

INSTRUCCIONES

Antes de comenzar, lea atentamente las instrucciones, los requerimientos de entrega y el problema

- La evaluación es de carácter individual. Cualquier indicio de intervención de otra persona será calificado con nota mínima a la evaluación y quedará impedido de rendir cualquier instancia de evaluación optativa.
- En caso de detectar programas iguales o con suficientes similitudes entre sí, tanto en una misma sección como en secciones distintas, se calificará con la nota mínima a todos los involucrados. Para ello se compararán las entregas con un algoritmo de reconocimiento de texto y se revisarán los casos sospechosos mediante un proceso de validación visual que realizará el cuerpo docente del curso.
- Estudiantes que entreguen fuera de plazo serán calificados con nota mínima en esta evaluación.
- Entregas que se realicen por vías distintas a Campus Virtual serán calificados con nota mínima. En caso de que la plataforma presente un problema, puede enviar su archivo como respaldo al correo de contacto del profesor de Teoría, y posteriormente subir el archivo a Campus Virtual.
- Estudiantes que entreguen sin seguir los criterios de identificación estipulados en el apartado de “entrega” serán calificados con nota mínima en esta evaluación.
- La difusión de este enunciado, solicitando ayuda, ofreciendo dinero a cambio de la resolución e incluso la publicación de éste dentro del plazo de desarrollo de la tarea, se considera un acto deshonesto, que daña la imagen de la Facultad de Ingeniería y de la Universidad de Santiago de Chile y significará la reprobación inmediata de la asignatura y la aplicación de los procedimientos disciplinarios que el reglamento de ética de la Universidad dispone para estos casos.
- Considere que cualquier supuesto que el estudiante haga debe ser explicitado en los comentarios de código.
- La subida del archivo es responsabilidad de su autor, por lo que, archivos que no estén en el formato estipulado, que vengan corruptos o con problemas para ser leídos, no serán revisados.
- La evaluación es optativa, y reemplaza la peor nota entre promedio de quizzes, PEP 1 y PEP 2. Este reemplazo se realiza tanto si la nota es más alta, como si fuese más baja que la nota obtenida previamente.

- Cualquier situación no contemplada en este documento será dirimida por la Facultad de Ingeniería.

ENTREGA

Se recibirán soluciones hasta las 23:59 del día 29 de enero de 2022, en la plataforma Campus Virtual, en el espacio habilitado para ello en la pestaña “Evaluaciones -> Pruebas”.

Se requiere entregar un único archivo .py con la solución al problema, el archivo debe llevar por nombre el RUT de su autor (sin puntos, ni guión) por ejemplo:

209414321K.py

Además, al inicio del archivo, se debe añadir el siguiente encabezado del programa, con los datos solicitados para identificar su trabajo (Rellene el encabezado del programa con sus datos personales en formato IDÉNTICO al indicado en el ejemplo del Código a continuación):

```
# FUNDAMENTOS DE PROGRAMACIÓN PARA INGENIERÍA
# SECCIÓN DEL CURSO: 2-L-1
# PROFESOR DE TEORÍA: FELIPE MORENO
# PROFESOR DE LABORATORIO: JUAN PADILLA
#
# AUTOR
# NOMBRE: Juan Carlos Perez Gonzalez
# RUT: 20.345.432-2
# CARRERA: Ingeniería Civil Mecánica
# DESCRIPCIÓN DEL PROGRAMA ...<CONTINÚE CON EL PROGRAMA A PARTIR DE AQUÍ>
```

Código 1: Ejemplo de encabezado

PROBLEMA

La gente de Spotify está tratando de generar una herramienta que presente estadísticas al usuario que vayan más allá del resumen que generan cada año. Esto debido a que se han dado cuenta de que la gente compartiendo sus estadísticas musicales les reporta anualmente un flujo nuevo de usuarios comprando el servicio cada fin de año.

Afortunadamente, la compañía cuenta con registros completos de la actividad de sus usuarios. Cada registro del usuario es un archivo que tiene por nombre el ID interno del usuario y contiene líneas de registro, donde cada línea lleva la fecha y hora en que se escuchó cada canción, la canción y su artista, el género al que pertenece y la duración de la canción separados por comas.

A partir de estos registros, Spotify desea generar un programa que permita al usuario, a partir de su ID y un mes del año, ver su actividad durante ese período en específico, un gráfico de los minutos escuchados por cada género y su Top 10 de artistas de ese período.

Un ejemplo de cómo estos archivos se presentan puede verse a continuación:

User430092.txt
2020-01-01 12:56:23,Sonic Youth,Sacred Trickster,Blues,05:09
2020-01-01 14:15:25,Red,Faceless,Rock,02:03
2020-01-01 15:15:43,The Bones Of J.r. Jones,Sing Sing,Blues,02:29
2020-01-01 15:44:06,Breaking Benjamin,So Cold,Pop,06:53
2020-01-01 16:11:19,Reignwolf,Lonely Sunday,Blues,02:20
2020-01-01 17:21:29,Fantan Mojah,Thanks & Praise,Reggae,04:28
2020-01-01 21:05:07,The Chiffons,He's So Fine,Pop,06:03
2020-01-02 13:27:27,Ajr,I'm Not Famous,Rock,04:16
...

Como implementar directamente esto en la aplicación es costoso, se le ha solicitado a usted la construcción de un prototipo de esta aplicación en dónde, a partir del ID de un usuario y un par año-mes, la salida sea:

- Un gráfico de barras indicando cuántos minutos de cada género fueron escuchados el mes escogido.
- Un archivo de texto “TOP-10-AÑO-MES.txt” con los 10 artistas más escuchados, contando las veces en que se escucharon canciones de dicho artista. A continuación, se presenta un ejemplo de este archivo.

TOP-10-2022-01.txt
1. Aurora, 320 reproducciones
2. Taylor Swift, 240 reproducciones
3. Lana Del Rey, 123 reproducciones
4. Chayanne, 122 reproducciones
5. Björk, 35 reproducciones
6. Zalo Reyes, 30 reproducciones
7. Beto Cuevas, 29 reproducciones
8. Princesa Alba, 23 reproducciones
9. Ghost, 12 reproducciones
10. Volbeat, 5 reproducciones

Para su solución considere que:

- Los archivos que se entregan junto con la prueba **solo constituyen ejemplos** y que su solución debería funcionar para cualquier archivo que respete el mismo formato.
- No sabe de antemano cuántos géneros podría escuchar cada usuario.
- Es posible que un usuario en un mes no haya escuchado música. Por ejemplo, si el archivo de registros indica que el usuario se incorporó el 2020 a la plataforma, al intentar generar el reporte de cualquier mes del 2019 el programa debería informar que esto no es posible.
- Es posible que un usuario en un mes haya escuchado un solo género. El programa en este caso debería generar igual el gráfico.

- Es posible que un usuario haya escuchado a menos de 10 artistas en un mes. El programa en este caso debería generar igual el TOP, sólo considerando los artistas que haya.

Restricciones de implementación

Para esta prueba, se prohíbe el uso de las siguientes herramientas de programación:

- Funciones y métodos para obtener mínimos y máximos, así como para ordenar listas.
- Construcciones que usen las palabras reservadas `try` y `except`.
- Expresiones regulares.
- Listas por comprensión.
- Uso de strings con formato, ya sea utilizando la sintaxis de llaves `{, }` o de porcentaje `%`.
- Los tipos de dato conjunto (`set`), tupla (`tuple`) y diccionario (`dict`).
- Módulos que no han sido vistos en clases de teoría (e.j. `pandas`, `csv`, etc).
- Programación orientada a objetos, es decir, definición de clases y métodos.
- Las funciones `map()` y `filter()`.

El uso de alguna de estas herramientas significará la nota mínima.

OBSERVACIONES

En caso de que su equipo le entregue el siguiente error al leer o escribir el archivo:

UnicodeDecodeError: 'charmap' codec can't decode byte

Utilice el parámetro `encoding='utf-8'` luego del modo de apertura al utilizar la función `open()`.

EVALUACIÓN

Considere que la evaluación considera 60 puntos como puntaje máximo y una exigencia del 60%.

Los aspectos a evaluar de la solución son:

- Abstracción de datos.
- Abstracción de procesos.
- Buenas prácticas de programación.
- Organización del código.
- Manejo de estructuras de control.
- Manejo de tipos de datos.
- Uso de funciones.

- Construcción del gráfico.
- Flujo de programa.
- Cálculo de la solución.