CYDEO

Synchronization

Agenda

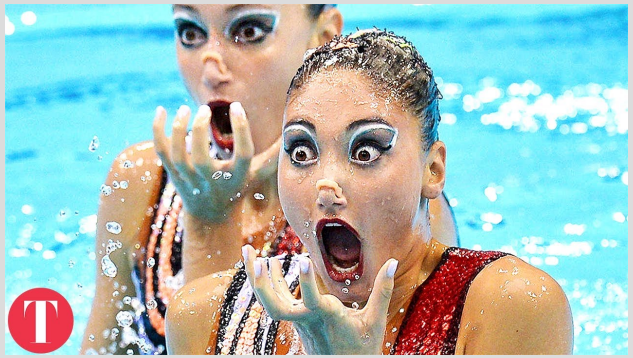
- Thread.sleep
- ImplicitlyWait
- ExplicitWait

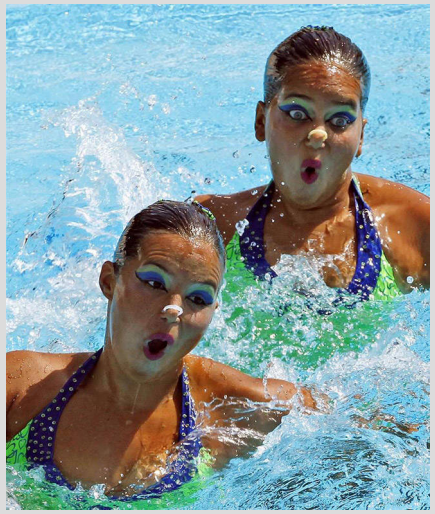
After today's session you should be able to:

- Handle the execution synchronization
- Use different ways of waiting for certain actions
- Minimize the test failure due to timeouts or exceptions

What does synchronization mean?

- Moving, working, operating at the same time.
- One of the most obvious examples:
  - from the movies, sound and action has to match precisely, (OR SYNCHRONIZE), otherwise the outcome will not be as expected.





What needs to be synchronized in our case?

- Our code and our browser has to be synchronized.
- Otherwise it will break.
- The outcome will not be as expected.
- We have been using our code synchronized, but we will go in more depth what it means today.

Why Synchronize?

What happens if our code is not synchronized?

- Code runs faster than browser can handle
- Element appears on the HTML after certain time
- Element is in the HTML but appears on the page after certain time
- We get exceptions such as;
- NoSuchElementException
- ElementNotInteractableException

What can we do?

Option 1:

- **Thread.sleep**
- This adds hard coded wait inside of our program.
- It does not depend on any kind of condition
- It will wait for the given time period no matter what
- This is not considered a good practice
- Try not to use this unless you have to
- Too much use will make the test long and heavy

What can we do?

```
driver.manage().timeouts().implicitlyWait()
```

Option 2:

- **Implicit Wait**
- **Not used for a specific condition, applies to many lines until that driver instance is killed or changed.**
- Makes driver continue looking for the WebElement for the given duration
- Stops polling as soon as the element is found
- If element is not found it will throw exception after the given time → no such element exception
- It will apply to every single line where **findElement()** or **findElements()** is used

What can we do?

Option 3:

- **Explicit Wait**
- **Can be used to wait for a certain condition**
- Wait until element is visible
- Wait until element is available
- Wait until element is clickable
- Wait until element is not available etc…

Explicit Wait: Syntax

```
WebDriverWait wait = new WebDriverWait(Driver.getDriver(), Duration.ofSeconds(10));
wait.until(ExpectedConditions.elementToBeClickable(By.xpath( xpathExpression: "locator")));
```

- It will wait 10 seconds using the locator provided then if condition does not happen throws exception → TimeOutException
- Does not depend on the value of implicit wait
- Only applies once when that line called
- Will continue waiting if the element is loaded but not clickable, or many similar scenarios