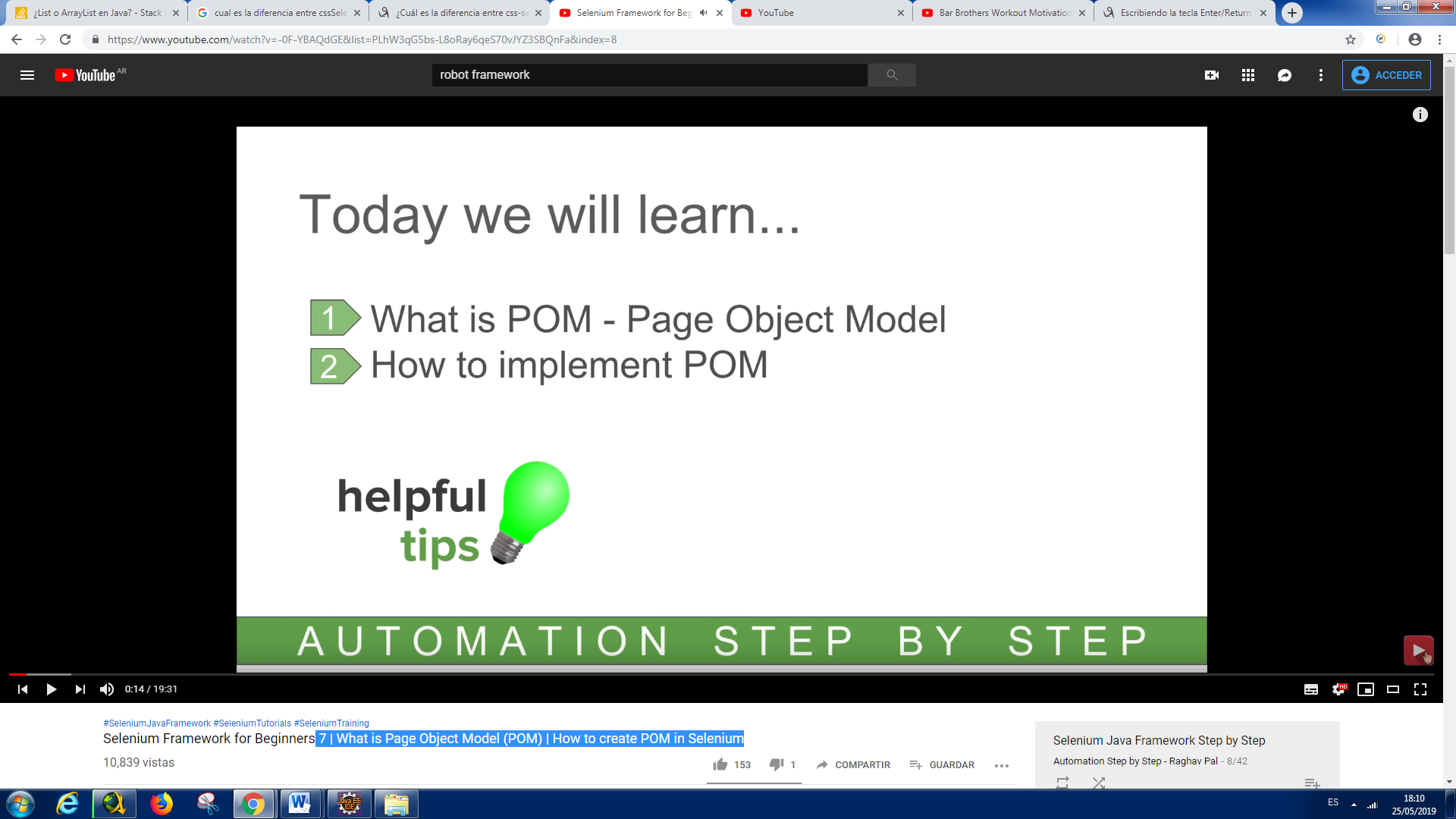
<https://www.youtube.com/watch?v=Jvd9txq-Qw0>

<https://www.youtube.com/watch?v=Jvd9txq-Qw0>

<https://www.youtube.com/watch?v=I0aeDQwGywc>

https://www.youtube.com/watch?v=aR9nh01zkAQ



1. What is POM - Page Object Model

2. How to implement POM

What is POM

===========

Design Principle

Create separate Object Repository to store object locators

For each web page there should be a separate page class

The page class will have all object locators and preferably action methods for the corresponding page

Step 1 :

Create a class for each web page

Step 2 :

Add object locators

Step 3 :

Create class for test

Step 4 :

Refer objects from page classes

Step 5 :

Run and validate

Benefits of POM

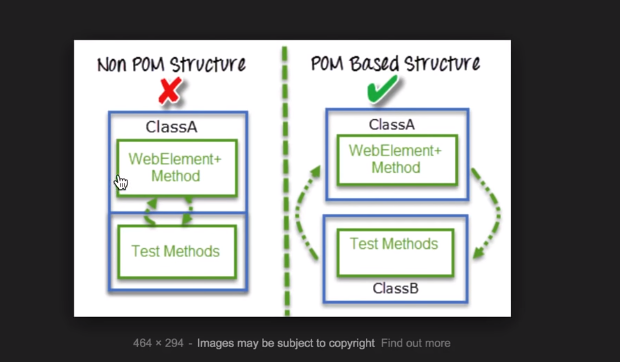
=============

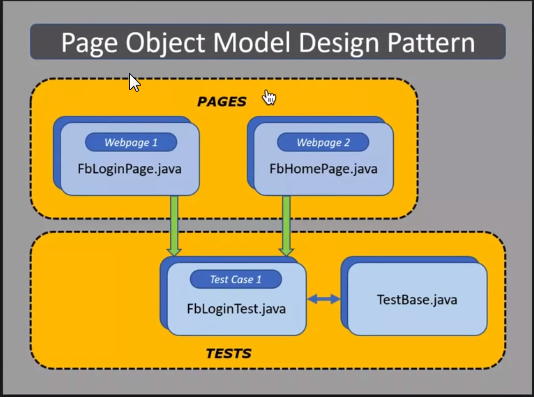
Test Objects and Functions are separated for easy maintenance and keeping the code clean

Objects are kept separated from test scripts and can be used multiple times in multiple tests

Every unique object locator is created only once

Easy Maintenance and Less Rework





//Español

-Se debe crear un repositorio de objetos (elementos de la pagina: botón , textos , links) independientemente de las clases

-Clada pagina estará representada en una clase con sus elementos (objetos) como atributos y los métodos

Crear un repositorio de objetos separado para almacenar los localizadores de objetos

Para cada página web debe haber una clase de página separada.

La clase de página tendrá todos los localizadores de objetos y, preferiblemente, métodos de acción para la página correspondiente.

Paso 1 :

Crea una clase para cada página web.

Paso 2 :

Añadir localizadores de objetos

Paso 3 :

Crear clase para prueba

Etapa 4 :

Referir objetos de clases de página.

Paso 5:

Ejecutar y validar

Beneficios de POM

=============

Los objetos y funciones de prueba están separados para facilitar el mantenimiento y mantener el código limpio

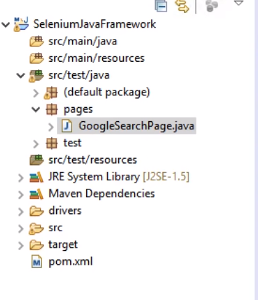
Los objetos se mantienen separados de los scripts de prueba y se pueden usar varias veces en múltiples pruebas

Cada localizador de objetos único se crea una sola vez.

Fácil mantenimiento y menos retrabajo

#SeleniumJavaFramework #SeleniumTutoriales #SeleniumTraining #SeleniumFramework #SeleniumJavaTutorial #SeleniumBeginnerTraining

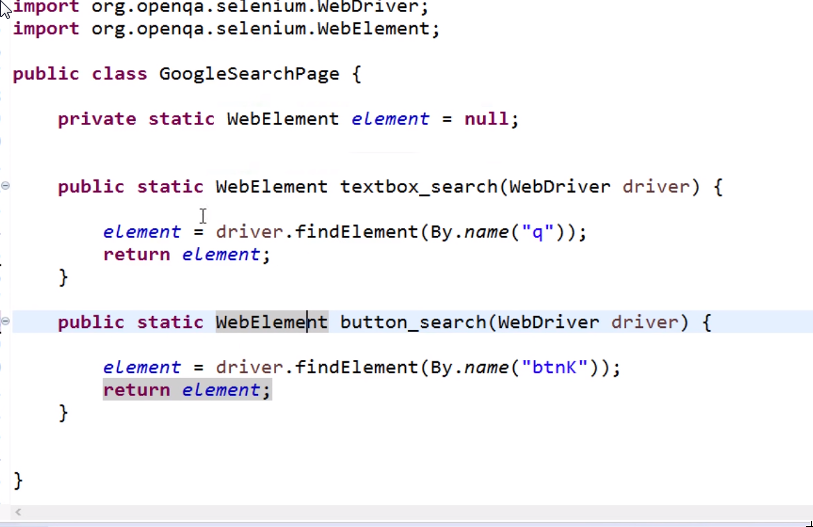
//Creo una clase por pagina



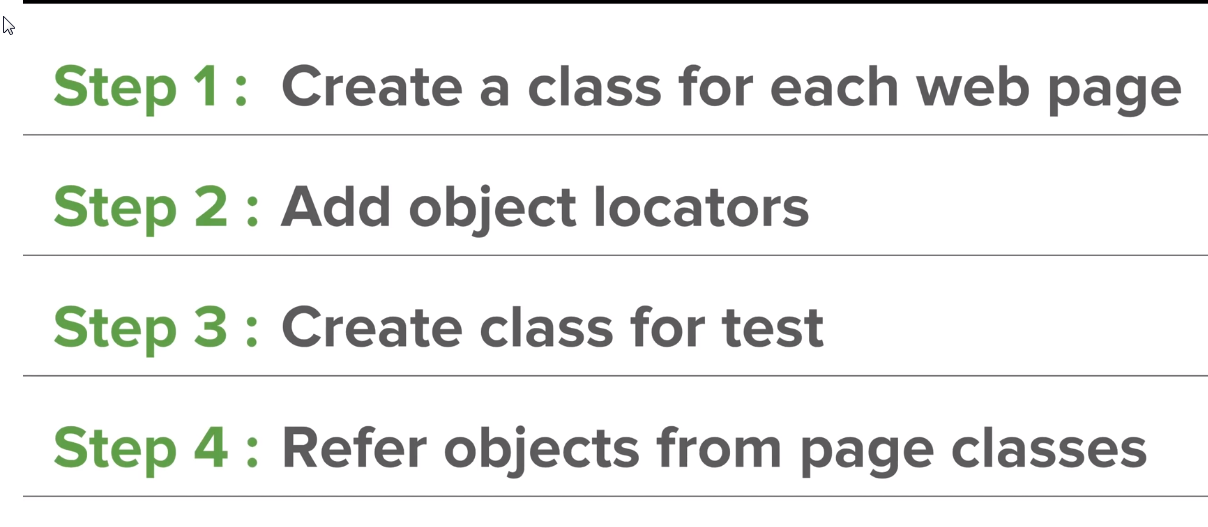
/Objeto

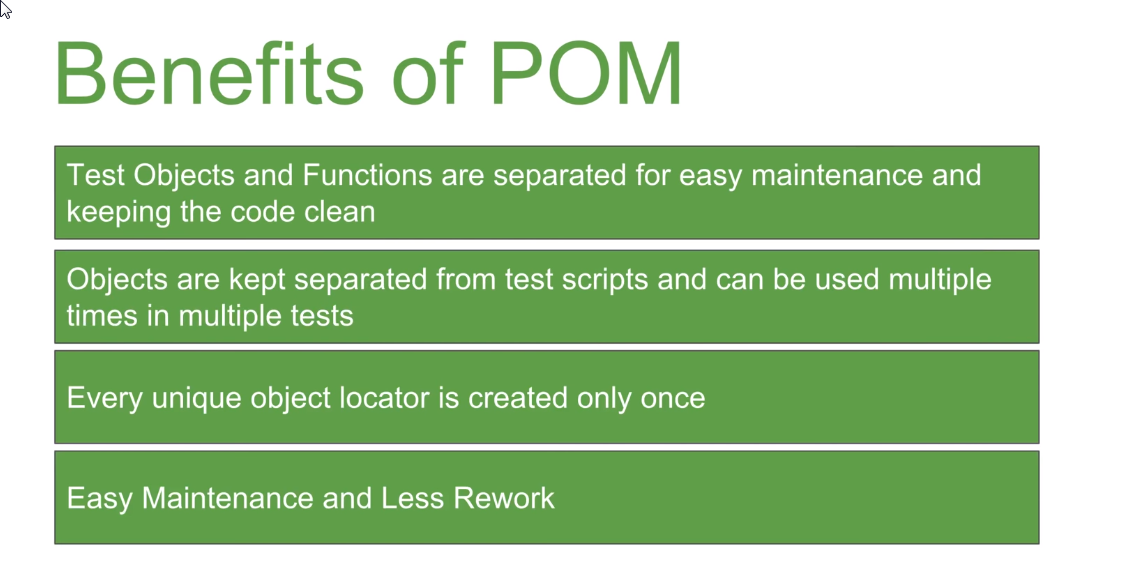
Declararlo : Clase objeto=null;

Iniciarlo: objeto=new Clase();







++

<https://www.tutorialselenium.com/2019/02/05/page-object-model-selenium-webdriver/>