

## SEGONA PRÀCTICA DE PROGRAMACIÓ

### Anàlisi de la quantitat d'aigua als embassaments

#### Introducció al problema

En la plana web Portal de Dades Obertes de la Generalitat hi ha, entre d'altres, la informació de la quantitat d'aigua als embassaments de les Conques Internes de Catalunya. Està disponible a: <https://analisi.transparenciacatalunya.cat/Medi-Ambient/Quantitat-d-aigua-als-embassaments-de-les-Conques-/gn9e-3qhr> De les dades que conté, hem preparat una versió per a treballar en la pràctica. El fitxer que conté aquesta versió el teniu disponible dins el projecte de partida de la pràctica, junt amb aquest enunciat a l'espai moodle (*data descàrrega dades originals: 25/10/2023*).

El contingut del fitxer és el que es mostra a continuació:

Dia	Nom embassament (Població)	Província	Nivell absolut (msnm)	Percentatge volum embassat (%)	Volum embassat (hm3)
24/10/2023	Embassament de Sau (Vilanova de Sau)	Barcelona	388.97	17.0	28.15
24/10/2023	Embassament de la Baells (Cercs)	Barcelona	596.19	25.3	27.69
24/10/2023	Embassament de Foix (Castellet i la Gornal)	Barcelona	97.38	54.5	2.04

La primera línia és la capçalera i conté el nom dels diferents camps. La resta de línies tenen els valors dels diferents camps separats pel caràcter , (coma). El contingut de les dades de la primera línia després de la capçalera és:

24/10/2023,Embassament de Sau (Vilanova de Sau),Barcelona,388.97,17.0,28.15

Nota a tenir en compte, els diferents camps estan separats per la coma excepte el nom de la població que està junt amb el nom de l'embassament i es distingeix perquè està entre parèntesis.

Volem dissenyar una aplicació que permeti fer consultes sobre aquestes dades.

#### Especificacions del programa

Us proporcionem la classe de dades per a guardar la informació d'una línia del fitxer, l'hem anomenat `NivellAigua`. No heu de modificar el codi d'aquesta classe. Però sí que hi heu d'afegir els comentaris JavaDoc per a que estiguin també junt amb els que posareu a la resta de classes.

Us demanem que, per tal de guardar les dades de les diferents mesures, dissenyeu i implementeu una nova classe que emmagatzemarà una llista d'instàncies de la classe `NivellAigua`. La classe llista s'ha d'implementar seguint l'estructura i els procediments explicats a classe, és a dir, no es poden fer servir estructures de dades predefinides de Java.

Aquesta nova classe l'anomenarem `LlistaNivellsAigua`. Ha de tenir tots els mètodes bàsics necessaris per manejar una llista (constructor, get del número d'elements de la llista, consultar la mesura d'una certa posició, copia, toString), així com els mètodes que permetin les següents accions:

1. Afegir les dades d'una nova mesura al final de la llista.
2. Retornar la instància de la primera mesura que trobem a la llista d'un cert embassament. El nom de l'embassament es rep per paràmetre. Si no tenim dades de cap mesura de l'embassament, el mètode retornarà *null*.
3. Retornar un duplicat de la instància de la mesura que té un percentatge de volum més alt. En cas d'empat es pot retornar qualsevol de les que compleix la propietat.
4. Fer el mateix pel percentatge de volum més baix de totes les mesures que tenim a la llista.
5. Retornar una taula de String amb els noms dels embassaments dels què tenim dades.
6. Retornar un duplicat de la instància de la mesura que té el volum d'aigua més alt. En cas d'empat es pot retornar qualsevol de les que compleix la propietat.
7. Consultar les dades de mesura dels embassaments que estan ubicats en una província. El nom de la província es rep per paràmetre. Es retorna les dades de les mesures que compleixen la condició en una nova **LlistaNivellsAigua**.
8. Consultar les dades de mesura en un període de temps. Es rep per paràmetre la franja de dates. Es retorna les dades de les mesures que compleixen la condició en una nova **LlistaNivellsAigua**.
9. Eliminar el conjunt de mesures que són d'un embassament (recorda que no poden quedar posicions buides a la taula que conté la llista). El mètode rep per paràmetre el nom de l'embassament.

A més, hem de construir un **programa principal**. Et proporcionem una primera versió d'aquest programa principal amb un mètode que et permet llegir un nombre determinat de línies de l'arxiu que et donem. L'arxiu és el `csv1` que està inclòs en el projecte `Practica2_pdp`. Cal usar aquest fitxer i aquest mètode per fer el joc de proves. El mètode que llegeix un nombre determinat de línies de l'arxiu és el següent:

```
String[] llegirLiniesFitxer(int numLinies);
```

El mètode rep un paràmetre que indica quantes línies volem llegir de l'arxiu i retorna un vector d'Strings amb tantes posicions com línies llegides<sup>2</sup>. Cada posició del vector correspon a una línia de l'arxiu i el contingut de l'String correspon als diferents camps d'informació d'una mesura del nivell de l'aigua en un embassament, en la forma que s'ha indicat a l'inici de l'enunciat. La primera línia del fitxer, que no es retorna, és la capçalera amb la descripció del contingut.

L'aplicació demana, en primer lloc, el número de línies que es volen carregar del fitxer de text i crida al mètode per carregar la taula amb els Strings de les línies. A continuació, has d'afegir el codi per processar i separar la informació de cada una de les línies llegides, extreure els valors dels diferents atributs que es necessiten per crear una instància de la classe `NivellAigua` i afegir aquesta instància a

---

<sup>1</sup> Els fitxers `csv` són fitxers de text on la informació està separada per comes (o punt i coma) i per salts de línia.

<sup>2</sup> El valor de `numLinies` ha de ser vàlid i no pot ser superior al número de línies real del fitxer de text que utilitzeu.

la llista. Per tant, **les línies del fitxer s'han de fer servir per inicialitzar la llista d'instàncies que farà servir el programa principal**.

Un cop processades les línies i guardades les dades a la llista, el *main* ha de mostrar **un menú** que permeti a l'usuari interactuar amb el programa i, en particular, ha d'oferir les següents funcionalitats usant els mètodes implementats a la classe *llista* (tota la informació necessària s'introduirà per teclat):

1. Mostrar el conjunt de mesures de la llista.
2. Mostrar el conjunt de mesures d'una província entre una franja de dates. El nom de la província i els valors de les dates s'indiquen per teclat.
3. Consultar les dades de la primera mesura que hi ha a la llista d'un cert embassament. El nom de l'embassament es llegirà de teclat.
4. Consultar les dades de la mesura que té un percentatge de volum més alt dels embassaments de cada una de les províncies, és a dir, calcular les dades per als embassaments de Tarragona, Barcelona, Lleida i Girona. Mostra la informació i comprova si són del mateix any.
5. Consultar les dades de la mesura que té un percentatge de volum més baix de totes les mesures que tenim a la llista. En aquest cas no separem per províncies.
6. Consultar i mostrar de quants embassaments tenim dades en una certa província. Mostra també els noms dels embassaments. El nom de la província s'indica per teclat.
7. Agafem les mesures d'un cert any. L'any s'indica per teclat. Calculem les dades de mesura d'aquell any en dues províncies també indicades per teclat. Volem calcular en quina de les dues províncies s'ha trobat aquell any amb un volum d'aigua més alt.
8. Consultar les dades de mesura en un període de temps. Els valors de dates s'indiquen per teclat.
9. Eliminar el conjunt de mesures dels embassaments d'una província. El nom de la província es llegirà de teclat. Per comprovar el bon funcionament de l'operació, mostrarem en primer lloc les dades de mesura dels embassaments de la província, cridarem a l'operació eliminar i després tornarem a mostrar les dades de mesura dels embassaments de la província per comprovar que ja no n'hi ha cap.
10. Sortir del programa.

## Instruccions per la fase de validació de l'exercici

Un codi és correcte quan no té cap error de sintaxi i, **sobretot, funciona segons els requeriments de l'enunciat**. Validar un programa és comprovar que funciona bé en tots els casos. Els jocs de proves són el recull de tots aquests casos que s'han de comprovar.

Vosaltres heu de fer validació de totes les operacions en les diferents situacions que es poden donar. Per a fer una comprovació parcial hem generat un qüestionari a Moodle on us apareixeran algunes preguntes aleatòries que haureu de provar i donar els resultats que obteniu. La pràctica és en grup però el qüestionari, per qüestions tècniques de moodle, és individual, així cada estudiant ha de respondre les preguntes que li sortiran.

L'ús del debug és una eina imprescindible en la fase de validació del codi. Per assegurar el seu coneixement, en algunes de les preguntes del qüestionari de validació us demanarà afegir una captura de pantalla del debugger en un moment concret de l'operació. La captura s'afegirà en la pregunta 2 del qüestionari, pregunta preparada per a pujar fitxers.

## Instruccions generals pel desenvolupament de l'exercici

- **Data límit pel lliurament de l'exercici: 20/11/2023.**
- Les pràctiques s'han de fer en grups de **dues persones**. Per apuntar-vos a un grup teniu la consulta a l'espai de l'assignatura "**Apunta't a un grup per fer la pràctica 2**".
- Treballs iguals o molt similars entre diferents grups tindran una nota de 0.
- Heu d'entregar el projecte Java en un fitxer comprimit en format zip anomenat: nom1Cognom1\_nom2Cognom2\_pract2.zip.
- L'arxiu comprimit s'ha de carregar a la tasca Moodle corresponent. Per a l'entrega, només un dels dos alumnes cal que publiqui la pràctica, si esteu apuntats al grup ja apareixerà als dos membres.
- El qüestionari d'avaluació dels jocs de proves l'ha de fer cada alumne per separat.
- No s'acceptaran programes amb errors de compilació. Eviteu també els warnings.

## Guia d'avaluació

- Correctesa en la definició dels atributs (que siguin del tipus adequat).
- Correctesa de la implementació dels mètodes (que rebin els paràmetres que s'indica i facin correctament el que es demana, de la forma més senzilla i òptima possible). Usar correctament els esquemes de cerca i de recorregut.
- Manegament correcte de les llistes d'objectes, usant el model explicat a classe (afegir, eliminar, ...)
- El programa principal crea les instàncies necessàries i fa crides als mètodes passant adequadament els paràmetres requerits.
- El codi que s'implementi ha de ser clar: ha d'estar tabulat, ha d'estar comentat adequadament (cada mètode ha de tenir una capçalera indicant què fa, i quins paràmetres rep) i els noms que useu pels atributs, mètodes, ... han de ser representatius. Els mètodes han de ser estructurats, no hi pot haver cap mètode que sigui massa llarg.
- No s'acceptarà que les sortides per pantalla (System.out.println) es facin en qualsevol altre lloc que no sigui en la classe que conté el main. El mateix passa amb les entrades (lectures de teclat).
- Classe aplicació on hi ha el main preparada per permetre realitzar un joc de proves complet.
- Qüestionari de jocs de proves completat.
- Avaluació de l'ús de debug.
- Entrega del projecte comprimit de forma correcta.