



PONTIFICIA UNIVERSIDAD CATÓLICA DE CHILE
ESCUELA DE INGENIERÍA
DEPARTAMENTO DE CIENCIA DE LA COMPUTACIÓN

IIC2233 - Programación Avanzada
2° semestre 2015

Actividad 07

Programación funcional: funciones lambda, map, filter y reduce.

Instrucciones

Ingenieros de la NASA han recurrido al DCC de manera urgente para analizar información de sus servidores web. La agencia espacial teme ser el próximo objetivo del grupo *Anonymous*, por lo que han solicitado ayuda para generar un informe que les permita tomar medidas a fin de garantizar el buen funcionamiento de los servidores durante el posible ataque. Lamentablemente el DCC se encuentra muy ocupado desarrollando un proyecto llamado *Bummer UC*, por lo que la generación del informe se le ha delegado a los talentosos alumnos del curso Programación Avanzada.

Dentro de la carpeta `utils` podrá encontrar un archivo generador de logs, el que para su sorpresa se llama `generador.py`, el que deberá ejecutar para generar el archivo de texto plano `nasa_logs_week.txt`. Cada línea del archivo de texto representa una solicitud hecha al servidor y contiene información muy valiosa, formateada de acuerdo al Apache Common Log Format. Asegúrese de que el archivo con extensión `.txt` quede inmediatamente dentro de la carpeta `utils`.

En un intento por hacer su vida más sencilla, los ingenieros de la NASA le han proporcionado un programa dentro de la carpeta `utils` llamado `parser.py` que define las clases `ApacheLogsParser` y `ApacheLog`. Al llamar el método `get_apache_logs` de una instancia de la clase `ApacheLogsParser`, el programa leerá línea por línea el archivo generado en el párrafo anterior y retornará una lista de `ApacheLogs`. Cada instancia de `ApacheLog` tiene los siguientes atributos:

- `remote`: dirección del computador que solicitó el recurso.
- `timestamp`: fecha y hora en que el recurso fue solicitado.
- `request`: dirección del recurso solicitado.
- `status`: código HTTP que envió el servidor.
- `size`: número de bytes transferidos.

Requerimientos

- Mediante el paradigma de programación funcional, debe generar los siguientes informes:
 - Bytes totales transferidos durante la semana.
 - Número de solicitudes que produjeron un error en el servidor.

- Número de solicitudes que fueron respondidas sin problemas.
- Cuál fue el recurso (URL) más solicitado.
- Deberá completar cada uno de los métodos definidos dentro de la clase `BigAnalizador` en el archivo `main.py`. Cada método deberá imprimir la respuesta que genera.
- Usted es libre de agregar los métodos/clases/funciones que estime conveniente para resolver correctamente esta actividad

Notas

- Si resuelve algún problema sin programación funcional tendrá 0 puntos en esa parte (i.e: entre otras cosas: no usar comprensión de listas, diccionarios, etc..).
- Tenga cuidado de **no** agregar el archivo `nasa_logs_week.txt` a su repositorio debido al gran tamaño que tiene. Subirlo a *Github* junto con la solución será penalizado.
- Está prohibido el uso de las funciones `sum`, `len` y el uso de ciclos `for` y `while`.

To - DO

- (1.00 pt) Imprimir número de bytes transferidos.
- (1.00 pt) Imprimir número de solicitudes erróneas.
- (1.00 pt) Imprimir número de solicitudes exitosas.
- (3.00 pt) Imprimir URL del recurso más solicitado.

Tips

- Para la parte de la URL más solicitada la función `groupby` del módulo `itertools` puede ser de utilidad.
- Recuerde que para usar la función `reduce` debe importarla desde el módulo `functools`.
- Una solicitud es exitosa si el servidor responde un código HTTP igual a:
 - 200 (OK)
 - 302 (Found)
 - 304 (Not Modified)
- Una solicitud falla debido a un error si se responde con un código HTTP igual a:
 - 404 (Not Found)
 - 500 (Internal Server Error)
 - 501 (Not Implemented)