

A dark blue vertical bar runs down the left side of the page. A blue arrow points to the right from the bar, containing the text '2017-2018'.

2017-2018

LSTUB

Pràctica 1- DPO

Several thin, curved lines in dark blue and light grey originate from the bottom left corner and sweep upwards and to the right.

Marc Llorc Maulion
Alex Almansa Casanova

marc.llort.2016
alex.almansa.2016

ÍNDIX

Resum Enunciat	2
Diagrama de Classes	3
Classes	4
Provatures del codi	6
Temps dedicat	7
Conclusions	8
Bibliografia	9

Resum Enunciat

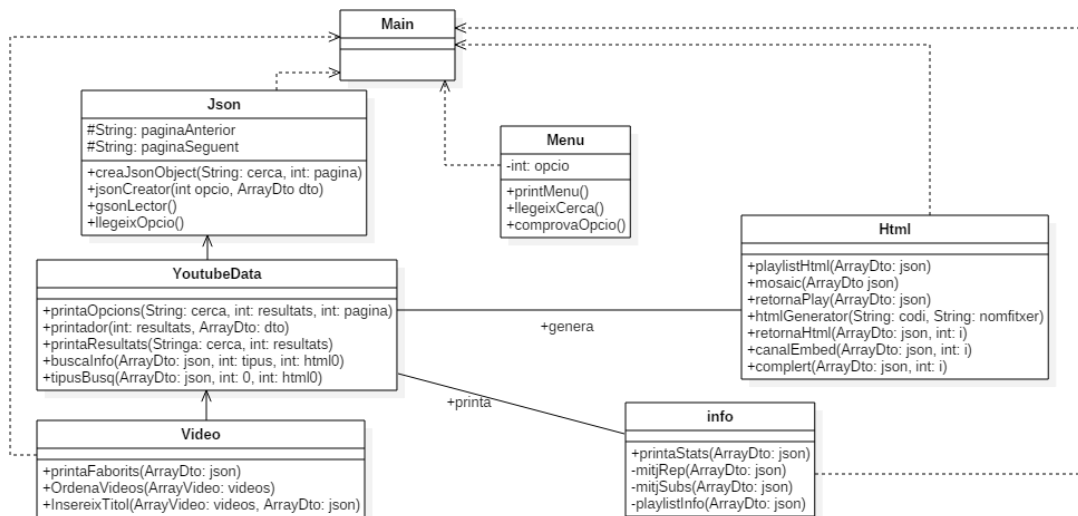
En la pràctica de LsTub, tenim l'encàrrec de fer a terme amb Java, i programat des de IntelliJ un programa capaç de, cercar busquedes a YouTube, desar a preferits aquells que nosaltres desitgem, classificar la busqueda per mostrar els vídeos amb més likes, mostrar les estadístiques dels vídeos preferits, crear un mosaic també dels vídeos preferits, i ja per últim mostrar les llistes de reproducció en un fitxer html.

Per obtenir les informacions dels vídeos, ens caldrà utilitzar el webservice (API) de Google, el qual ens permetrà accedir a aquelles informacions del vídeo que necessitem.

És necessari abans de començar a programar crear un diagrama de classes per organitzar com haurem de escriure i organitzar el codi.



Diagrama de Classes



Alhora de realitzar la pràctica, vam començar sobre la idea de tenir una classe que juntes vídeos, playlist i channels, perquè ens semblava la forma més lògica de fer-ho, però durant la programació vam decidir canviar i ajuntar els tres tipus ja que molts atributs eren els mateixos, i compartien les mateixes funcions.

YoutubeData és la nostra classe central on ajuntem totes les diferents funcions de les seves subclasses per poder dur a terme les diferents opcions de la practica (Busqueda, afegir a preferits...).

Alhora, la classe deriva de Json, que és on tenim totes les funcions necessàries per generar el json, llegir-lo i tractar-lo.

Classes

JSON:

Tractem tot allò relacionat amb els json, tant com de llegir-los com de escriure'ls. No tenim cap atribut, però les funcions més destacades són:

- `ArrayDto creaJsonObject(java.lang.String cerca, int pagina)`
Funció que s'encarrega de realitzar la cerca i ens retorna el `JsonObject` amb els resultats de la mateixa.
- `public void jsonCreator(int opcio, ArrayDto dto)`
Funció que s'encarrega de crear el fitxer `favoritplaces.json` on hi haurà els vídeos, canals i playlists que guardem a favorits
- `public ArrayDto gsonLector()`
Funció que s'encarrega de llegir el json que tenim actualment la carpeta

YoutubeData:

En aquest cas ens encarreguem de tractar la informació subministrada per el JSON.

- `public void printaOpcions(java.lang.String cerca, int resultats, int pagina)`
Funció que s'encarrega de que en cas que calgui, es passi de pagina i si es vol guardar un vídeo, actualitzar el json.
- `public void printador(int resultats, ArrayDto dto)`
Funció que s'encarrega de printar els resultats de la cerca.
- `public void printarResultats(java.lang.String cerca, int resultats)`
Funció que s'encarrega de cridar les funcions de crear el json object i després de printar els resultats.
- `protected ArrayVideo buscaInfo(ArrayDto json, int tipus, int html0)`
Funció que s'encarrega de classificar els elements de json en un array de en vídeos, canals o playlists.

ArrayDto i ArrayVideo:

Ambdues classes, estan formades per un conjunt de classes també, cadascuna conte les diferents variables que ens interessin agafar del json, per poder dur a terme les tasques necessàries. Tots tenen els seus getters i setters.

Cal destacar que a `publishedDate` hem utilitzat el tipus `date` per poder comparar fàcilment les dates.

Vídeo:

Classe que gestiona tota la informació relacionada amb els vídeos.

- `public void printaFavorits(ArrayDto json)`
Funció que s'encarrega de printar els vídeos favorits ordenats per la relació de likes/dislikes.

HTML:

En aquesta classe tenim totes les funcions relacionades amb la opció 5 i 6, que són la generació de fitxers html amb diferents funcions. Les funcions destacades són:

- `public static void playlistHtml(ArrayDto json)`
S'encarrega de generar l'arxiu html de les playlists que hi ha a favoritePlaces.json.
- `public static void mosaic(ArrayDto json)`
Funció que s'encarrega de generar un mosaic amb tots els vídeos, channels i playlists que tenim a favorits.

Menú:

Classe que s'encarrega de gestionar el menú printar-lo i llegir i comprovar la opció que entra l'usuari.

- `public void printMenu()`
S'encarrega de printar el menú.
- `public java.lang.String llegeixCerca()`
Llegeix la cerca i la prepara per introduir-la a la url.
- `public void comprovaOpcio()`
Comprova que la opció que entra l'usuari és correcta, si no ho és la torna a demanar.

Provatures del codi

Al llarg de la realització de la pràctica hem anat fent proves de com funcionava el nostre codi, cada cop que acabàvem de generar una opció, o una funció important de la pràctica, com podria ser la de creació del JSON, o la de a partir del link, obtenir el json d'internet.

Un cop fetes les provatures, per arreglar els errors que anàvem tenint, si eren de codi (no de classes), apart de utilitzar els típics `System.out` per veure on el nostre codi fallava, ens ha sigut molt útil fer us del sistema de gestió d'errors `try catch`, ja que quan fallava i anava al `catch`, teníem la funció `printStackTrace`, que ens indicava en quina línia de quin fitxer estàvem tenint, la majoria de cops, un `NullPointerException` o un `indexOutOfBounds`.

En el codi no hem pogut posar les inicials de les classes en majúscules ja que al fer el refactori que ens canvia totes les lletres a tots els fitxers, ens ha donat molts errors, i tot dedicar-hi molta estona no hem pogut arreglar-ho, i al no tenir més temps hem tingut que deixar-ho.

Temps dedicat

Per començar, si tenim en compte l'exercici d'avaluació continua on havíem de llegir el json, podríem dir que només per entendre com funciona el fitxer, la seva lectura i totes les seves característiques vam estar unes 20 hores.

Tenint compte que no hem tingut gaire temps per realitzar la pràctica, degut a tenir moltes més practiques, i la entrega ser just abans de Nadal, hem dedicat aproximadament unes 40h o més.

Dins aquestes 40 hores podríem dir que al principi, no vam dedicar-ne prou al disseny i organització de la pràctica, potser unes 2 hores inicialment que van fer que després de tenir una bona part del codi ja feta, haguéssim d'invertir bastant temps en reorganitzar-lo en classes que tinguessin sentit ja que en un principi teníem classes desordenades, que no estaven ben relacionades i que no tenien sentit.

Conclusions

Durant la realització de la pràctica hem après molt sobre el funcionament de les apis que diferents serveis, en aquest cas YouTube, ofereixen als desenvolupadors, per accedir a informacions exactes que poden ser necessàries per els desenvolupament de qualsevol programa.

Des de un primer moment ja ens vam trobar amb dificultats per aconseguir el json de la api de YouTube, però un cop aconseguit la realització de la primera part de la practica (el tractament del json) no va ser gaire complicat, ja que tots els problemes ja els vam tenir alhora de realitzar el exercici de continua, per tant ja sabíem com calia enfocar el tractament del json.

La part on més problemes hem tingut, ha sigut alhora de crear el diagrama de classes, ja que tot hi que en un principi vam crear-nos un model el qual volíem seguir, l'aparició de més classes i variables ens va complicar molt el poder seguir-lo, per tant a la meitat de la realització de la pràctica ens vam adonar de que teníem tot molt desorganitzat i ens donava molts problemes alhora de continuar programant, per tant vam tenir que reorganitzar tot el codi, però aquest cop millor, ja que coneixíem la majoria de classes i variables que ens feien falta.

Per altre banda, podem dir que hem après a treballar amb apis externes que pot ser molt útil i necessari per a programar tot tipus d'aplicacions. A més ens hem familiaritzat amb el intelij, que serà el programa que utilitzarem la resta del curs per a programar.

Bibliografia

Stack Overflow - Where Developers Learn, Share, & Build Careers. Stack Overflow - Where Developers Learn, Share, & Build Careers [online]. [Utilitzada Durant tota la pràctica].

Disponible a: <https://stackoverflow.com/>

YouTube Data API | Google Developers. Google [online]. [Utilitzada Durant tota la pràctica]. Disponible a: <https://developers.google.com/youtube/v3/>

Document Auxiliar – LaSalle [online]. [Utilitzada Durant tota la realització html (15/12/17)].

Disponible a: <http://estudy.salle.url.edu/>