

TICS200: Proyecto Semestral

Universidad Adolfo Ibáñez

Profesor: Loreto Arriagada - loreto.arriagada.v@edu.uai.cl

Ayudante: Joaquín Leal - joleal@alumnos.uai.cl

29 de mayo de 2023

Objetivos

- Estructurar una aplicación multiparadigma, aplicando paradigma **procedural**, **orientado a objetos** y **funcional** en lenguaje **Python**.
- Desarrollar un entregable funcional utilizando **herramientas de desarrollo en ambiente local**.

1. Productos de Tecnología

Debido al excelente desempeño en las aplicaciones desarrolladas por Ustedes para la franquicia **BigMuscle**, tanto en paradigma procedural, como en paradigma orientado a objetos, y por haber salido sanos y salvos del **Laberinto de Creta** en paradigma funcional, la empresa **TodoSolo** le encarga que desarrolle una aplicación **multiparadigma** que permita realizar el procesamiento de información de productos de tecnología a partir de patrones de búsqueda, capturando para ello la información disponible en Internet. La aplicación deberá realizar lo siguiente:

- Leer un paquete de patrones de búsqueda de productos tecnológicos.
- Para cada producto tecnológico del paquete, capturar la información disponible en **al menos 2** sitios web de las Multitiendas **Ripley**, **Falabella** y **Almacenes Paris**.
- Capturar el valor de la UF en el sitio web del **Banco Central de Chile**.
- Almacenar los resultados de todas las búsquedas en 2 archivos de salida.
- Generar estadísticas de la información procesada.

2. Requerimientos NO Funcionales

1. Desarrollar la aplicación en lenguaje **Python 3+** y ambiente **Anaconda**.
2. Desarrollar en **paradigma procedural** la lectura del archivo de entrada y de los datos disponibles en Internet utilizando técnica **Web Scraping**.
3. Desarrollar en **paradigma orientado a objetos** las estructuras de datos para administrar todo lo relativo a información de productos tecnológicos y precios, utilizando solo **mangling** + decoradores **@property** y **@setter** para todas las propiedades de las clases.
4. Desarrollar en **paradigma funcional** la generación de los archivos de salida y las estadísticas solicitadas.

3. Requerimientos Funcionales

1. Realizar la captura del valor de la unidad de fomento (UF) desde el sitio web del **Banco Central de Chile** <https://www.bcentral.cl/inicio> y guardar la fecha y el valor de la unidad de fomento en el archivo de salida **parametros.csv** (se adjunta ejemplo), el que contendrá solo un registro de datos.
2. Leer paquete de productos tecnológicos desde el archivo de entrada **patrones_busqueda.txt** (se adjunta ejemplo) y por cada registro, hacer búsquedas **en al menos 2** de los sitios web de las Multitiendas:
 - Ripley <https://simple.ripley.cl/>
 - Falabella <https://www.falabella.com/falabella-cl>
 - Almacenes Paris <https://www.paris.cl/>
3. Por cada búsqueda realizada, deberá capturar todos los productos tecnológicos disponibles, incluyendo aquellos que se encuentren paginados¹. Los resultados de todas las apariciones deben quedar almacenadas en el archivo de salida **todosolo.csv** (se adjunta ejemplo), el que debe contener los siguientes campos:
 - Patrón de búsqueda.
 - Multitienda (Ripley, Falabella o Almacenes Paris).
 - Descripción producto tecnológico.
 - Precio en pesos chilenos, sin decimales, que corresponde al precio final al público publicado, incluida todas las ofertas si es que la hay capturando el menor precio.
 - Precio en UF, con 2 decimales, que corresponde a la conversión del precio en pesos chilenos a unidades de fomento.
4. Generar las siguientes estadísticas por consola:
 - **Precios mínimos por patrón de búsqueda y Multitienda:** Patrón de búsqueda, Multitienda, precio **mínimo** en pesos y UF. 1 registro por cada combinación de **patrón de búsqueda y Multitienda**. Ordenado por patrón de búsqueda más Multitienda.
 - **Precios máximos por patrón de búsqueda y Multitienda:** Patrón de búsqueda, Multitienda, precio **máximo** en pesos y UF. 1 registro por cada combinación de **patrón de búsqueda y Multitienda**. Ordenado por patrón de búsqueda más Multitienda.

¹ Por ejemplo, al buscar el producto tecnológico **notebook hp** en el sitio web de **Falabella** se realiza paginación, para lo cual aparecen **7 páginas** de información y en cada una de ellas **hasta 56 productos tecnológicos**. Usted deberá captura la información de todos los productos tecnológicos de todas las páginas.

- **Precios promedio por patrón de búsqueda y Multitienda:** Patrón de búsqueda, Multitienda, precio **promedio** en pesos y UF. 1 registro por cada combinación de **patrón de búsqueda y Multitienda**. Ordenado por patrón de búsqueda más Multitienda.
- **Precios mínimos por patrón de búsqueda:** Patrón de búsqueda, Multitienda con el menor precio, descripción del producto tecnológico con el menor precio, precio **mínimo** en pesos y UF. 1 registro por cada patrón de búsqueda. Ordenado por patrón de búsqueda.
- **Precios máximos por patrón de búsqueda:** Patrón de búsqueda, Multitienda con el mayor precio, descripción del producto tecnológico con el mayor precio, precio **máximo** en pesos y UF. 1 registro por cada patrón de búsqueda. Ordenado por patrón de búsqueda.
- **Precios promedio por patrón de búsqueda:** Patrón de búsqueda, precio **promedio** en pesos y UF. 1 registro por cada patrón de búsqueda. Ordenado por patrón de búsqueda.
- **Precio mínimo en el paquete (mínimo del bulto).** Patrón de búsqueda con el menor precio, Multitienda con el menor precio, descripción del producto tecnológico con el menor precio, precio mínimo en pesos y UF. 1 registro.
- **Precio máximo en el paquete (máximo del bulto).** Patrón de búsqueda con el mayor precio, Multitienda con el mayor precio, descripción del producto tecnológico con el mayor precio, precio máximo en pesos y UF. 1 registro.

4. Sobre la entrega

- La aplicación debe ser implementada en lenguaje **Python 3+** con **paradigmas procedural, orientado a objetos y funcional**, según corresponda a lo indicado en los **Requerimientos NO Funcionales**.
- Los equipos de trabajo pueden estar integrados por hasta 4 personas.
- El plazo para entrega es el lunes, 26 de junio de 2023 a las 13:59:59.
- Se debe entregar solo un archivo y empaquetado en formato zip o rar.
- La corrección se verificará en la **plataforma de desarrollo local con Anaconda** (no se permite **replit** para este entregable), para lo cual deben incluir en el paquete de entrega el archivo **env_todosolo.yml** que se crea desde la consola de **Anaconda** con el comando: **conda env export > env_todosolo.yml**
- El entregable señalado, debe incluir un comentario con los integrantes del grupo y solo uno de ellos debe realizar la entrega en el buzón habilitado en la plataforma WebCursos.
- Se aplicará 1 punto de descuento por cada día de atraso o fracción.

5. Sobre el Plagio

1. El plagio es el uso de las ideas o trabajo de otra persona sin el adecuado consentimiento. El plagio puede ser intencional o no. El plagio intencional es el claro intento de hacer pasar el trabajo o ideas ajenas como el suyo propio para su beneficio. El plagio no intencional puede ocurrir si Ud. no conoce el mecanismo adecuado de referenciar la fuente de sus ideas e información. Si utiliza códigos fuentes obtenidos de Internet, referéncielos con un comentario en su código fuente que indique el link a la fuente y las modificaciones que fueron realizadas. Si no está seguro de los métodos aceptados para referenciar, debería consultar con su profesor, tutor o personal de biblioteca. El plagio comprobado es una actitud que puede resultar en sanciones disciplinarias y/o en la exclusión de la Universidad (Reglamento del Estudiante de Pregrado).
2. Los entregables son comparados con un detector de similitudes de código fuente, en el que se incorporan al análisis todos los entregables de las secciones de Santiago y Viña del Mar, por lo que es recomendable que no use por ninguna vía, incluido códigos públicos de **replit**, autorías de terceros sin la debida referencia. Con esto, evita las medidas precedentemente señaladas.
3. Para que no se exponga a que sus ideas y trabajos en desarrollo sean plagiados por terceros, no los publique en Internet mientras los realiza (por ejemplo, en cuenta básica **replit** que por defecto deja públicos sus códigos fuentes). Con esto, evita las medidas precedentemente señaladas a causa de terceros.

6. Pauta de Evaluación

La aplicación será revisada utilizando la siguiente pauta de evaluación:

Concepto de evaluación	Descripción	Puntos	Descuento
Punto base	Punto base	1.0	No aplica
Aplicación de los paradigmas	Paradigma procedural: <ul style="list-style-type: none"> - NO usar paradigma imperativo - Usar un orquestador - Usar procedimientos/funciones con propósitos específicos a cada uno Paradigma orientado a objetos: <ul style="list-style-type: none"> - Usar solo mangling + decoradores @property y @setter para las propiedades de las clases. Paradigma funcional: <ul style="list-style-type: none"> - NO usar bucles nativos (for, while u otros), en su reemplazo debe utilizar funciones de orden superior (map, filter o reduce), list comprehension o generador de expresiones - No usar condicionales nativos (if u otros), en su reemplazo debe usar función de orden superior filter, list comprehension o generador de expresiones con filtro - NO usar variables mutables. Excepcionalmente se permite uso reducido de variables inmutables, es decir, como constantes (en mayúsculas según norma) y sin realizar modificación de ellas después de ser definidas. - Las funciones deben ser puras, es decir, no deben tener efectos colaterales como alterar listas, variables de globales o de niveles superiores 	4.0	0.5 por cada desviación
Funcionalidades	Cumplimiento del Requerimientos Funcionales	1.0	0.5 por cada incumplimiento funcional
Lógica y lenguaje	Uso adecuado del lenguaje y lógica empleada: <ul style="list-style-type: none"> - Optimización códigos fuentes - Uso de comentarios - Orden del código fuente - Uso adecuado de las estructuras de datos y sentencias del lenguaje 	1.0	0.3 por cada desviación 0.5 por cada caída del programa

7. Bonus Track

Se premiará con 0,5 puntos adicionales por el desarrollo de un menú en **paradigma orientado a objetos**, que contenga como opciones las estadísticas solicitadas más una opción de salida de la aplicación. Cada opción deberá mostrar por consola la estadística que le corresponda a la opción.

Se premiará con 0,5 puntos adicionales por ampliar la búsqueda a la tercera Multitienda.