



Curso de

Introducción a Selenium con Python

Héctor Vega

Héctor Vega



@Terragnigm
Ark

- Lic. en Administración
- Apasionado por la tecnología
- Desde 2015 en tech
- Online Community Relations en Student Success en Platzi

▶ ¿Qué aprenderás?



Entender
¿Qué es Selenium?



Comunicar
Selenium con el
navegador

¿Qué aprenderás?



Automatizar
Pruebas unitarias y
funcionales



Generar
Reportes de pruebas

Requisitos



**Curso de Introducción
al Pensamiento
Computacional con
Python**



**Curso de Programación
Orientada a Objetos:
POO**

Requisitos



**Curso de Desarrollo
Web Online**



**Curso de Fundamentos
de Pruebas de Software**

¡Comencemos!





Historia de **Selenium**



¿Qué es Selenium?

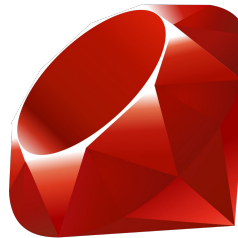
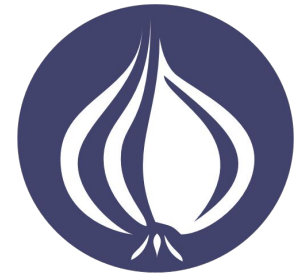
- “Suite de herramientas para automatización de navegadores”



Navegadores



Lenguajes



Selenium NO es una herramienta de...



Testing



Web scraping



Todo comienza en 2004

- **Jason Huggins** buscaba automatizar pruebas manuales, creando así una aplicación llamada “JavaScriptTestRunner”. Después llamada “Selenium Core”

- **Paul Hammant** vio el demo y buscó una solución a la “Same Origin Policy” y se creó “Selenium RC”.



Más fácil, mejor

- **Shinya Katasani** en Japón envolvió el código de Selenium convirtiéndolo en un plugin para Firefox.
- **Selenium IDE** es capaz de grabar, repetir, importar y exportar automatizaciones.



Un cliente para todos

- **Simon Stewart** trabajó en varias herramientas llamadas “WebDriver”. Reemplazando JS por un cliente para cada navegador y una API de alto nivel.
- **Selenium RC** se fusionó a este proyecto, dando lugar a Selenium WebDriver.

Origen del nombre



Selenio para el envenenamiento
por mercurio



Características, pros y contras





Selenium IDE

PROS

- Excelente para iniciar
- No requiere saber programar
- Exporta scripts para Selenium RC y Selenium WebDriver
- Genera reportes



Selenium IDE

CONTRAS

- Disponible solo para Firefox y Chrome
- No soporta DDT



Selenium RC

PROS

- Soporte para:
 - Varias plataformas, navegadores y lenguajes
 - Operaciones lógicas y condicionales
 - DDT
- Posee una API madura



Selenium RC

CONTRAS

- Complejo de instalar
- Necesita de un servidor corriendo
- Comandos redundantes en su API
- Navegación no tan realista



Selenium WebDriver

PROS

- Soporte para múltiples lenguajes
- Fácil de instalar
- Comunicación directa con el navegador
- Interacción más realista



Selenium WebDriver

CONTRAS

- No soporta nuevos navegadores tan rápido
- No genera reportes o resultados de pruebas
- Requiere de saber programar



Selenium Grid

Características

- Se utiliza junto a Selenium RC
- Permite correr pruebas en paralelo
- Conveniente para ahorrar tiempo



Otras **herramientas** de **testing** y **automatización**



Puppeteer

PROS

- Soporte por parte de Google
- Datos del Performance Analysis de Chrome
- Mayor control de Chrome
- No requiere de drivers externos

CONTRAS

- Funciona solo en Chrome y con JavaScript
- Comunidad pequeña



Cypress.io

PROS

- Comunidad emergente
- Buena documentación con ejemplos
- Bastante ágil en pruebas E2E
- Orientado a desarrolladores
- Excelente manejo de asincronismo

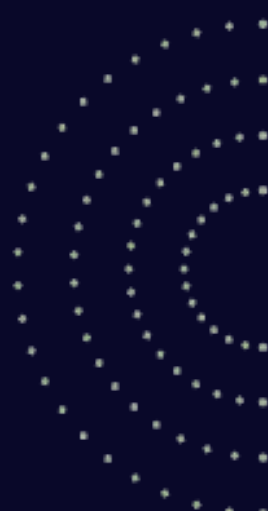
CONTRAS

- Funciona solo en Chrome y con JavaScript
- Pruebas en paralelo solo en versión pago



**¿Cuál es la mejor
opción?**





Configurar entorno de **trabajo**



Instalaciones

- Python 3.6 en adelante
- Selenium
- PyUnitReport



¡Hola, **mundo!**



Unittest (PyTest)

- **Test Fixture:** preparaciones para antes y después de la prueba.
- **Test Case:** unidad de código a probar.
- **Test Suite:** colección de Test Cases.
- **Test Runner:** orquestador de la ejecución.
- **Test Report:** resumen de resultados.



Múltiples pruebas en una misma instancia

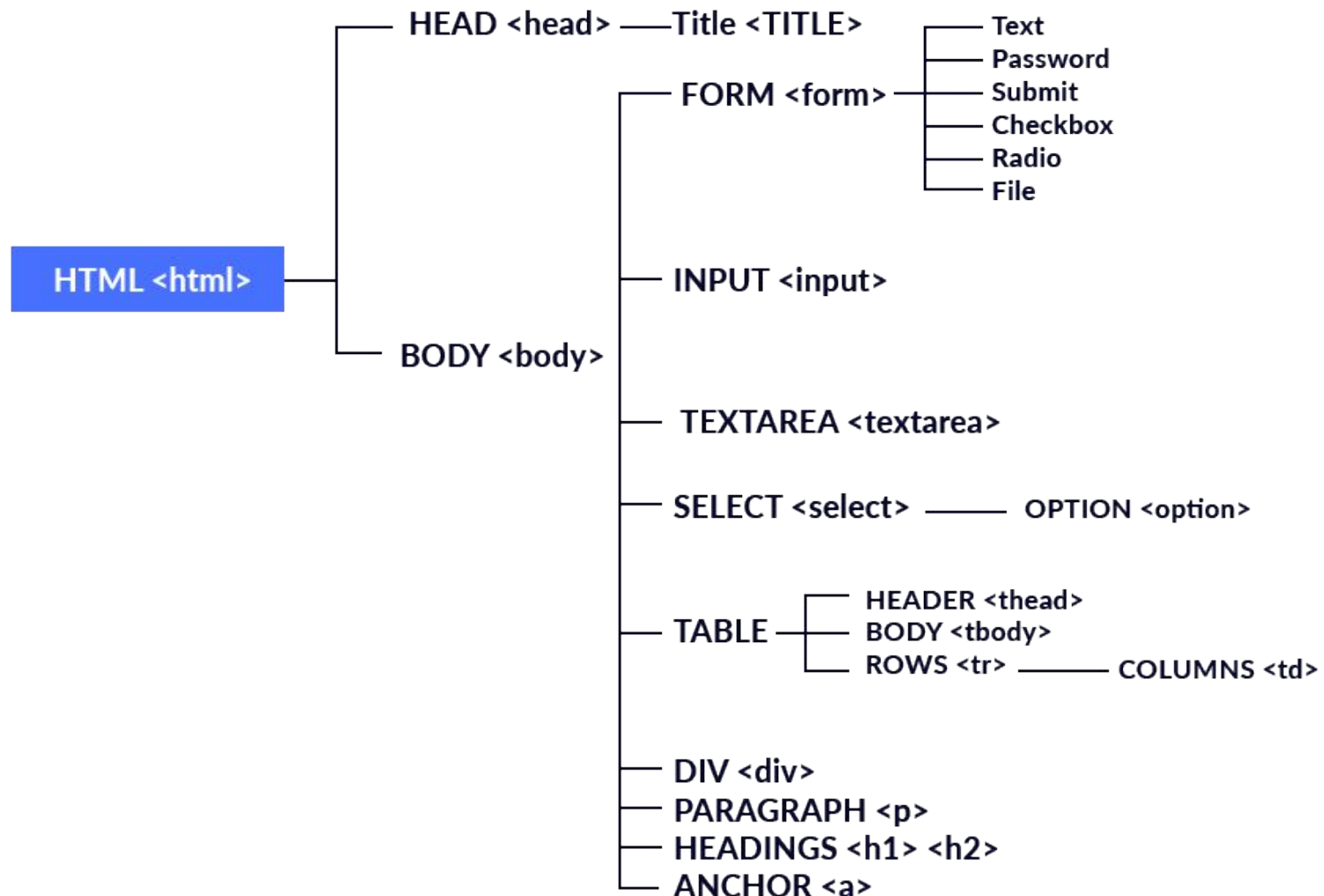




Encontrar
elementos con
find_element



Estructura de un sitio





Selectores

- ID
- Nombre del atributo
- Nombre de la clase
- Nombre de la etiqueta
- XPath
- Selector de CSS
- Texto del link
- Texto parcial del link



Preparar **assertions** y **test suites**



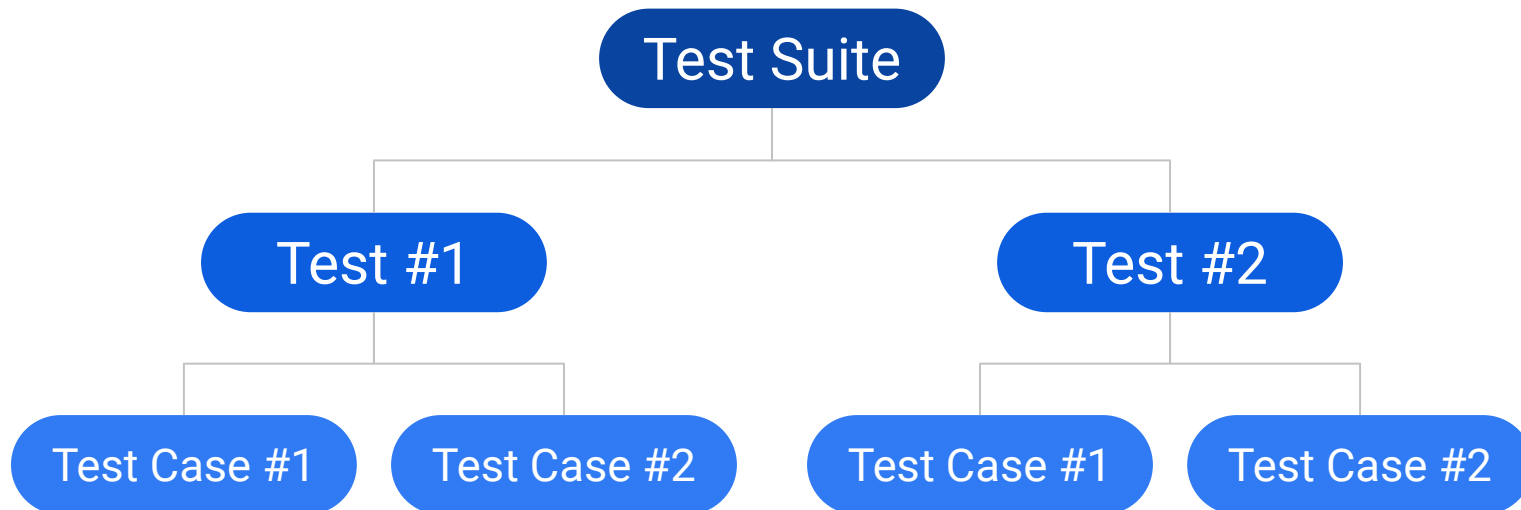
Assertions

- Métodos que permiten validar un valor esperado en la ejecución del test. Si el resultado es verdadero el test continúa, en caso contrario “falla” y termina.
- **Ejemplo:** `assertEqual(price.text, “300”)`



Test suites

Colección de test unificados en una sola prueba, permitiendo tener resultados grupales e individuales.





Entender las clases **WebDriver** y **WebElement**


Manejar
**form, textbox,
checkbox y radio
button**



Manejar **dropdown y listas**



Manejar **alert** y **pop-up**



Automatizar navegación



Demora **implícita** y **explícita**

Manejar errores

try y **except**



Demoras

- **Implícita:** busca uno o varios elementos en el DOM si no se encuentran disponibles por la cantidad de tiempo asignado.
- **Explícita:** utiliza condiciones de espera determinadas y continúa hasta que se cumplan.



RETO 1: Agregar y eliminar elementos



Elementos **dinámicos**



Controles dinámicos



Typos



Ordenar **tablas**



Data Driven **Testing (DDT)**



¿TDD o DDT?

- **Test Driven Development:** desarrollar código en base a pruebas para que pueda cumplirlas.
- **Data Driven Testing:** desarrollar pruebas en base a código ya existente para validar en qué escenarios pasan o fallan.



Ejemplo de un formulario

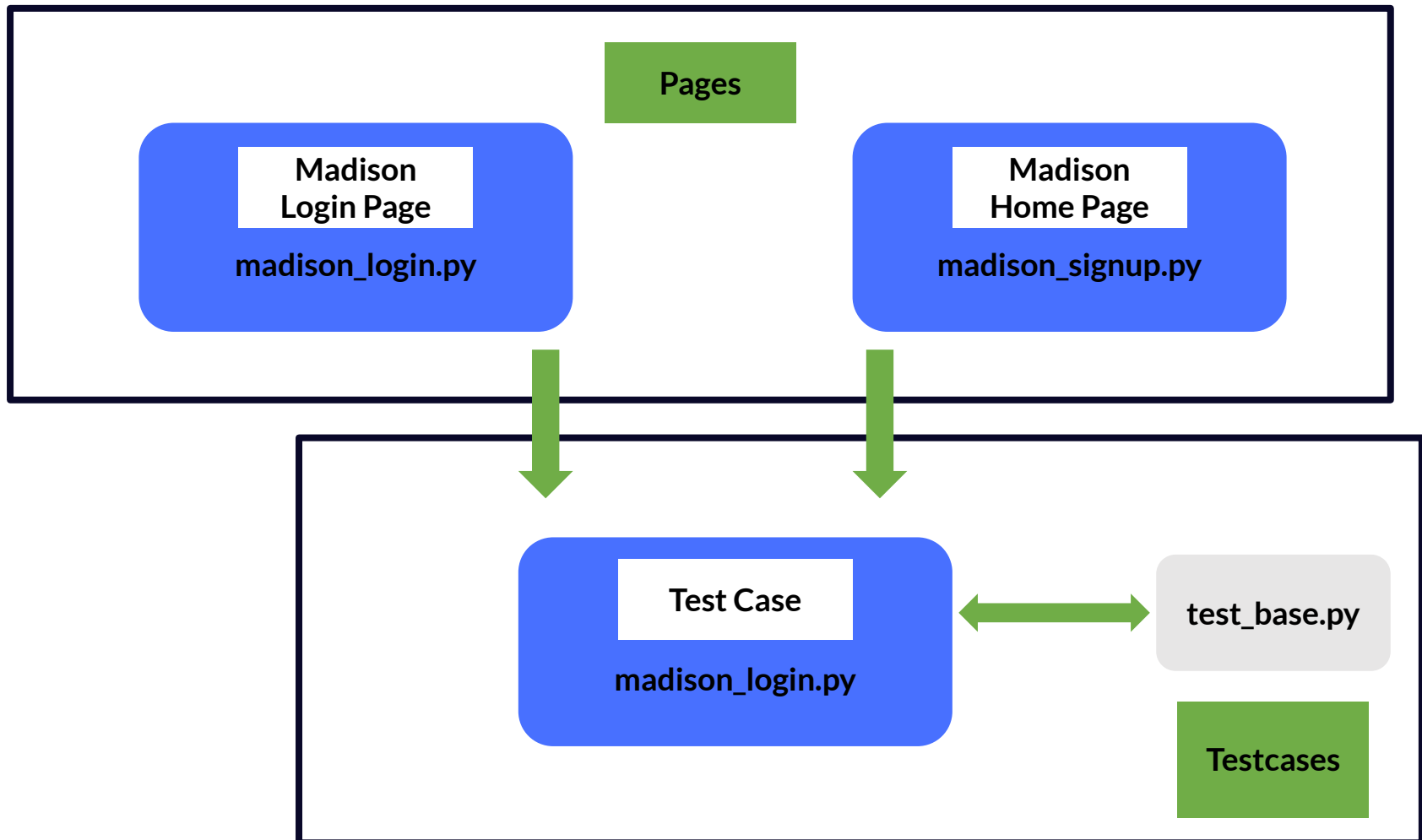
DESCRIPCIÓN	DATOS DE PRUEBA	SALIDA ESPERADA
Colocar valores numéricos en un campo de “monto a pagar”.	500000	El usuario debería poder pagar y recibir mensaje de confirmación.
Colocar valores numéricos y de caracteres en un campo de “monto a pagar”.	\$500.000,00	El usuario debe recibir un mensaje de error indicando que coloque solo valores numéricos



Page Object **Model (POM)**



Page Object Model





Beneficios

- Crea un alto nivel de abstracción para minimizar cambios en las pruebas si los desarrolladores modifican el sitio.
- Crea código reutilizable que se puede utilizar en múltiples pruebas.
- Las pruebas son más legibles, flexibles y vigentes.



Realizar una
prueba **técnica**



Consideraciones

- Practicar en sitios complejos
- Preguntar acerca acerca de las expectativas
- Define paso a paso el flujo
- Piensa como usuario final
- Programa como desarrollador



Flujo en Mercado Libre

1. Ingresar a [mercadolibre.com](https://www.mercadolibre.com).
2. Seleccionar 'Colombia' como país.
3. Buscar el término "playstation 4".
4. Filtrar por condición "Nuevos".
5. Filtrar por ubicación "Bogotá".
6. Ordenar de mayor a menor precio.
7. Obtener el nombre y precio de los primeros 5 artículos.

Conclusiones



Lo que puedes lograr

- Automatizar el navegador para testing, scraping, tareas repetitivas... ¡Lo que se te ocurra!
- Automatizar casos de pruebas, suites de pruebas y generar sus reportes.
- Aplicar DDT y POM a tus scripts.



Recuerda

- Probar sitios con autorización o diseñados para esto.
- Haz un buen uso de las automatizaciones.
- **Aún puedes hacer más:** tienes Selenium Grid para pruebas en paralelo y Appium para mobile testing.

Felicidades
¡Happy automation!

