Lab 2:

Basic:

Se defineste clasa Persoana: campuri-> Nume, Prenume, Varsta

Se defineste clasa Student extends Persoana: campuri-> An de studiu, Nume facultate

Se defineste clasa Profesor extends Persoana: campuri-> Titlu (conferentiar, profesor, etc), Vechime, Catedra

Sa se realizeze un program care citeste un fisier CSV de pe disc (cale hardcodata initial) si populeaza doi vectori, unul de studenti unul de profesori cu datele din CSV

Datele sunt serializate in felul urmator:

S-> student, Nume,Prenume,Varsta,AnStudiu,NumeFacultate (trebuie facut trim).

P-> Pofesor, Nume,Prenume,Varsta,Titlu,Vechime,Catedra

Sa se construiasca apoi o singura colectie de elemente Persoana si sa fie afisata la ecran (cu supraincarcarea lui toString()).

-> Extra:

Standard:

1. Orice obiect de tip Persoana sa aiba o metoda "readFromScanner” care sa stie sa citeasca dintr-un obiect scanner. Aceasta metoda sa fie definita in clasele copil.

2. Orice obiect de tip persoana sa aiba o metoda "writeToStream" care sa faca opusul lui read.

3. Pentru student sa se adauge o lista de note (camp List<Nota> unde Nota e o clasa noua cu campuri String disciplina, Int nota) si sa se introduca in mecanismul de serializare. Astfel, pentru student, dupa citirea celorlalte date, va exista un camp integer care reprezinta nr de note.

4. Sa se creeze un obiect de tip BazaDeDate care sa stocheze persoane, sa contina operatii Load,Export (from/to file)

Good:

5. Sa se adauge un camp ID la persoana, care NU este serializat.

6. Sa se adauge la obiectul BazaDeDate operatia AddPersoana(Persoana persoana) care sa functioneze indiferent de obiectul trimis

7. Sa se adauge la obiectul BazaDeDate operatia EditPersoana(int id, Persoana persoana) care sa modifice datele persoanei cu ID=id.

Better:

8. Sa se adauge o interfata grafica in care sa se afiseze un textfield cu calea catre fisierul bazei de date si doua butoane, Load,Save,Close. Initial campul este gol. Cand se introduce o cale si se apasa butonul Load, se incarca baza de date din acea locatie. Daca nu s-a putut face incarcarea, o eroare va fi afisata intr-un label aflat sub campul cale. Daca incarcarea s-a facut cu succes, campul cale devine inactiv, la fel si butonul Load. Save este activ doar daca o baza de date a fost incarcata. La apasarea butonului Save (cand e activ) se salveaza baza de date (si atat!). La apasarea butonului Close, se inchide baza de date curenta (se sterge din memorie si se reactiveaza campul cale si butonul Load. Datele despre persoanele din baza de date sunt prezentate in doua tablouri dupa cum se vede in imagine.

Excelent:

9. Modificati tablourile profesor si student pentru a contine TextField in loc de label. Permiteti editarea directa din GUI a obiectelor.

Wonderful:

10. Adaugati functii de sortare capetelor de tabela (asc si desc). Performanta nu e o problema faceti cum vreti.

Poza

https://docs.google.com/document/d/14b6KFjUTX5LaXiobdQHl70Azk8DhLQDc4v2zhqtZfWU/edit?usp=sharing

https://goo.gl/1RTiGr

Main

https://docs.google.com/document/d/1c5DAi1UGOjDJuT3qI9-4hkIvxTiYTbZiIuz69pgMOxY/edit?usp=sharing

https://goo.gl/YtjE6Z

Date1:

https://docs.google.com/spreadsheets/d/1jNvwQMkxqfpsv3kpU209krtGjuYo5kArl4DhNGUEmeY/edit?usp=sharing

https://goo.gl/cAfZcL

Date2:

https://docs.google.com/spreadsheets/d/1ImJej7WoiXfVNORdShI8GaMWGXOztQdGz8LzYKcvc6w/edit?usp=sharing

https://goo.gl/19ZhCW