

Uppgift 1: Personalregister

Beskrivning

Skapa ett program som hanterar information om anställda på ett företag.

Krav

- Skapa ArrayList för namn, löner och avdelningar
 - Implementera följande metoder:
 - `läggTillAnställd(ArrayList<String> namn, ArrayList<Double> löner, ArrayList<String> avdelningar, String nyttNamn, double lön, String avdelning)`
 - `visaAllaAnställda(ArrayList<String> namn, ArrayList<Double> löner, ArrayList<String> avdelningar)`
 - `sökAnställd(ArrayList<String> namn, String sökNamn)`
 - `genomsnittslön(ArrayList<Double> löner)`
 - `högstaLön(ArrayList<Double> löner)`
 - `antalPerAvdelning(ArrayList<String> avdelningar, String avdelning)`
 - Skapa en meny i main som låter användaren välja mellan funktionerna
 - Programmet ska fortsätta tills användaren väljer att avsluta
-

Uppgift 2: Recepthanterare

Beskrivning

Bygg ett system för att hantera matrecept och ingredienser.

Krav

- Skapa ArrayList för receptnamn, ingredienser och portioner
- Implementera följande metoder:
 - `läggTillRecept(ArrayList<String> receptnamn, ArrayList<String> ingredienser, ArrayList<Integer> portioner, String namn, String ingrediens, int portion)`
 - `visaAllaRecept(ArrayList<String> receptnamn, ArrayList<String> ingredienser, ArrayList<Integer> portioner)`
 - `sökRecept(ArrayList<String> receptnamn, String sökterm)`
 - `receptMedIngrediens(ArrayList<String> receptnamn, ArrayList<String> ingredienser, String ingrediens)`

- `skalaRecept(ArrayList<Integer> portioner, int receptIndex, int nyaPortioner)`
 - `antalRecept(ArrayList<String> receptnamn)`
 - Användaren ska kunna söka recept baserat på ingredienser
 - Möjlighet att skala upp/ner antal portioner för ett recept
-

Uppgift 3: Filmbibliotek

Beskrivning

Skapa ett program för att hantera en samling filmer med betyg och genrer.

Krav

- Skapa ArrayList för filmtitlar, genrer, år och betyg
 - Implementera följande metoder:
 - `läggTillFilm(ArrayList<String> titlar, ArrayList<String> genrer, ArrayList<Integer> år, ArrayList<Double> betyg, String titel, String genre, int årtal, double filmbetyg)`
 - `visaAllaFilmer(ArrayList<String> titlar, ArrayList<String> genrer, ArrayList<Integer> år, ArrayList<Double> betyg)`
 - `filtreraGenre(ArrayList<String> titlar, ArrayList<String> genrer, String valdGenre)`
 - `bästaFilmer(ArrayList<String> titlar, ArrayList<Double> betyg, double minBetyg)`
 - `genomsnittsbetyg(ArrayList<Double> betyg)`
 - `filmFrånÅr(ArrayList<String> titlar, ArrayList<Integer> år, int årtal)`
 - Användaren ska kunna filtrera filmer baserat på genre
 - Visa endast filmer över ett visst betyg
 - Sök filmer från specifika år
-

Uppgift 4: Träningslogg

Beskrivning

Utveckla ett program för att logga och analysera träningspass.

Krav

- Skapa ArrayList för träningstyp, duration, datum och kalorier

- Implementera följande metoder:
 - `loggaTräning(ArrayList<String> träningstyp, ArrayList<Integer> duration, ArrayList<String> datum, ArrayList<Integer> kalorier, String typ, int minuter, String träningsdatum, int förbrändaKalorier)`
 - `visaAllaTräningar(ArrayList<String> träningstyp, ArrayList<Integer> duration, ArrayList<String> datum, ArrayList<Integer> kalorier)`
 - `totalTräningstid(ArrayList<Integer> duration)`
 - `totalKalorier(ArrayList<Integer> kalorier)`
 - `populärTraining(ArrayList<String> träningstyp)`
 - `träningPerDag(ArrayList<String> datum, String valdDatum)`
 - Beräkna total träningstid för veckan/månaden
 - Visa vilken träningstyp som utförs mest
 - Räkna ut genomsnittlig kaloriförbränning per pass
-

Uppgift 5: Bokhandel

Beskrivning

Skapa ett kassasystem för en bokhandel med lager och försäljning.

Krav

- Skapa ArrayList för boktitlar, priser, lagerantal och försäljning
- Implementera följande metoder:
 - `läggTillBok(ArrayList<String> titlar, ArrayList<Double> priser, ArrayList<Integer> lager, ArrayList<Integer> försäljning, String titel, double pris, int antalILager)`
 - `visaLager(ArrayList<String> titlar, ArrayList<Double> priser, ArrayList<Integer> lager)`
 - `sälBok(ArrayList<String> titlar, ArrayList<Integer> lager, ArrayList<Integer> försäljning, String titel, int antal)`
 - `fyllPåLager(ArrayList<String> titlar, ArrayList<Integer> lager, String titel, int antal)`
 - `totalFörsäljningsvärde(ArrayList<Double> priser, ArrayList<Integer> försäljning)`
 - `mestsåldaBöcker(ArrayList<String> titlar, ArrayList<Integer> försäljning)`
- Kontrollera att det finns tillräckligt med böcker i lager innan försäljning
- Beräkna total försäljningssumma

- Lista de mest populära böckerna baserat på försäljningsantal
- Varna när lagret blir lågt (under 5 exemplar)