Modelos de Calidad de Software

Tema Nº10:Introducción al Selenium WebDriver – Parte 2

Indicador de logro Nº10:Interpreta los resultados de la automatización de pruebas realizadas, generando los scripts según funcionabilidades de la web con data en duro y dinámica.

**TEMA 01 Teoría de los**

Imagen que contiene Icono

Descripción generada automáticamente

**TEMA Nº10:**

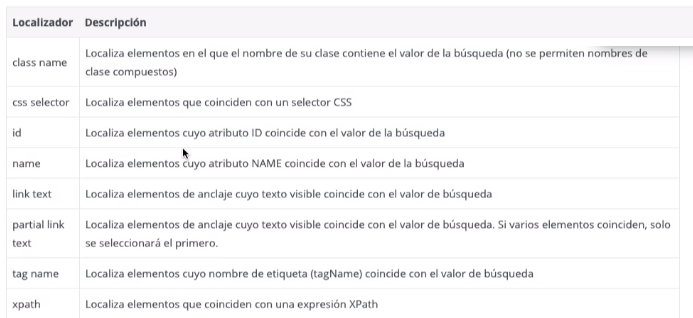
Introducción al Selenium WebDriver – Parte 2

**Subtema 10.1:**

Detectar elementos en la web (Localizadores)

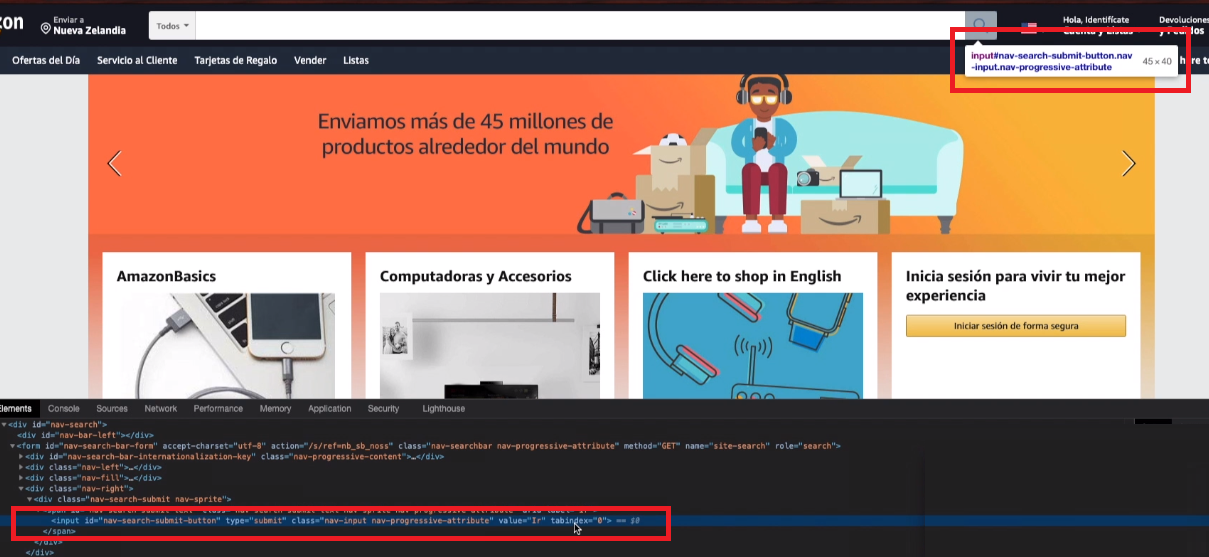
Locators son la dirección o el domicilio para poder llegar a los Web Elements con los cuales necesitamos interactuar (botón, dropdown, etc.), al mismo tiempo estos We Elements están dentro del DOM (Html donde están todos los elementos)

Siendo un conjunto de parámetros con los cuales podemos identifcar un elemento en una página, los tipos de locators que tenemos:



Hay varios tipos de locators pero se trata de trabajar con los Id’s, en teoría el ID es un locator único, pero en la practica el ID es dinámico.

Ver los Locators con Clic derecho e inspeccionar.

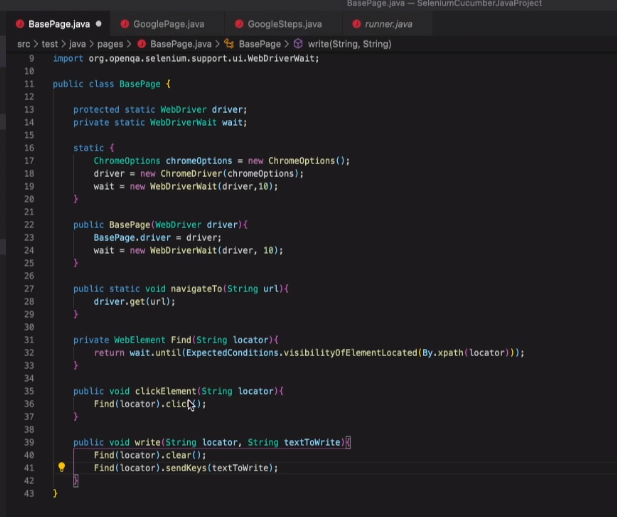


**Subtema 10.2:**

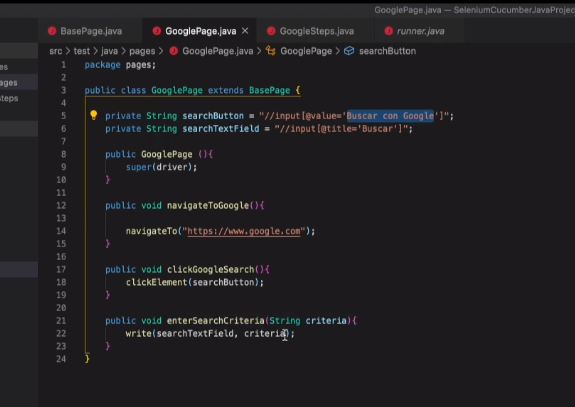
Creación de scripts de pruebas para certificar la web

Vamos a realizar un script con dos localizadores para la página de Google, vamos a escribir en una caja de texto (Código duro) y luego hacer click.

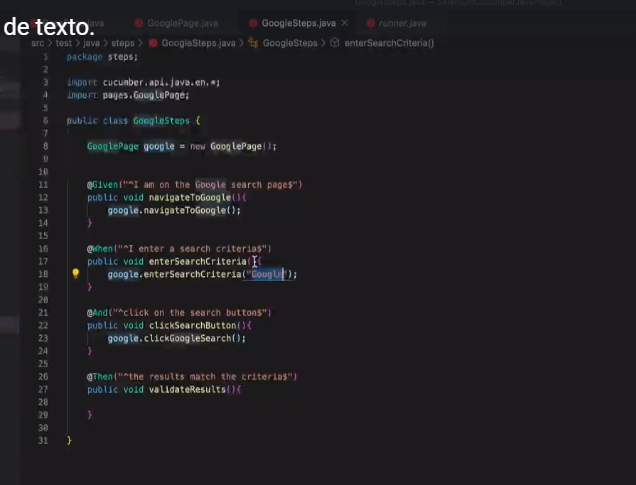
BasePage



GooglePage



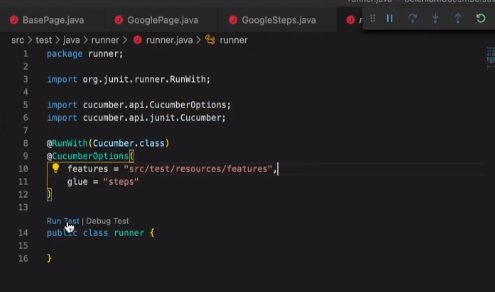
GoogleSteps



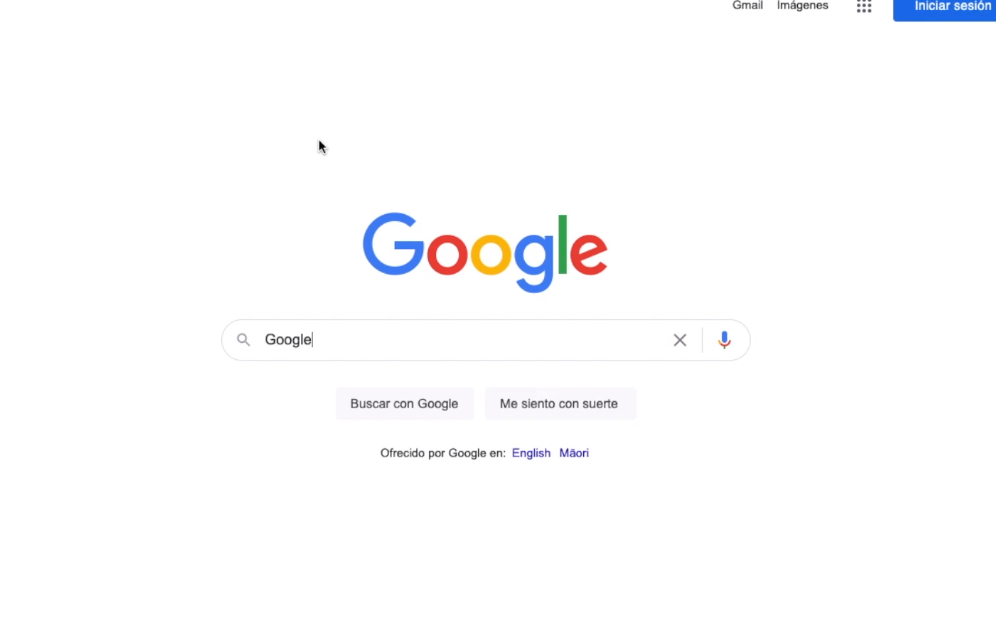
**Subtema 10.3:**

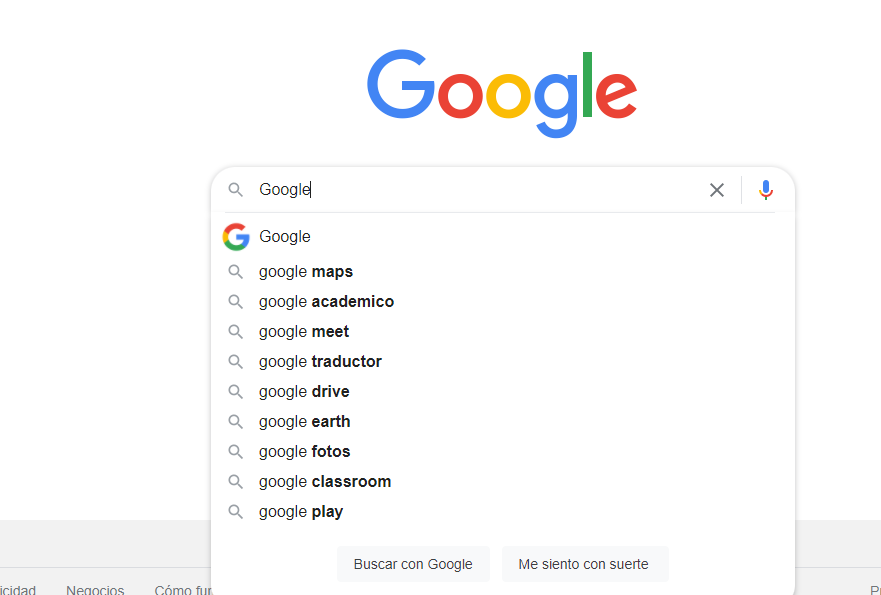
Ejecución del script automatizado

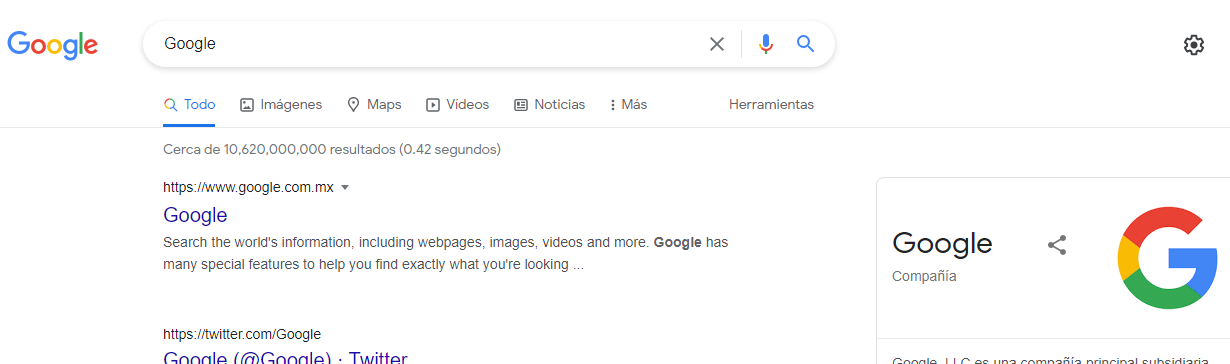
Ejecutamos el Runner



A continuación Carga la Página de Google, escribe texto y Click



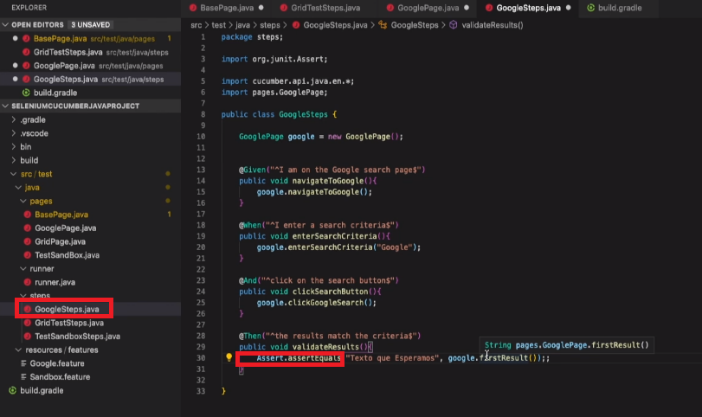




**Subtema 10.4:**

Integrar JUNIT con Selenium WebDriver

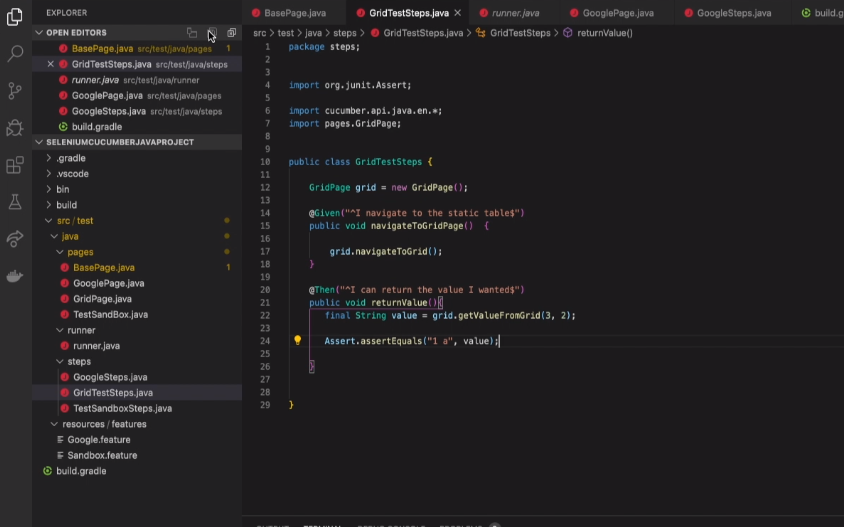
Está integración permite realizar validaciones para los fujos básicos y alternativos definidos, es importante mencionar que como buena práctica la clase que modela la página no debe contener los Test, los Test deben ir en la clase de steps esto se hace potente cuando se trabaja con un equipo grande de desarrolladores y probadores.



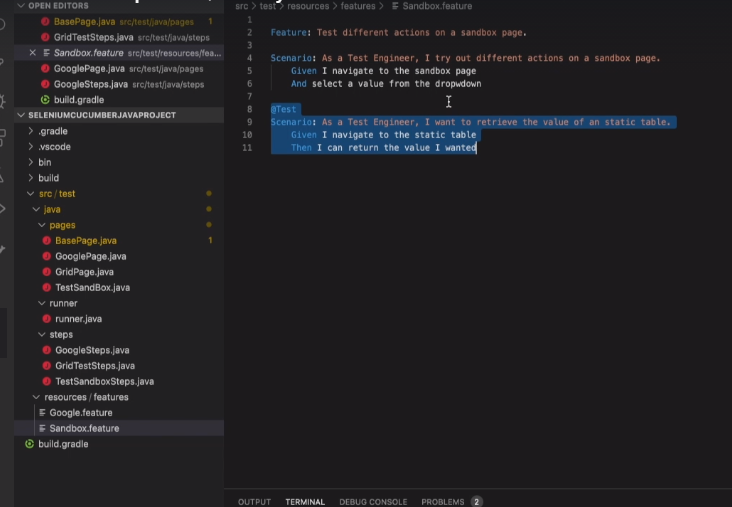
**Subtema 10.5:**

Revisión de los resultados

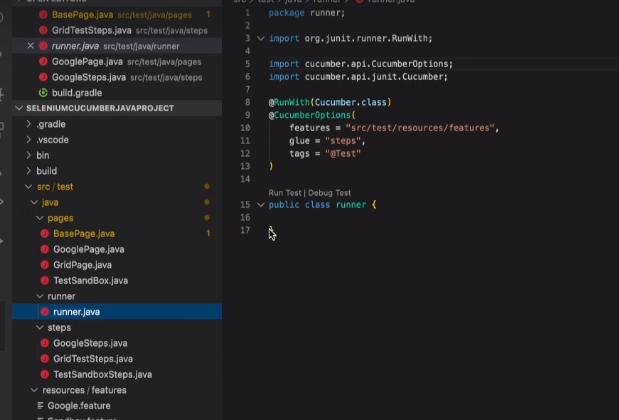
Vamos a validar el resultado respecto a una automatización con una grilla estática, recordatorio consideremos los Test en los Step Definition, se va buscar que falle el flujo de prueba.



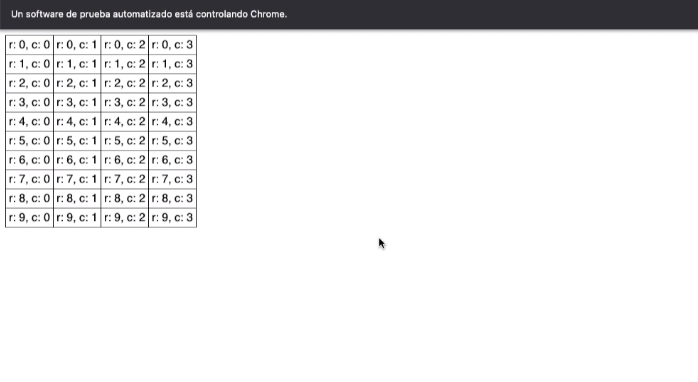
Va ejecutar el Feature



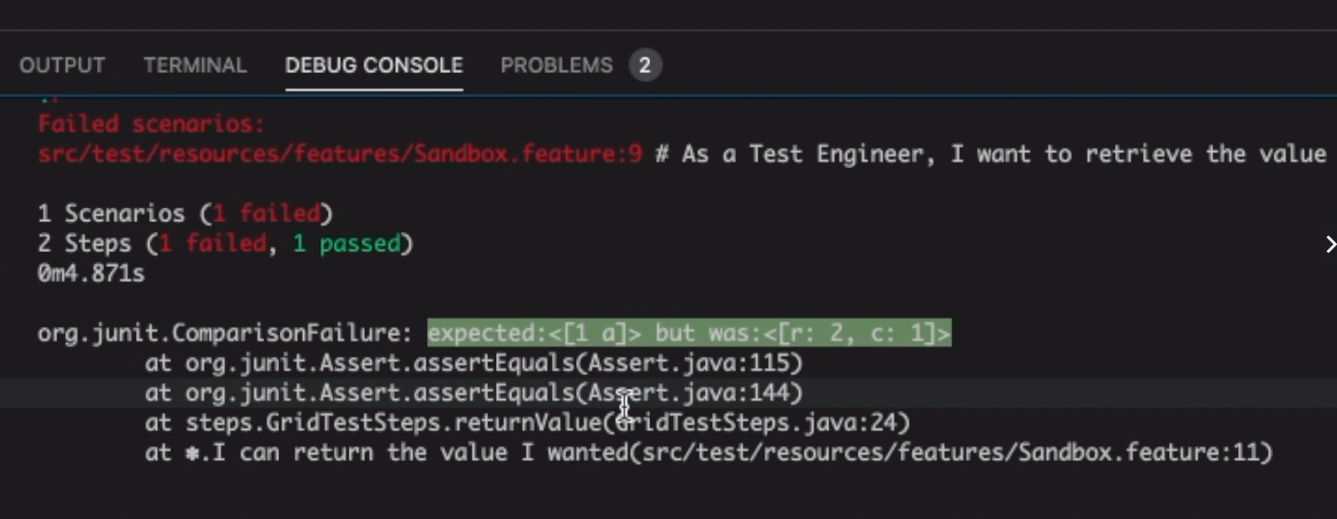
Vamos al Run a ejecutar



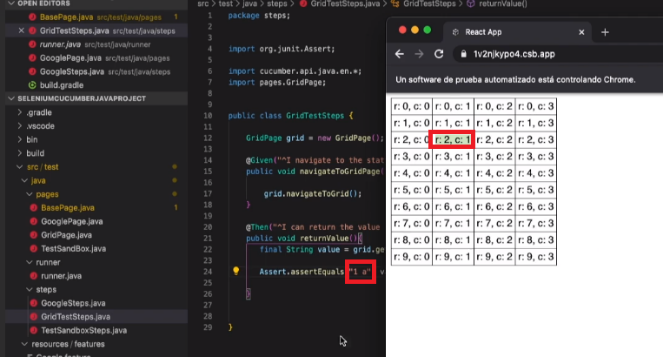
Brinda



Revisamos el Debug



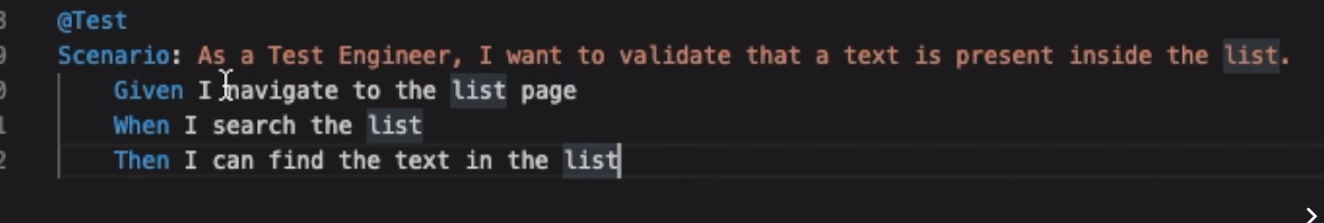
Validación Final



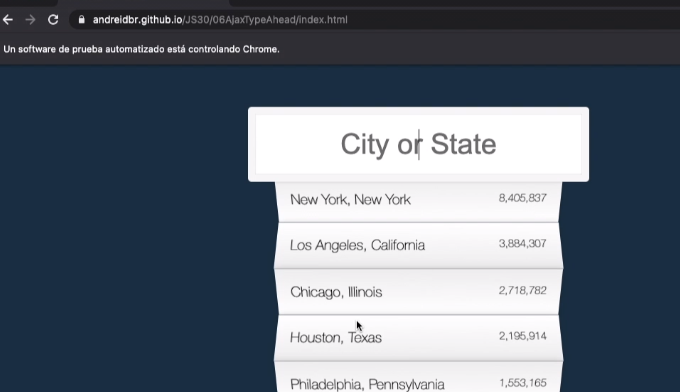
**Subtema 10.6:**

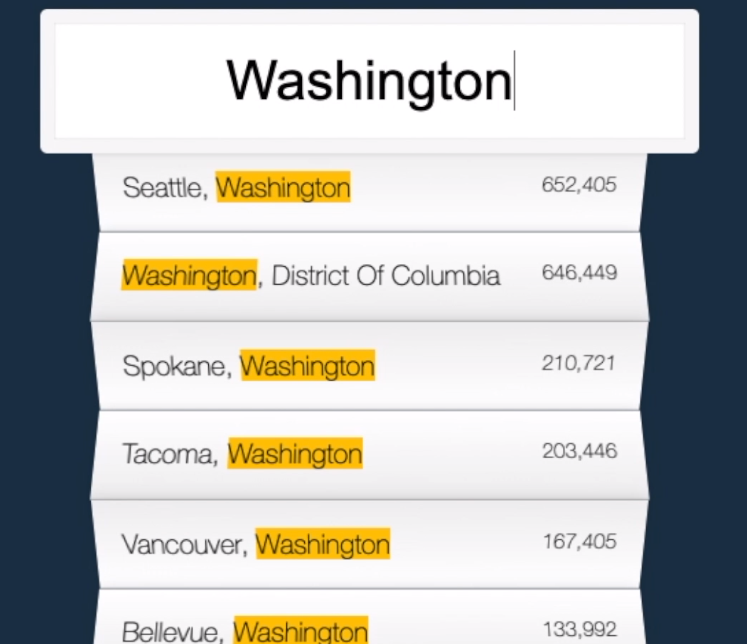
Características avanzadas de script de pruebas

Vamos a validar una lista (colección de objetos de un determinado tipo) con la creación de un nuevo escenario con 3 pasos

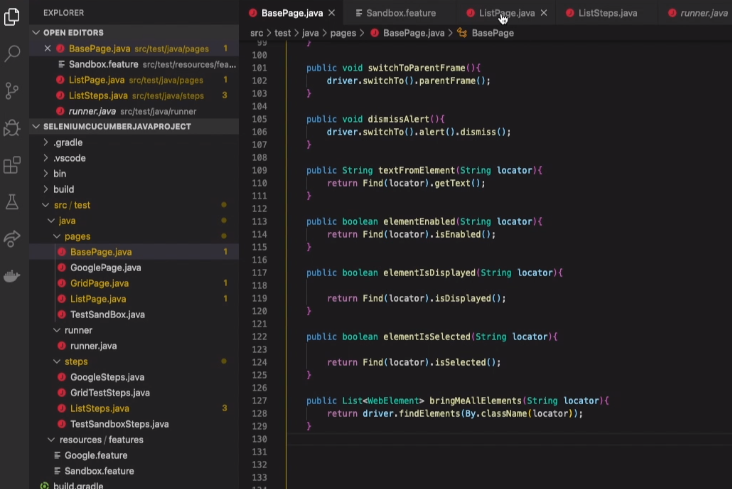


Página de lista a probar

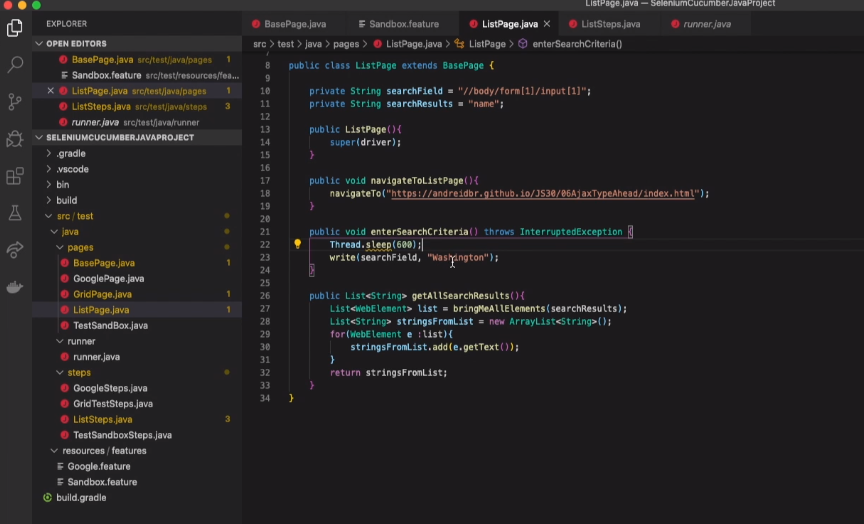




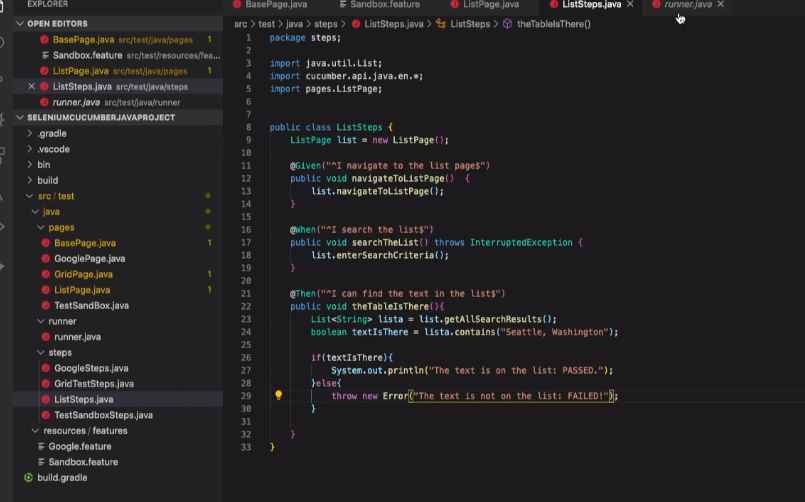
Creamos una lista para devolución en el BasePage.



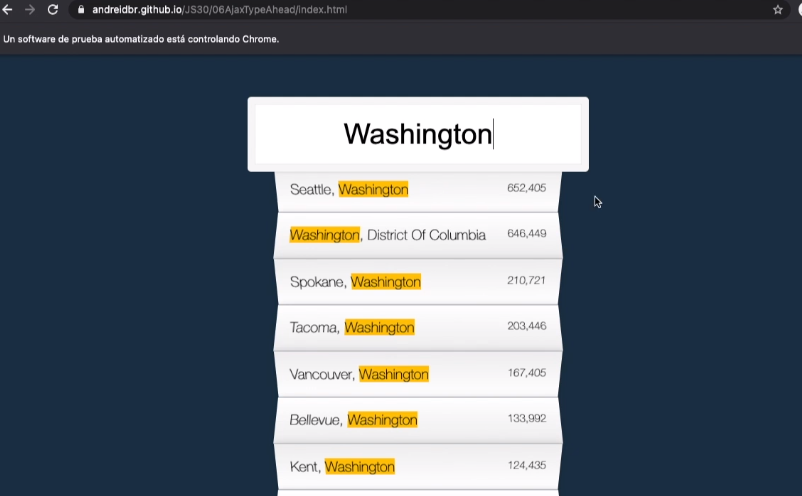
Creación de clase List Page con 3 escenarios (Donde el texto de cada Web Element va a un array)



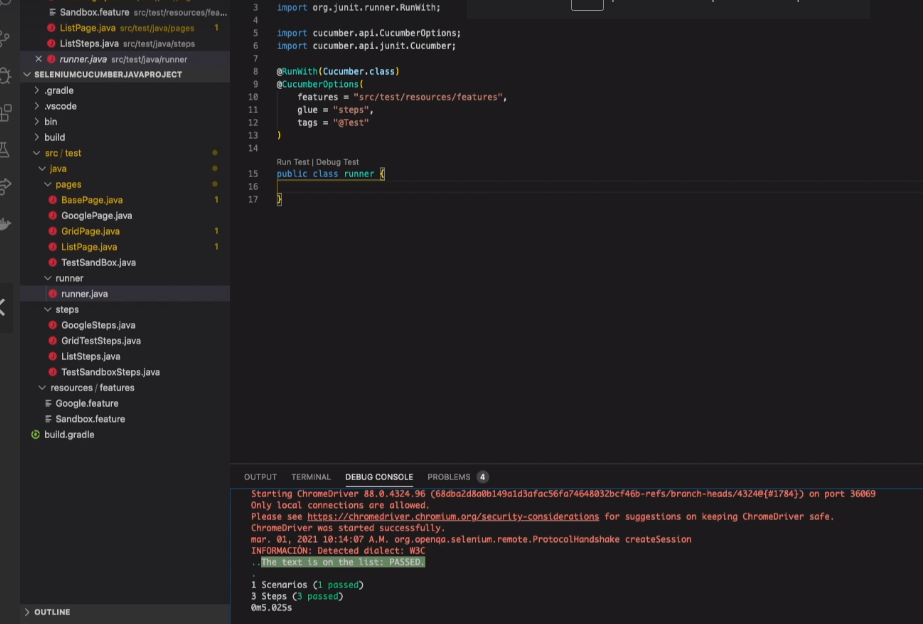
Ahora vemos la validación con mensaje True o False



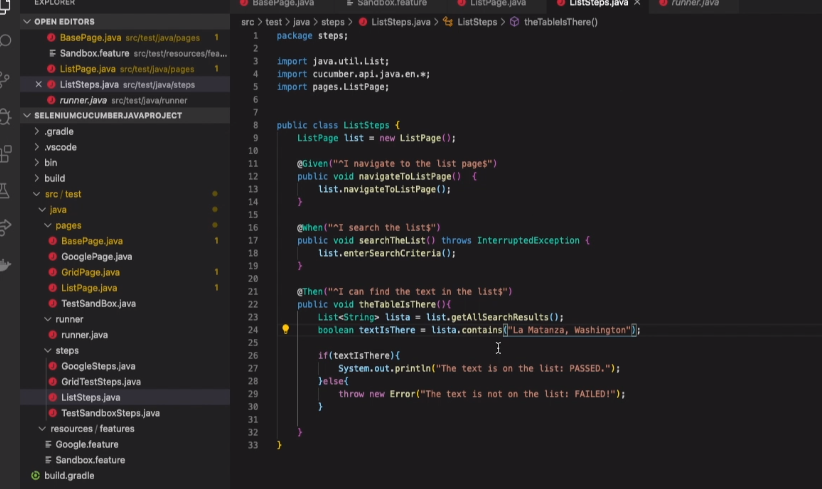
Ejecutamos



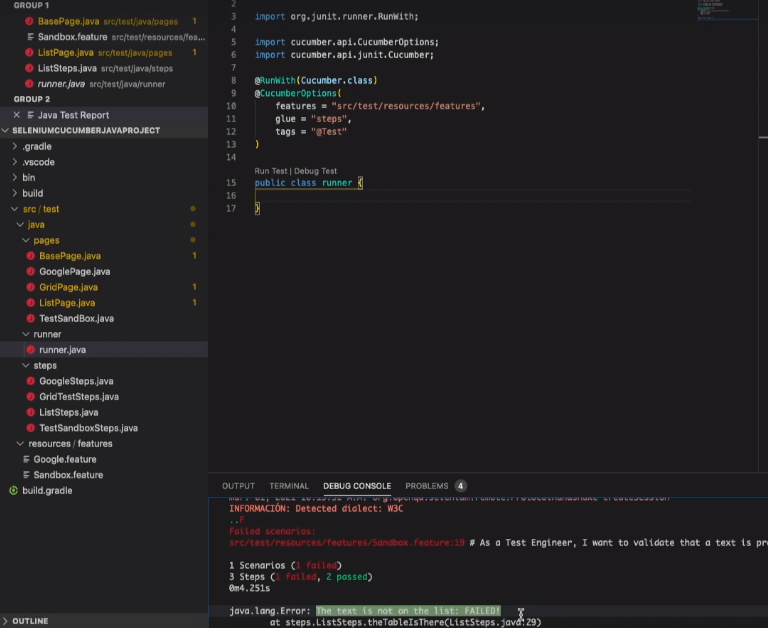
Vemos que pasó en la búsqueda de una lista



Ahora forzamos un flujo alternativo que falle (nuevo Testo – La Matranza. Washington)



Ejecutamos y Vemos que falló



Actividad:

Revisa y analiza el enlace sugerido\* en la plataforma virtual, luego realiza la actividad

Propuesta:

* ¿Qué son las pruebas funcionales?
* Empleando los conocimientos adquiridos en clase, automatizar la web del caso
* práctico.

\*Cada docente deberá proporcionar el enlace que considere apropiado para la ejecución de la actividad.