



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

IIC2233 - Programación Avanzada
1er semestre 2016

Actividad 14

Serialización Binaria - JSON

Instrucciones

Usted es un especialista de redes sociales trabajando para la empresa IICFriends, donde su superior le ha encargado reunir toda la información que pueda de un grupo de personas utilizando sus perfiles de distintas redes sociales. Utilizando sus conocimientos de serialización binaria y `json`, usted debe reunir datos de perfiles de Facebook, Instagram y Twitter, combinarlos y actualizarlos.

Lo primero que usted debe hacer es leer la información de las distintas redes sociales. Para esto, IICFriends le entrega un archivo `main.py` que contiene las clases de los distintos tipos de perfil con sus atributos para su correcta deserialización, estas son `FacebookProfile`, `TwitterProfile` y `InstagramProfile`. **Debe** usar estas clases **sin modificarlas** para leer la información. Los distintos perfiles se encuentran en la carpeta `db` en las subcarpetas `facebook` e `instagram`, donde los perfiles están serializados de forma **binaria**, y en la subcarpeta `twitter`, donde los perfiles están serializados en formato `json`.

Luego debe juntar la información de cada persona, utilizando el identificador de persona `pid`. Para ello debe implementar una clase `Persona`, que será la clase que agrupe la información de todas las redes sociales de una persona.

Cada persona **debe** contener los siguientes atributos:

- `Pid` (Id común en las 3 redes sociales).
- Nombre Completo (String).
- Twitter handle (String).
- Instagram handle (String).
- Amigos de Facebook (Una lista con los nombres de sus amigos en Facebook).
- Seguidos en Twitter (Una lista con los handles de aquellos que la persona sigue en Twitter).
- Seguidores de Twitter (Una lista con los handles de aquellos que siguen a la persona en Twitter).
- Seguidos en Instagram (Una lista con los handles de aquellos que la persona sigue en Instagram).
- Seguidores de Instagram (Una lista con los handles de aquellos que siguen a la persona en Instagram).

Una vez que tienen toda la información de una persona deberán serializar esta información en formato `json` en la subcarpeta `personas`. Además, hay que actualizar el formato de los archivos originales de las redes, bajo nuevos estándares. Esta vez todos deben estar escritos en `json` y siguiendo distintos formatos:

- Facebook: `profile_id`: pid, `full_name`: nombre, `twitter`: Twitter handle, `instagram`: Instagram handle, `friend_list`: lista de amigos de Facebook.
- Twitter: `id`: pid, `handle`: Twitter handle, `name`: nombre, `followers`: seguidores de Twitter, `follows`: seguidos de Twitter.
- Instagram: `identification`: pid, `user`: Instagram handle, `alias`: nombre, `followers`: seguidores de Instagram, `follows`: seguidos de Instagram.

Requerimientos

`Persona` debe completar los siguientes métodos:

- `get_social(archivo, red_social)`: Debe retornar el perfil de red social especificado por `red_social` deserializando `archivo`.
- `get_persona(id)`: Debe retornar un objeto de tipo `Persona` con todos los atributos de la persona con el ID respectivo.
- `write_persona(persona)`: Debe serializar un objeto de tipo persona que está siendo recibido como parámetro.
- `existe_persona(id)`: Retorna `True` si existe el archivo correspondiente al de la persona con el ID indicado. En caso contrario, retorna `False`.
- `add_info(profile)`: Debe añadir la información dentro de `profile` a la persona que corresponda. Si el archivo no existía antes, debe crearlo antes.
- `write_social(persona)`: Debe serializar un objeto de tipo persona 3 veces bajo los estándares de cada red social en formato `json`.

Todos los métodos deben leer y escribir los archivos que sean necesarios. La idea no es mantener los objetos en memoria.

Notas

- Todos los archivos van en las subcarpetas de la carpeta `db`
- Los nombres de los archivos entregados no reflejan nada, solo una vez deserializados usted podrá identificar su contenido.
- Existen subcarpetas vacías: `personas`, `facebook_actualizados`, `twitter_actualizados` e `instagram_actualizados`, para que usted guarde las versiones actualizadas de la información de cada persona y para cada red social.
- Su programa debe realizar todo lo especificado al correrlo. Es decir, leer toda la información, crear y serializar todas las personas y serializar los datos de redes sociales.

To - DO

- (2.00 pts) Deserialización
 - (0.50 pts) Leer información de Facebook
 - (0.50 pts) Leer información de Instagram
 - (0.50 pts) Leer información de Twitter
 - (0.50 pts) Implementar `get_social(archivo, red_social)`
- (2.00 pts) Creación Persona
 - (0.50 pts) Implementar `get_persona(id)`
 - (0.50 pts) Implementar `existe_persona(id)`
 - (1.00 pts) Implementar `add_info(profile)`
- (2.00 pts) Serialización Persona
 - (1.00 pts) Implementar `write_persona(persona)`
 - (1.00 pts) Implementar `write_social(persona)`

Tips

- El método `os.listdir(directorio)` entrega una lista de strings con todos los elementos que contiene la carpeta `directorio`.

Entrega

- **Lugar:** GIT - Carpeta: Actividades/AC14
- **Hora:** 16:55