Note: "Break" statement can only useful with looping

If on friday your Test.xml works properly but on sunday it shows error then what
would be the problem?
1. Version of browser may updated.
2. Internet Speed or connectivity problems.
3. Popups may arrived
4.

Which type of exception you faced in selenium?
1. No such element
2. Webdriver exception
3. timeout excception
4. no such alert
5. stale element exception

if any feature works properly in testing envoirnment but not works in production
envirnment then what would be the issue? how would you do root cause analysis?
1. Might be flowing data is one of the resion, Testing envoirment and production
envoironment have different setup data.
2. There may be some modules not in testing envoirnment but in production
envoirnment it is present so that code may conflict with current code .
3. Load testing not performed correctly so as in testing envoironment no of
users less compare to production envoirment.

How do you click on an element which is hidden using Selenium WebDriver?
We can click on an element which is hidden with Selenium webdriver. The hidden
elements are the ones which are present in the DOM but not visible on the page.
Mostly the hidden elements are defined by the CSS property
style="display:none;".
Selenium by default cannot handle hidden elements and throws
ElementNotVisibleException while working with them. Javascript Executor is used
to handle hidden elements on the page. Selenium runs the Javascript commands
with the executeScript method. The commands to be run are passed as arguments to
the method.

First of all, the getElementById method can be used to identify the element.
Next to enter text to the field, the value method is used to set value to the
field.

```
WebDriver driver =
LounchBrowser.openChromeBrowser("https://courses.letskodeit.com/practice");
    driver.manage().timeouts().implicitlyWait(5, TimeUnit.SECONDS);
    // identify element and click
    driver.findElement(By.id("hide-textbox")).click();
    // Javascript executor class with executeScript method
    JavascriptExecutor j = (JavascriptExecutor) driver;
    // identify element and set value
    j.executeScript
("document.getElementById('displayed-text').value='Selenium';");
    String s = (String) j.executeScript("return
document.getElementById('displayed-text').value");
    System.out.print("Value entered in hidden field: " +s);
```

```
        driver.close();
```

Value entered in hidden field: Selenium

Can we go to parent element from child element by xpath?
Yes, by using
        xpath of child element/parent:tagname of parent

How do you find webElement by CSS selector?
By using xpath we can up in document (So we can move to the parent element) but
by CSS we can only down in document
Syntax
cssSelector=("HTML tag[attribute=Value of attribute]")
findElement(By.cssSelector("input[id='session_key']"));

What you got when driver.get(Url)?
We hit URL

How you check checkbox in selenium?
after click on checkbox we can check it by isSelected method

How you can refresh the page?
driver.navigate().To(driver.getCurrentUrl());
driver.get(driver.getCurrentUrl());
action.sendKeys(keys.F5);
action.perform();

How to find brockenlinks?
```java
public static void main(String[]args) throws IOException
{
        System.setProperty("webdriver.chrome.driver", "E:\\chromedriver.exe");
        WebDriver driver = new ChromeDriver();
        driver.manage().window().maximize();
        driver.navigate().to("http://www.zlti.com");

        List <WebElement> Links=driver.findElements(By.tagName("a"));//or use
By.xpath("//a"));
        Iterator<WebElement>iterate= Links.iterator();
        while(iterate.hasNext())
        {
                String urllink= iterate.next().getAttribute("hraf");
                System.out.println(urllink);
                URL url = new URL(urllink);
                URLConnection urlConnection = url.openConnection();
                HttpURLConnection httpURLConnection = (HttpURLConnection)
urlConnection;
                httpURLConnection.connect();
                if (httpURLConnection.getResponseCode()==200)
                {

System.out.println(urllink+"-"+httpURLConnection.getResponseCode()+"-"+httpURLCo
nnection.getResponseMessage());
                }
                else
```

```java
                {
                        System.out.println(urllink+" -
"+httpURLConnection.getResponseCode()+" -
"+httpURLConnection.getResponseMessage());
                }

                driver.quit();}
}
```

-------To get links on page -------------------

```java
WebDriver driver = LounchBrowser.openChromeBrowser("https://demoqa.com/links");


    List<WebElement> allURLs = driver.findElements(By.tagName("a"));
    System.out.println("Total links on the Wb Page: " + allURLs.size());

    //We will iterate through the list and will check the elements in the list.
    Iterator<WebElement> iterator = allURLs.iterator();

    while (iterator.hasNext()) {
        String url = iterator.next().getText();
         System.out.println(url);
    }
```

How to use drag and drop?
```java
public static void main (String[]args) throws IOException {


System.setProperty("webdriver.chrome.driver","D:\\chromedriver.exe");

                WebDriver driver = new ChromeDriver();

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

driver.navigate().to("https://www.globalsqa.com/demo-site/draganddrop");

                WebElement iframe =
driver.findElement(By.xpath("//iframe[@class='demo-frame lazyloaded']"));

                driver.switchTo().frame(iframe);

                List<WebElement> image
=driver.findElements(By.xpath("//li[@class='ui-widget-content ui-corner-tr
ui-draggable ui-draggable-handle']"));

                WebElement Trash =
driver.findElement(By.xpath("//div[@id='trash']"));

                Actions actions = new Actions (driver);
                int j=image.size();
                System.out.println(j);

                for (int i=0;i<4;i++)
```

```
                {
                actions.dragAndDrop(image.get(i), Trash);
                actions.perform();
        }



How to use scroll down?
1. First create object refrance and upcast driver

        JavascriptExecutor js = ((JavascriptExecutor)driver);
2. Use method of windows .scrollBy you can scroll
        js.executeScript("window.scrollBy(0,1000)");
-----(0,1000) means scroll down vertically by 1000 pixcels




How to handle dropdown?

        WebElement DropDown = driver.findElement(By.xpath("path")

        Select select = new Select(DropDown);

        select.selectByIndex();
        select.selectByValue();
        select.selectByVisibleText();

How do you get all options, selected option text/value?

            WebDriver driver = LounchBrowser.openChromeBrowser("
https://www.tutorialspoint.com/selenium/selenium_automation_practice.htm");
            driver.manage().timeouts().implicitlyWait(5, TimeUnit.SECONDS);

            WebElement
t=driver.findElement(By.xpath("//*[@name='continents']"));

            Select select = new Select(t);

     2.How do get default value of dropdown
            WebElement first = select.getFirstSelectedOption();
            String selectedoption = first.getText();

    3. How to get all options and select desired option
            List<WebElement> options =select.getOptions();
            Iterator<WebElement>itr = options.iterator();
            while (itr.hasNext())
            {
                    WebElement option = itr.next();
                    String selectedOption =option.getText();
                    if (option.getText().equals("Australia"))
                    {
                            select.selectByVisibleText(selectedOption);
                            System.out.println(selectedOption);
                    }
            }
```

```
                }
        }

}
```

How to handle iframe?
```
public static void main (String[]args) {

                WebDriver driver =
BrowserOpen.open("https://vctcpune.com/selenium/practice.html");

                WebElement IFrame =
driver.findElement(By.xpath("//iframe[@id='courses-iframe']"));
                driver.switchTo().frame(IFrame);

                WebElement Courses =
driver.findElement(By.xpath("(//a[text()='Courses'])[1]"));
                Courses.click();
```