

Exerciții

IMPORTANT

- Folosiți OOP pentru a rezolva problemele
- Folosiți enumerații, proprietăți, liste
- Folosiți <https://app.diagrams.net/> pentru a realiza diagramele UML
 - Proiectați diagramele UML pe cât posibil ÎNAINTEA redactării programului.

Spor la treaba!

Exercitii

Scrieti un program care va modela conturi bancare. Un cont bancar poate fi cont de economii, cont de investitii sau cont curent si va fi caracterizat de un nume.

- In toate conturile vom putea depune bani.
 - La depunerea banilor soldul curent va crește cu valoarea depusa

Din toate conturile vom putea extrage bani

- În situatia in care suma ceruta depășește soldul curent, pe ecran va fi afișat un mesaj corespunzator iar suma nu va fi extrasa.
- Contul curent
 - La extragerea numerarului dintr-un cont curent, in cazul in care suma extrasa este mai mare decat cea disponibila in cont, contul va permite descoperirea de cont in limita plafonului de 5000 ron.
 - Daca suma ceruta depaseste plafonul descoperirii de cont, atunci pe ecran va fi afișat un mesaj corespunzator iar suma nu va fi extrasa.
- Contul de economii
 - Este caracterizat de o rata a dobanzii.
 - Dupa fiecare depunere, soldul contului de economii va fi reactualizat tinandu-se cont de rata dobanzii :
 - $\text{ValoareSold} = \text{ValoareSold} * (1 + \text{rataDobanzii})$

Exercitii

- Contul de investiții este un cont bancar care se comportă precum un cont de economii, cu mențiunea ca extragerea banilor este posibilă abia după atingerea termenului de extragere
 - La deschiderea unui cont de investiții, ziua termenului de extragere va fi specificat sub forma unui întreg (1-31)
 - Dacă se încearcă extragerea banilor înainte de termen, contul va afișa un mesaj corespunzător iar suma nu va fi extrasă din cont.
 - Ziua curentă poate fi determinată folosind: `DateTime.Now.Day`
- Toate conturile vor pune la dispoziție descrierea acestora. Descrierea va conține toate detaliile relevante ale contului (nume, tip, sold curent, rată dobânzii, ziua termenului de extragere, etc)
- Suplimentar
 - Conturile vor avea câte un id unic de tip Guid (folositi google). Id-ul va fi atribuit automat fiecarui cont la creare și va fi expus în descrierea contului.
- Implementați clasele, instanțiați obiecte și apelați-le metodele.

Exercitii – partea a 2-a

- Clasa Cont
 - va detine o metoda prin care va fi calculat numarul total al tuturor conturilor create
- Apelati metodele si afisati-le rezultatul