

## **Taller 3 – Parte 1**

### **Pruebas y Errores**

#### **Objetivo general**

El objetivo de este taller es practicar los conceptos básicos de Python revisados en la clase anterior. Cada ejercicio le permitirá poner en práctica los conceptos vistos en las diapositivas anteriores. Recuerde poner en práctica las buenas prácticas de programación discutidas en el curso. Comente y ponga nombres significativos a las funciones y variables para que su código sea claro.

#### **Punto 0 | Preparación**

Retome las clases desarrolladas en el taller 4 del módulo 1. Estas serán la base para el desarrollo de este taller.

#### **Enunciado**

Retome el trabajo hecho con los Hurones y las Boas Constrictor. La guardería necesita probar que funcionen adecuadamente todos los métodos relacionados a ellos. También, un nutricionista ha puesto un alto en la cantidad de ratones que puede comer una boa constrictor.

#### **Punto 1 | Pruebas unitarias**

En primer lugar, construya dos clases que permitan realizar tests sobre las clases de las Boas y de los Hurones. Construya pruebas que permitan probar el correcto funcionamiento de:

- La función `hacer_sonido` tanto de Hurones como de Boas
- La función `calcular_flete` tanto de Hurones como de Boas
- La función `alimentar` en las boas.

#### **Punto 2 | Muchos ratones**

Modifique la función que permite comer ratones a las boas constrictor para que, si el número de ratones comidos es 10, no se aumente el contador. Por el contrario, se lanzará un `ValueError` con el mensaje “Demasiados Ratones!”.

#### **Punto 3 | Prueba y error**

Construya una pequeña clase de `Guarderia`, que tenga a su disposición 2 Boas y 2 hurones. Esta clase debe tener la función `alimentar_boa`, esta recibirá una de las boas y llamará la función de `alimentar` en cada una.

Configure con una estructura `try-except`, de modo que cuando el método de `alimentar` a la boa se retorne un mensaje que diga “La boa está llena” mientras que cuando se pueda alimentar retorne un mensaje que diga “Éxito”. También tenga en consideración el caso en que la boa no haya sido asociada y la boa recibida sea `None`, en este caso retorne un mensaje que diga “Esta Boa no existe!”

**Entrega**

No es necesario realizar una entrega. Guarde el trabajo desarrollado el día de hoy pues será fundamental y necesario para la parte 2 de este taller.