

Fiecare student va lucra la un proiect individual. Proiectul este structurat în mai multe etape. Condiția de punctare a proiectelor este aceea ca acestea să nu prezinte erori de compilare și să implementeze cerințele date.

Terme de predare:

- Etapa I: 31 martie
- Etapa II: 28 aprilie
- Etapa III: 26 mai

Pentru orice informație suplimentară sau neclarități: marius.scarlat@endava.com

Etapa I:

1. Definirea sistemului:

- a. sa se creeze o lista pe baza temei alese cu cel puțin 10 acțiuni/interogări care se pot face în cadrul sistemului și o lista cu cel puțin 8 tipuri de obiecte.

2. Implementare:

Sa se implementeze în limbajul Java o aplicație pe baza celor definite la punctul 1.

Aplicația va conține:

- clase simple cu attribute private / protected și metode de acces
- cel puțin 2 colecții diferite capabile să gestioneze obiectele definite anterior (List, Set, Map, etc.) dintre care cel puțin una sa fie sortata – se vor folosi array-uri uni-/bidimensionale in cazul in care nu se parcurg colectiile pana la data checkpoint-ului.
- utilizare moștenire pentru crearea de clase adiționale și utilizarea lor în cadrul colecțiilor;
- cel puțin o clasa serviciu care sa expună operațiile
- o clasa main din care sunt făcute apeluri către servicii

Etapa II:

1. Extindeți proiectul din prima etapa prin realizarea persistenței utilizând fișiere.

-Se vor realiza fișiere de tip csv ¹(comma separated values) pentru cel puțin 4 dintre clasele definite in prima etapa.

- se vor realiza servicii singleton generice pentru scrierea și citirea din fișiere
- la pornirea programului se vor încărca datele din fișiere utilizând serviciile create

2. Realizarea unui serviciu de audit

- se va realiza un serviciu care sa scrie într-un fișier de tip CSV de fiecare data când este executata una dintre acțiunile descrise in prima etapa. Structura fișierului:
nume_actiune, timestamp

Etapa III:

Înlocuiți serviciile realizate în etapa a II-a cu servicii care sa asigure persistenta utilizând baza de date folosind JDBC.

- sa se realizeze servicii care sa expună operații de tip create, read, update, delete pentru cel puțin 4 dintre clasele definite

¹ Fiecare coloana din fisier este separata de virgula. Exemplu: nume,prenume,varsta

Teme sugerate:

- 1) catalog (student, materie, profesor)
- 2) biblioteca (sectiuni, carti, autori, cititori)
- 3) programare cabinet medical (client, medic, programare)
- 4) gestiune stocuri magazin (categorii, produse, distribuitori)
- 5) aplicatie bancara (conturi, extras de cont, tranzactii, carduri, servicii)
- 6) platfora e-learning(cursuri, utilizatori, cursanti, quizuri)
- 7) sistem licitatii (licitatii, bids, produse, utilizatori)
- 8) platforma food delivery(localuri, comenzi, soferi, useri)
- 9) platforma imprumuturi carti - tip bookster (companii afiliate, utilizatori, carti)
- 10) platforma e-ticketing (evenimente, locatii, clienti)