

# Programación I - Examen final

Enero 2019

**Deustube** es un nuevo servicio de streaming de video para estudiantes de la Universidad de Deusto. Te han contratado para crear una aplicación que les ayude a gestionar sus canales en la que tendrás que programar las siguientes clases y métodos:

## Clase Video (1 puntos)

Esta clase gestiona los vídeos disponibles en Deustube y el número de likes y dislikes que reciben.

Sus propiedades son:

- **id**: cadena de caracteres (String), por ejemplo "X78234vFG".
- **título**: cadena de caracteres (String), por ejemplo "Mi primer video".
  - **duración**: número de segundos (double), mayor o igual a cero.
- **likes**: número de likes (entero), mayor o igual a cero.
- **dislikes**: número de dislikes (entero), mayor o igual a cero.

Sus métodos son:

- Constructor **sin argumentos, con argumentos y copia**.
- **Getters y setters**.
- Método **toString** que devuelve un String con la dirección web del vídeo "http://deustube.com/id", por ejemplo: "<http://deustube.com/X78234vFG>".

Tanto el constructor con argumentos como los setters de la clase Video deberán comprobar que son valores válidos los que se asignan a los atributos (consulta las indicaciones aportadas entre paréntesis en cada uno de ellos previamente). Si no lo son, el constructor pondrá valores por defecto y los setters dejarán los valores que ya estaban.

## Clase Canal (2 puntos)

Esta clase gestiona los canales en Deustube.

Sus propiedades son:

- **usuario**: cadena de caracteres, por ejemplo "garaizar".
- **título**: cadena de caracteres, por ejemplo "El canal de Pablo".
- **seguidores**: array de cadenas de caracteres , por ejemplo {"garaizar", "bosanz", "eguiluz"}.
- **videos**: array de objetos de la clase Video.

Sus métodos son:

- Constructor **con argumentos y sin argumentos**
- **Getters y setters**.

- Método **toString** que devuelve un String apropiado para mostrar por pantalla los atributos de la clase. Ejemplo:

**http://deustube.com/usuario (n vídeos, m seguidores)**

- Método **getNumeroSeguidores** que devuelve un int correspondiente al número de seguidores del canal.
- Método **getNumeroVideos** que devuelve un int correspondiente al número de vídeos disponibles en el canal.

## Clase Deustube

Clase en la que se implementarán los siguientes métodos estáticos:

- **main**: crea un array con capacidad para 5000 canales en el que haya al menos 3 canales con al menos 5 seguidores y 5 vídeos cada uno, así como un menú con las siguientes opciones **(1,5 puntos)**:
  - Mostrar las direcciones web de todos los vídeos disponibles en Deustube (ver método mostrarVideos).
  - Buscar canal con mayor número de likes (ver método buscarMasLikes).
  - Mostrar la duración total de todos los vídeos almacenados en Deustube (ver método getDuracionTotal).
  - Solicitar un nombre de usuario y mostrar todos los canales que sigue ese usuario (ver mostrarCanalesSeguidos).
- **mostrarVideos**: recibe un array de objetos de la clase Canal y muestra por pantalla el listado completo de vídeos de todos los canales. **(1 punto)**
- **buscarMasLikes**: recibe un array de objetos de la clase Canal y devuelve el Canal que más likes acumulados tenga. **(1,5 punto)**
- **getDuracionTotal**: recibe un array de objetos de la clase Canal y devuelve el total de segundos de todos los vídeos de todos los canales. **(1,5 puntos)**
- **mostrarCanalesSeguidos**: recibe un nombre de usuario y un array de objetos de la clase Canal y muestra por pantalla el listado de canales que sigue ese usuario. **(1,5 puntos)**

## Extra: sistema de recomendación (3 puntos)

Realiza esta parte solo si crees que tienes el resto del ejercicio bien. Si no, es mejor que no dediques tiempo a esta parte porque solo se tendrá en cuenta a partir de 7 puntos sobre 10 en el resto del examen.

Crea una clase Usuario con las siguientes propiedades:

- **id**: cadena de caracteres (String), por ejemplo "garaizar".
- **historial**: array de objetos de la clase Video.

Sus métodos son:

- Constructor **con argumentos y sin argumentos**
- **Getters y setters.**
- Método **toString** que devuelve un String apropiado para mostrar por pantalla los atributos de la clase.
- Método **imprimirHistorial** que muestra el historial de un usuario.
- Método **anyadirVideoHistorial** que añade recibe vídeo como parámetro y lo añade al historial de un usuario.

Además, añade en la clase Deustube lo siguiente:

- En el método **main**: crea un array al menos 5 usuarios y 5 vídeos en su historial. Haz que algunos de esos vídeos coincidan entre diferentes usuarios y que otros no coincidan con los vídeos del historial de otros usuarios (ejemplo: el usuario 1 ha visto los vídeos A, B, C, D y E y el usuario 2 ha visto los vídeos A, B, X, Y, Z).
- En el método **main**: añade una nueva opción al menú que se llame Sistema de recomendación, solicite un id de usuario y llame al método sistemaRecomendacion.
- Un nuevo método estático llamado **sistemaRecomendacion** que recibe un id de usuario y el array de usuarios y haga lo siguiente: 1) buscar en el array de usuarios el usuario que más vídeos en común haya visto; 2) una vez encontrado ese usuario, mostrar uno de los vídeos que no coinciden en el historial de ambos usuarios (ejemplo: el usuario 1 ha visto los vídeos A, B, C, D y E, el usuario 2 ha visto los vídeos A, B, X, Y, Z y el usuario 3 ha visto los vídeos A, B, C, S, T, cuando nos pidan una recomendación para el usuario 1, encontraremos que el usuario 3 es quien tiene el historial más parecido y miraremos los vídeos que no coinciden en sus historiales -S y T- para mostrar finalmente uno de ellos -T, por ejemplo-).
- El programa deberá mostrar algo similar a esto:

**Deustube: audientia melior auro.**

1. Mostrar direcciones web de todos los vídeos.
2. Buscar canal con más likes.
3. Mostrar la duración total de todos los vídeos.
4. Mostrar todos los canales que sigue un usuario.
0. Salir

**Opción: 1**

**Listado de vídeos almacenados en Deustube:**

<http://deustube.com/X78234vFG>  
<http://deustube.com/45613HaXZ>  
<http://deustube.com/A164RTH2G>  
<http://deustube.com/79DFB3vHJ>  
<http://deustube.com/X4823fweA>  
...

1. Mostrar direcciones web de todos los vídeos.
2. Buscar canal con más likes.
3. Mostrar la duración total de todos los vídeos.
4. Mostrar todos los canales que sigue un usuario.

0. Salir

Opción: 2

Canal con más likes: El canal de Pablo, 253 likes.

1. Mostrar direcciones web de todos los vídeos.
  2. Buscar canal con más likes.
  3. Mostrar la duración total de todos los vídeos.
  4. Mostrar todos los canales que sigue un usuario.
0. Salir

Opción: 3

Duración total: 346244623.23 segundos.

1. Mostrar direcciones web de todos los vídeos.
  2. Buscar canal con más likes.
  3. Mostrar la duración total de todos los vídeos.
  4. Mostrar todos los canales que sigue un usuario.
0. Salir

Opción: 4

Usuario: Pablo.

El usuario Pablo sigue los siguientes canales:

El canal de Pablo  
Borja's channel  
Andoni crazy stuff

- Si se implementa la parte extra, el programa deberá mostrar algo similar a esto:

Deustube: audientia melior auro.

1. Mostrar direcciones web de todos los vídeos.
2. Buscar canal con más likes.
3. Mostrar la duración total de todos los vídeos.
4. Mostrar todos los canales que sigue un usuario.
5. Sistema de recomendación.

0. Salir

Opción: 5

Usuario: Pablo.

El usuario Pablo coincide en historial con el usuario Andoni, por lo que le recomendamos el siguiente vídeo:

<http://deustube.com/79DFB3vHJ>