudan istediğiniz URL'ye götürür.

49. Çerez çerez bilgilerini saklayın

```
public class cookieRead, BasePage'i genişletir {
    public static void main (String [] args) {
        driver.get ("http://demo.guru99.com/test/cookie/selenium_aut.php");

        // E-posta kimliği ve Parola Girin Zaten Kaydolduysanız
        driver.findElement (By.name ("kullanıcı adı")).sendKeys ("abc123");
        driver.findElement (By.name ("şifre")).sendKeys ("123xyz");
        driver.findElement (By.name ("submit")) tıklayın ();

        Dosya dosyası = yeni Dosya ("Cookies.data"); // çerezleri saklamak için dosya oluştur

        Deneyin {
            file.delete (); // Varsa eski dosyayı silin
            file.createNewFile ();
            FileWriter fileWrite = new FileWriter (dosya);
            BufferedWriter Bwrite = new BufferedWriter (fileWrite);

            // çerez bilgilerini almak için döngü
            için (Çerez ck: driver.manage ().getCookies ()) {
                Bwrite.write ((ck.getName () + ";" + ck.getValue () + ";" + ck.getDomain () + ";" + ck.getPath ()
                    + ";" + ck.getExpiry () + ";" + ck.isSecure ());

                Bwrite.newLine ();
            }

            Bwrite.flush (); Bwrite.close (); fileWrite.close ();
        } catch (Exception ex) {
            ex.printStackTrace ();
        }
    }
}
```

- 65 -

Sayfa 66

50. Oturum açma bilgileri için saklanan çerezleri kullanın

```
public static void main (String [] args) {
    Deneyin{
        Dosya dosyası = yeni Dosya ("Cookies.data");
        FileReader fileReader = new FileReader (dosya);
        BufferedReader Buffreader = new BufferedReader (fileReader);

        String strline;
        while ((strline = Buffreader.readLine ()) != null) {
            StringTokenizer belirteci = new StringTokenizer (strline, ";");
            while (token.hasMoreTokens ()) {
```

```

        Dize adı = token.nextToken ();
        Dize değeri = token.nextToken ();
        Dize etki alanı = token.nextToken ();
        Dize yolu = token.nextToken ();
        Son kullanma tarihi = boş;

        Dize değeri;
        eğer (! (val = token.nextToken ()) . equals ("null")) {
            süre sonu = yeni Tarih (val);
        }
        Boolean isSecure = new Boolean (token.nextToken (). BooleanValue ());
        Çerez ck = yeni Çerez (ad, değeri, etki alanı, yol, süre sonu, isSecure);
        System.out.println (ck);
        driver.manage (). addCookie (ck); // Bu, saklanan çerezi mevcut oturuma ekleyecektir
    }
}
} catch (Exception ex) {
    ex.printStackTrace ();
}
}
driver.get ("http://demo.guru99.com/test/cookie/selenium_aut.php");
}

```

ÇIKTI: Giriş kullanıcı kimliği ve şifresini girmeden doğrudan oturum açma başarılı ekranına yönlendirilirsiniz.

NOT: Yukarıdaki komut dosyasını çalıştırdıktan sonra oturum açma sayfasını görürseniz, tam yenilemeyi kullanın.

51. JavaScriptExecutor kullanıyor musunuz?

- Bu, kendi JavaScript'imi yazmama yardımcı oluyor. JS'nin selenyumdan çok daha fazla kontrolü var.

- bu sınıfı kullanarak tarayıcıya JS komutları gönderebiliriz

JavaScriptExecutor jsExecutor = (JavaScriptExecutor) sürücüsü;

o executeScript (); komutu yerine getirir

o Parametrenin içinde JS kodunu koyduğunuz yerdir

- jsExecutor.executeScript ("alert ('UYARI: Bu gereksiz bir mesajdır');" → Bu kod bir JS açılır penceresi açar

- Ayrıca 2 parametre de koyabilirsiniz: .executeScript ("js kodu", öğe);

o Kaydırma için kullanılır (selenyum kaydırmada iyi değildir, terimler üzerinde çalışırken bir meydan okuma olduğunu söyleyebilirsiniz.

ve devam düğmesine tıklamadan önce sayfayı okumanız gereken durum sayfası.

o Selenyum ve eylemler sınıfını kullanmayı denediğimde işe yaramadı, bu yüzden javaexecutor kullandım) ve bir öğeye tıkladım;

Sayfa 67

52. Ögenin mevcut / görünür / etkin / olup olmadığı ve metnin mevcut olup olmadığı nasıl kontrol edilir?

- Eleman Var olup olmadığını kontrol etmek için:

```

eğer (sürücü.findElements (By.xpath ("değer")). size () != 0) {
    System.out.println ("Öge Mevcut");
} Başka{
    System.out.println ("Öge Yok");
}

```

- veya

```

eğer (driver.findElement (By.xpath ("değer")) != null) {
    System.out.println ("Öge Mevcut");
} Başka{
    System.out.println ("Öge Yok");
}

```

- Görünür'ü işaretlemek için:

```

eğer (driver.findElement (By.cssSelector ("a> yazı tipi")). isDisplayed ()) {
    System.out.println ("Öge Görünür");
} Başka{
    System.out.println ("Öge Görünmez");
}

```

- Etkinleştir'i işaretlemek için:

```

if (driver.findElement (By.cssSelector ("a> yazı tipi")). isEnabled ()) {
    System.out.println ("Öge Etkin");
} Başka{
    System.out.println ("Öge Devre Dışı Bırakıldı");
}

```

- Mevcut metni kontrol etmek için


```
eğer (driver.getPageSource (). içerir ("Kontrol edilecek metin")) {
    System.out.println ("Metin mevcut");
}Başka{
    System.out.println ("Metin yok"); }
```

53. Açılır menüde birden çok seçili değeri nasıl kontrol edebilirim?

- ArabalarListesi seçin = yeni Seç (el)
- carList.getSelectedOptions (): // seçilen seçenekleri bir liste olarak döndürür (List <WebElement>)
- her biri için: carList.getSelectedOptions ()

54. Eylemler sınıfı nasıl kullanılır?

- Eylemler sınıfı, gelişmiş fare ve klavye işlemleri yapmamızı sağlar:
- Fareyi kontrol edin
- Gelişmiş kullanıcı etkileşimleri için yöntemler sağlayan sınıf
 - o Gezinme
 - o Kaydırma
 - o Çift tıklama
 - o Sürükle ve bırak
 - o Sağ tıklama
 - o mix / match operatörleri
- Eylemler eylemi = yeni Eylemler (sürücü)
- Eylem yöntemleri
 - o tıkla ()
 - o gerçekleştirmek ()
 - o dragAndDrop (kaynak, hedef) .perform ()
 - o sendKeys () genellikle kullandığımızdan farklı
 - o tut ()
 - o keydown () build ()
 - o moveToElement (öge)

- 67 -

Sayfa 68

- Sendkeys işlemini farklı elemanlar üzerinde yapalım
- WebElement'tan gelen normal gönderme anahtarları, metin girişi olmayan bir şeye bir istisna atar.
- Uzun yol;
 - actions.moveToElement (kaynak) .clickAndHold (). moveToElement (hedef) .release (). perform ();
- perform () kullanılmadığı sürece eylemler çalışmayacaktır
- Zincirleme yöntemleri kullanıyorsanız, perform () işleminden önce build () kullanmanız gerekir.

55. Çift tıklama eyleminin sözdizimi nedir?

- Eylemler sınıfını kullanarak web ögesine karşı herhangi bir eylem gerçekleştirmek için önce ögeyi bulmamız gerekir:

```
WebElement el = driver.findElement
Eylemler eylemleri = yeni Eylemler (sürücü). Eylemleri gerçekleştir.doubleClick (el) .perform ()
actions.moveTo (el) .perform actions.doubleClick.perform
actions.moveTo (el) .doubleClick (). build.perform ()
```

56. Dosya indirme ve yükleme**• İndir**

- o Selenium'un kendisi dosya indirmelerini doğrulayamaz, indirme bağlantısına tıklayabilir ancak tarayıcının dışına çıkamaz ve indirilmiş dosya
- o Robot ve AutoIT için diğer araçların kullanılması gerekir

• Yükle

- o Yüklemeyi Selenium yönetir, ancak gerçek kullanıcıya göre farklı bir şekilde yapar
- o Adımlar
 - Yükleme penceresini tetikleyen ögeyi bulun
 - Yükleme istediğiniz dosyanın yolunu bulun
- o Bir Dizide Saklayın
 - Örn: Dize → dosya = "C: \\\ Kullanicılar \\\ Andy \\\ Masaüstü \\\ klasör1 \\\ dosya.key";
 - Sonra driver.findElement (yükleme düğmesi) .sendKeys (dosya);

57. Açılır menüden seçilen değer nasıl kontrol edilir?

```
ArabalarListesi seç = yeni Seç (el)
carList.getFirstSelectedOption ()
assertEquals ("bir metin", carList.getFirstSelectedOption (). getText ())
```

58. Select etiketi olmadan açılır menü ile nasıl çalışılır?

- Açılır listede seçme etiketi yoksa, seçili sınıfı kullanamayız
- Açılır listeyi ve seçeneklerini ayrı öğeler olarak değerlendirin, her ögeyi ayrı ayrı bulun
- Bir seçenek seçmek için:
 - o 1. Listeyi bulun ve tıklayın
 - o 2. Seçeneği bulun ve tıklayın

59. Ya seçme etiketi yoksa?

- Açılır liste için etiketi bir web elemanı olarak ayrı ayrı seçmelisiniz.
- Daha sonra manuel olarak tıklama yöntemini kullanın

60. Bazen sendKeys çalışmıyor

- Robot veya AutoIT
- kitaplık == jar dosyası == bağımlılık

- 68 -

Sayfa 69

61. Çerçeve değiştirmek için sözdizimi nedir?

- Çerçeve, başka bir html belgesinin içindeki bir html belgesidir.
- Web sürücüsü bir seferde bir sayfa / html belgesini işler. Başka bir kareyi kontrol etmek için her zaman geçiş yapmalıyız
- Driver.switchTo.frame (webelement) → iframe'i bulun ve param olarak geçirin
- Driver.switchTo.frame (string) → iframe'in kimliğini veya adını bulun ve param olarak geçirin
- Driver.switchTo.frame (int) → dizini bulun ve param olarak geçirin

62. Pencereleeri değiştirmek için sözdizimi nedir?

- Ayrı sekmeleri / pencereleri işlemek için o sekmeye geçmeliyiz
- Web sürücüsü bir seferde bir sayfa / html belgesini işler.
- Başka bir sekmeyi kontrol etmek için her zaman geçiş yapmamız gerekir
- Değiştirebilmek için önce pencere tutamacını kullanmalıyız

```
getWindowHandles () yöntemi sürücüsü.switchTo.window (Dize) // → pencere tutamacı
// her döngü için: driver.getWindowHandles:
Driver.switchTo.window ("tutamaç")
If(driver.getTitle == beklenen başlık;
Kırmak;
```

63. Bir dosya karşıya yüklemek için sözdizimi nedir?

```
Public void fileUpload (String yolu) {
    WebElement karşıya yükleme = sürücü.findElement(); Upload.sendKeys (yol)
}
```

- Yükle düğmesini html'de bulmamız gerekiyor.
- Elemanın etiket girişi olacaktır.
- Daha sonra yüklemek istediğimiz dosyanın yolunu ileterek anahtarları göndeririz

64. Bazen sendKeys / yol çalışmıyor

- Projemizin içindeki bir dosya için dinamik bir yol oluşturma Proje konumuna giden yol:

```
String projectDir = System.getProperty ("user.dir") // proje dizini
Dize dosyası = "src / test / kaynaklar / test_data / dosyam.txt";
Element.sendKeys (projectDir + dosya);
```

65. sendKeys () çağrılmadan metin kutusuna metin nasıl girilir?

```
// Kullan
JavascriptExecutor JS = (JavascriptExecutor) web sürücüsü;
// için giriş Kullanıcı adı
JS.executeScript ("document.getElementById ('Kullanıcı'). Value = 'www.google.com'");
// Parola girmek için
JS.executeScript ("document.getElementById ('pass'). Value = 'tester'");
```

66. Selenium WebDriver'da metin kutusunda ENTER tuşuna nasıl basılır?

- Selenium WebDriver kullanarak Enter tuşuna basmak için,
- Selenium Enum anahtarlarını sürekli Enter ile kullanmamız gerekiyor
- Driver.findElement (By.xpath ("xpath")) sendKeys (Keys.ENTER);

67. Çapraz tarayıcı testi yaptınız mı? tarayıcılar arası test

- Tarayıcı türü, ana url, kullanıcı adı, şifre, ortam gibi anahtar kelimeler için her zaman bir kontrol dosyanız olduğunu belirtin.

- 69 -

Sayfa 70

68. Sertifika sorununu nasıl çözersiniz?

- CHROME, IE → DesiredCapabilities yeteneği = **DesiredCapabilities.chrome** ();
- Jenkins'e → eklememiz gerekiyor . **relaxedHTTPSValidation**

```
Yanıt yanıtı = RestAssured.given ()
.contentType (ContentType.JSON)
```

```
.relaxedHTTPSValidation ()
.get ("https://api.got.show/api/continents");

System.out.println (response.asString ());
```

69. Web Öğesinin sayfadaki konumunu nasıl doğrularsınız?

- `element.getLocation ();`
- `WebElement` sınıfı, öğenin sol üst köşesini döndüren bir `get Location` yöntemine sahiptir

70. Page Factory sınıfı?

- `Page Factory` sınıfı `Selenium` ile birlikte gelir.
- Ve sayfa nesnesi sınıfları oluşturduğumuzda kullanılır.
- Amacı, sınıfta tanımlanan `WebElement`'leri başlatmaktır.

71. Bana salatalıklı yaptığınız test yürütme akışınızı açıklayın.

- `Runner> Feature file> Scenario> Steps> Step def> POM` kullanarak `Selenium` kodu

72. UX ve Restful webServices'i test etmek için hangi araçları kullanıyorsunuz?

- UX → Kullanıcı Deneyimi. Öncelikle UX'in manuel olarak kabul edilebilir olduğundan emin olun.
- Bundan sonra UI testi olduğu için otomatikleştirmek için `Selenium WebDriver` kullanıyorum.
- `RESTFul API Otomasyonu> RestAssured Library`, manuel testler için `PostMan`

73. Selenium WebDriver kullanarak tarayıcı penceresi nasıl yeniden boyutlandırılır?

- Tarayıcı penceresini belirli boyutlara yeniden boyutlandırmak için, tarayıcı penceresini yeniden boyutlandırmak için 'Dimension' sınıfını kullanırız.
- `// Boyutlar sınıfının nesnesini oluşturun`
`Boyut d = yeni Boyut (480,620);`
`// Geçerli pencereyi give n boyutuna göre yeniden boyutlandır`
`driver.manage (). window (). setSize (d);`

74. Selenium'da hangi istisnaları biliyorsunuz?

- Sık sık `NoSuchElementException` yaşıyorum
- `StaleElementException`
 - o Öğe tamamen silindi.
 - o Öğe artık DOM'a eklenmez.
 - o `StaleElementException`'i nasıl ele alırsız;
 - Öğe DOM'a eklenmemiş → 'for döngüsü' içinde 'yakalama bloğunu dene'
 - Veya
 - 1. Sayfayı yenileyin ve aynı öğe için tekrar deneyin.
 - 2. Öğenin kullanılabilir hale gelmesini bekleyin
- `TimeoutException`

75. ASSERT (sert iddia) DOĞRULA KARŞISI (yumuşak iddia)

- Hard assert, bir assert ifadesi başarısız olduğunda hemen bir `AssertException` oluşturur ve test paketi bir sonraki ile devam eder
@Ölçek. Assert adımları başarısız olursa, testin yürütülmesi bu noktada durur! ve varsa bir sonraki teste gidecek!

- 70 -

Sayfa 71

- o (Örnek: sadece basit `Assert.assertTrue (boolean);`)
- Soft assert, `@Test` sırasında hataları toplar Soft Assert, bir onaylama başarısız olduğunda ve devam ederse bir istisna atmaz
assert ifadesinden sonraki adımla. Adımları doğruyla başarısız olursa, bir başarısızlık bildirecek, ancak yürütmeye devam edecektir!
o Örnek: `SoftAssert soft = new SoftAssert (); // yumuşak nesne oluşturma için`
`soft.assertTrue (boolean);`
`soft.assertAll (); // sonunda neyin başarısız olduğunu rapor edecek!`

76. Selenium'da bulunan doğrulama noktası nedir?

- `Selenium IDE`'de, `Selenium Doğrulama` ve `Onaylama Komutlarını Doğrulama noktaları` olarak kullanıyoruz
- `Selenium WebDriver`'da, doğrulama noktaları için yerleşik özellikler yoktur, tamamen bizim kodlama stilimize bağlıdır. Bazı
Doğrulama noktaları
o sayfa başlığını kontrol etmek için
o belirli bir metni kontrol etmek için
o belirli bir öğeyi kontrol etmek için (metin kutusu, düğme, açılır menü vb.)

77. Metnin var olduğunu doğrulayın mı?

- `VerifyTextPresent` → belirtilen metin dizesi sayfada bir yerde BULUNDU ise DOĞRU döndürür; Aksi takdirde YANLIŞ.
- `VerifyTextNotPresent` → belirtilen metin dizesi sayfanın herhangi bir yerinde BULUNAMADI ise DOĞRU döndürür; YANLIŞ, eğer öyleyse bulundu.

78. Bir web sayfasında bir metni nasıl bulursunuz?

- // etiket adı [içerir (metin (), 'metin')] belirli test içerir
- //tagname[.='text'] tam metni içeriyor bazen Selenium çalışmıyor

79. Apple'ın önceki tüm kardeşlerine nasıl sahip olunur?

- Xpath: "// ul / li [içerir (text (), 'Apple Mobiles')] / precedingsibling :: li"
- Bu, "Samsung Mobiles" verecektir

80. Apple'ın aşağıdaki kardeşlerinin tümü nasıl edinilir?

- Xpath: "// ul / li [içerir (text (), 'Apple Mobiles')] / followingsibling :: li"
- Bu, önceki tüm kardeşleri verecektir (Nokia Mobiles, HTC Mobiles, Sony Mobiles, Micromax cep telefonları)

81. Web Tabloları / ızgaraları nasıl kullanılır?

- Tablo verileri için kullanılan tablo etiketi bir ızgara biçiminde düzenlenmiştir

Sütun adı için etiket Örnek -

```
<tr>
  En üst satırda <th> FirstName </th> sütun adları
<th> Soyadı </th>
<th> Yaş </th>
</tr>
```

o </tr> bir satırı belirtmek için kullanılan tr etiketi, bir satırdaki bir sütunu belirtmek için tüm sütuna uygulanır td etiketi Örnek

```
<tr>
  <td> Danny </td> actual_data_on_the_very_first_row
  <td> Smith </td>
  <td> 29 </td>
</tr>
```

- Bazı tablolarda tablonun verilerini göstermek için kullanılır, genellikle sütun adlarını içermez (th)

Sayfa 72

82. Excel nasıl kullanılır?

```
FileInputStream ExcelFile = new FileInputStream (yol);
excelWBook = yeni XSSFWorkbook (ExcelFile);
excelWSheet = excelWBook.getSheet (sayfaAdı);
 hücre = excelWSheet.getRow (rowNum) .getCell (colNum);
```

83. Selenium sürüm 3'ü nasıl buluyorsunuz? Selenium 3, Selenium 2'den büyük ölçüde farklı mı?

- Selenium 3, selenium 2'den hata düzeltmelerine sahiptir ve ayrıca daha mobil otomasyon odaklıdır.
- Selenium 3'ün "mobil ve web uygulamalarının kullanıcı odaklı otomasyonu için bir araç" olmasını hedefliyoruz.
- İşte değişikliğin özeti.
 - o WebDriver kullanıcıları için, daha çok hata düzeltmesi ve 2 yerine bırakma işlemidir.
 - o Selenium Grid hata düzeltmeleri de yapıldı.
 - o Selenium projesi aktif olarak yalnızca WebDriver API'yi desteklemeyecektir.
 - o Zamanlamayla ilgili olarak Mozilla, Firefox'ta değişiklikler yaptı, bu da Firefox 48'den bunları kullanmanız gerektiği anlamına gelir.
 - geckodriver, Selenium 2 kullanıp kullanmadığınıza bakılmaksızın o tarayıcıyı kullanmak için
 - o Bildiğimiz gibi Selenium 3.0, Selenium Jar'ın en son sürümüdür