

# EJERCICIO 1:

## INSTRUCCIONES:

1. Se puede utilizar cualquier framework, sin embargo, el deseable es: Serenity BDD.
2. Al finalizar el ejercicio se debe subir en un repositorio github público. El ejercicio debe contener archivos, scripts, reportes y cualquier otro insumo que sustente su implementación para que pueda ser reproducido por la persona a revisar. Adicional debe contener:
  - Un archivo readme.txt con las instrucciones paso a paso de ejecución.
  - Un archivo conclusiones.txt con los hallazgos y conclusiones del ejercicio

## EJERCICIO:

Realizar una prueba funcional automatizada (Prueba E2E) de un flujo de compra en la página <https://www.demoblaze.com/> que incluya:

- Agregar dos productos al carrito
- Visualizar el carrito
- Completar el formulario de compra
- Finalizar la compra

# EJERCICIO 2:

## INSTRUCCIONES:

1. Se puede utilizar cualquier herramienta, sin embargo las deseable es Karate
2. Al finalizar el ejercicio se debe subir en un repositorio github público. El ejercicio debe contener archivos, scripts, reportes y cualquier otro insumo que sustente su implementación para que pueda ser reproducido por la persona a revisar. Adicional debe contener:

Un archivo readme.txt con las instrucciones paso a paso de ejecución.

Un archivo conclusiones.txt con los hallazgos y conclusiones del ejercicio

## EJERCICIO:

La página <https://petstore.swagger.io/> proporciona la documentación sobre apis de una "PetStore". Utilizando un software para pruebas de servicios REST realizar las siguientes pruebas, identificando las entradas, capturando las salidas, test, variables, etc, en cada uno de los siguientes casos:

- Añadir una mascota a la tienda
- Consultar la mascota ingresada previamente (Búsqueda por ID)
- Actualizar el nombre de la mascota y el estatus de la mascota a "sold"

- Consultar la mascota modificada por estatus (Búsqueda por estatus)