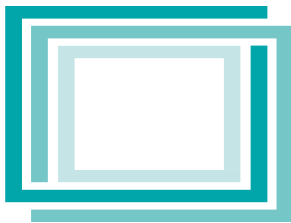


# Pre-presentación Proyecto Webscraping



UNIVERSIDAD ADOLFO IBÁÑEZ

## Integrantes:

- Rodrigo Bruna Durán (GitHub: rbrunaduran)
- José Mora González (GitHub: josemoragonzalez)
- Miguel Abarzúa Ludueña (GitHub: MiguelAbarzua)

# Contexto del Proyecto



UAI  
UNIVERSIDAD ADOLFO IBÁÑEZ

Actualmente Chile es un país sumamente dependiente de otras naciones respecto de sus necesidades alimenticias. Esto se traduce en una debilidad para la economía pues frente a situaciones como la guerra en Ucrania o desastres naturales, se generan enormes variaciones de precio debido a la escasez. Frente a esta situación nos hemos enfocado en analizar la situación productiva chilena en alimentos lácteos, con la finalidad de conocer si esta actividad productiva tiene una tendencia positiva o negativa, y establecer un panel de control que permita monitorear estos comportamientos.

Con lo anterior queremos visibilizar esta situación dejando el análisis en un plataforma abierta para que cualquier ciudadano pueda conocer esta información en visualizaciones de rápido entendimiento.



# Objetivos



UAI  
UNIVERSIDAD ADOLFO IBÁÑEZ

- Obtener datos históricos sobre la producción de productos de origen lácteo mediante procedimiento scraping con lenguaje python en sitio web de ODEPA.
- Almacenar los datos encontrados en un formato tabular mediante lenguaje python y guardarlos en un archivo de formato .csv
- Visualizar la información mediante plots y/o visualizador BI para comprender desde la perspectiva gráfica los datos.



# Pasos a seguir



UAI  
UNIVERSIDAD ADOLFO IBÁÑEZ

- Analizar la web

<https://www.odepa.gob.cl/avance-mensual-y-series-de-tiempo-de-productos-por-planta-o-region-de-la-industria-lactea>



- Crear un scraper que extraiga la información necesaria.
- Almacenar los datos en archivos csv.
- Generar visualizaciones públicas de actualización mensual usando streamlit.
- Generar un panel mediante Power BI para analizar interactivamente la información.

# Pseudocódigo (1/3)



UAI  
UNIVERSIDAD ADOLFO IBÁÑEZ

1 Importar librerías pandas, requests y BeautifulSoup

2 Crear variable url = "<https://aplicativos.odepa.gob.cl/avancemensual.do>"

3 Crear diccionario headers = {'Cookie': 'JSESSIONID=2E9275A55DAFA54F9AFEE077EA378990'}

4 Hacer un request del tipo GET y almacenar el resultado en la variable page

5 Crear diccionario campos = {'mesini': 'cboMesIni', 'anioini': 'cboAgnoIni', 'mesfin': 'cboMesFin', 'aniofin': 'cboAgnoFin', 'producto': 'cboProducto', 'region': 'cboRegion'}

6 Crear diccionario 'parametros' vacío

7 Parsear el contenido de la variable page y almacenar el resultado en la variable soup

8 Para cada key y value en el diccionario campos:

8.1 Encontrar el select con el id igual a value y almacenarlo en la variable temp

8.2 Obtener el listado de valores option almacenados en temp y almacenarlo en la variable valor

8.3 Adicionar el listado en el diccionario 'parametros'



# Pseudocódigo (2/3)



UAI  
UNIVERSIDAD ADOLFO IBÁÑEZ

9 Establecer los años a revisar minimo y maximo y almacenar el valor en la variable anios

10 Establecer los productos a revisar almacenarlo en la variable productos

11 Establecer las regiones a revisar y almacenarlo en la variable regiones

12 Para cada producto en la variable productos:

12.1 Para cada región en la variable regiones:

12.1.1 Crear el diccionario payload con los valores payload = {'dataExport': ' ', 'compressed': 'false', 'fileNameExcel': ' ', 'decimales': ' ', 'cboMesIni': '01', 'cboAñoIni': valor mínimo de año almacenado en la variable anios, 'cboMesFin': '06', 'cboAñoFin': valor máximo de año almacenado en la variable anios, 'cboProducto': producto i en productos, 'rdoTipo': 'region', 'cboRegion': region i en regiones, 'rdoFormatoTabla': 'mesaño'}

12.1.2 Hacer el request tipo POST con usando las variables url, headers y payload y almacenar el resultado en la variable page

12.1.3 Crear un dataframe vacío llamado dfFinal en el que se almacenarán los resultados

# Pseudocódigo (3/3)



UAI  
UNIVERSIDAD ADOLFO IBÁÑEZ

12.1.4 Si el request no devuelve error:

12.1.4.1 Parsear el contenido de la variable page y almacenar el resultado en la variable soup

12.1.4.2 Encontrar la tabla con valores y almacenarla en la variable rows

12.1.4.3 Encontrar los encabezados de la tabla y almacenarla en la variable columns

12.1.4.4 Construir la tabla usando las variables columns y rows

12.1.4.5 Añadir la columna Producto a la tabla, haciendo que su contenido sea igual al producto i

12.1.4.6 Añadir la columna Region a la tabla, haciendo que su contenido sea igual al región i

12.1.4.7 Adicionar los valores de tabla en el dataframe dfffinal

# Entregables

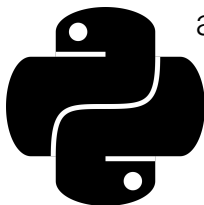


UAI  
UNIVERSIDAD ADOLFO IBÁÑEZ



- Archivo con columnas:
  - Meses: Meses del año en palabras (string).
  - Producto: Productos lacteos, ejemplo: leche condensada en litros (string).
  - Region: nombre de la regiones (string).
  - Año: Año de la producción (integer).
  - Valor: Cantidad producida de cada producto (integer).

Este archivo contendrá más de 27.000 registros para el periodo 2002-2022 y su actualización será mensual.



- Archivo con código de scraping (.py).
- Archivo con código de plots (.py)
- Archivo con código para levantar una aplicación en Streamlit (.py).



Power BI

Archivo con dashboard (.pbix).