

2023/Primer Semestre

# Lenguajes y Paradigmas de Programación

**Profesor:** Ma. Loreto Arriagada - [loreto.arriagada.v@edu.uai.cl](mailto:loreto.arriagada.v@edu.uai.cl)

**Ayudante:** Joaquín Leal - [joleal@alumnos.uai.cl](mailto:joleal@alumnos.uai.cl)

# Consultas y revisión clases anteriores

# Largo de un iterador

Debido a que adhiere al concepto evaluación floja (lazy evaluation), no es posible obtener el largo de un iterador de forma directa, sin embargo hay 2 enfoques que se pueden utilizar:

- Caso 1: Convertirlo en un iterable usando **list()** y obtener el largo con **len()**
- Caso 2: Recorrerlo en un ciclo hasta **StopIteration** contando los elementos

<https://replit.com/@Ma-LoretoLoreto/clase11ejem07largoierador>

# Otras funciones de itertools

**reduce**(func, it) → Aplica una función binaria (de 2 argumentos) de manera acumulativa sobre una secuencia de valores

<https://replit.com/@GustavoJara/clase10ejem03python>

**cycle**(it) → Repite infinitamente los elementos de la secuencia de manera consecutiva

<https://replit.com/@GustavoJara/clase11ejem04itertoolscycle>

**accumulate**(it, [func=add]) → Aplica una función acumulativa a una secuencia y generar los resultados intermedios en otra secuencia

<https://replit.com/@GustavoJara/clase11ejem05itertoolsaccumulate>

# Otras funciones de itertools

**islice**(seq, [start=0], stop, [step=1]) → Realiza un recorte (“lazy slice”) de una secuencia

<https://replit.com/@GustavoJara/clase11ejem06itertoolsislice>

**partial**(func, \*args, \*kargs) → Predefine argumentos de una función, pasando el resto de los argumentos requeridos al llamarla

<https://replit.com/@GustavoJara/soluciontarea12>

**tee**(secuencia, n=2) → Crea múltiples iteradores a partir de una secuencia.

<https://replit.com/@GustavoJara/clase11ejem08itertoolstee>

# Solución Tarea

## Tarea Racing Funcional Python (Opcional)

<https://replit.com/@Ma-LoretoLoreto/racingimperativo-2>

<https://replit.com/@Ma-LoretoLoreto/racingfuncional>

# Web Scraping

# Agenda

- Web Scraping
- Trabajaremos con...
- Ejemplo Banco Central
- Ejemplo Farmacias Cruz Verde
- Actividad extracción de datos desde Multitienda



# Web Scraping

Usar bots (automatizaciones) para explotar sitios web con el objetivo de extraer o subir contenidos

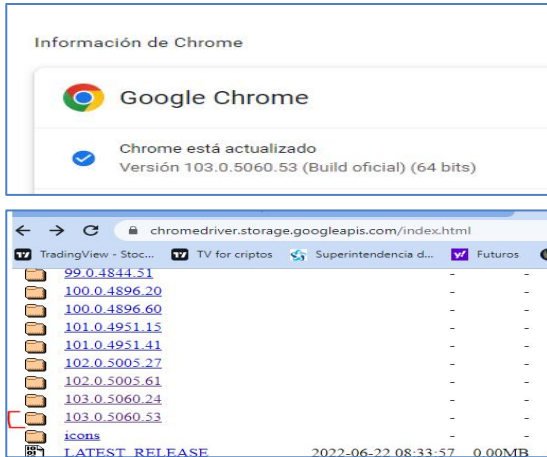


# Trabajaremos con:

- Python 3.11 <https://www.python.org/downloads/>
- Anaconda 3 <https://www.anaconda.com/products/distribution>
- Selenium [https://www.selenium.dev/documentation/.../install\\_drivers/](https://www.selenium.dev/documentation/.../install_drivers/)  
(Descargar y revisar versión del chrome)
- BeautifulSoup 4
- Parser xlmx

# Anaconda (ambiente todosolo)

- Lista de ambientes: `conda env list`
- Crear ambiente: `conda create -n todosolo python=3.11`
- Borrar ambiente: `conda env remove --name todosolo`
- Activar ambiente: `conda activate todosolo`
- Instalar Selenium: `pip install selenium`
- Instalar Beautiful Soup: `pip install beautifulsoup4`
- Instalar Parser xml: `pip install lxml`
- Exportar a yml: `conda env export > todosolo.yml`
- Importar desde yml: `conda env create -f todosolo.yml`



# Ejemplos Banco Central

- Selenium extracción directa de datos
- Selenium + BeautifulSoup extracción usando contenedor

[https://drive.google.com/drive/folders/1eGfGG8\\_3nYuN-0s3RMJKH0On6cUT10Jp?usp=sharing](https://drive.google.com/drive/folders/1eGfGG8_3nYuN-0s3RMJKH0On6cUT10Jp?usp=sharing)

# Ejemplos Farmacias Cruz Verde

- Selenium + BeautifulSoup extracción usando contenedor
- Selenium + BeautifulSoup extracción usando contenedor y paginación

[https://drive.google.com/drive/folders/1eGfGG8\\_3nYuN-0s3RMJXH0On6cUT10Jp?usp=sharing](https://drive.google.com/drive/folders/1eGfGG8_3nYuN-0s3RMJXH0On6cUT10Jp?usp=sharing)

# Actividad extracción de datos Multitienda

- Seleccione una Multitienda

- Ripley <https://simple.ripley.cl/>
- Falabella <https://www.falabella.com/falabella-cl>
- Almacenes Paris <https://www.paris.cl/>

- A partir de los ejemplos vistos en clases, implemente Web Scraping multi-paginación a partir de una lista de patrones de búsqueda

[https://drive.google.com/drive/folders/1eGfGG8\\_3nYuN-0s3RMJXH0On6cUT10Jp?usp=sharing](https://drive.google.com/drive/folders/1eGfGG8_3nYuN-0s3RMJXH0On6cUT10Jp?usp=sharing)

# ¿Qué hemos visto?

- Preparación de ambientes Python con Anaconda
- Primeros pasos con Web Scraping utilizando Selenium y BeautifulSoup
- Ejemplos extracción de datos desde Banco Central y Farmacias Cruz Verde
- Actividad extracción de datos desde Multitienda



# ¿Preguntas?

# Próxima Clase

- Prerrequisitos

Instalación de Rocketbot. Las instrucciones serán enviadas por correo.

- RPA con Rocketbot (Auditorio Edificio E)

- Mi primer robot (paradigma imperativo)
- Modularización en subrobots (paradigma procedural)

- Ejercicio RPA