

Tema POO:

Grad de dificultatea: mediu- multe detalii la fiecare clasa

Timp de lucru: aproximativ 11 zile, in ultimele zile 24/7

-pentru primele clase doar le am declarat le-am declarat conform cerintei, am realizat constructorii si am implementat metodele cerute

-Pentru clasa Course am declarat mai intai variabilele si am realizat si aici constructorii pentru ca ulterior sa implementez metodele: addAssistant -> pentru a adauga un asistent am luat ID ul grupei si l-am introdus in dictionar pe pozitia corespunzatoare, acelasi lucru si pentru addStudent si addGroup; pentru getGrade am parcurs lista de note, iar in momentul in care am gasit studentul pe care il cautam i-am returnat nota; getAllStudents creeaza un ArrayList de studenti in care introduce fiecare student parcurgand fiecare grupa existenta la fiecare curs, iar getAllStudentsGrade implementeaza un principiu asemanator. Pentru ca Course implementeaza sablonul

builder am creat mai jos clasa interna CourseBuilder si am implementat o urmarind sablonul, aceasta urmand pe viitor sa ne ajute la crearea cursurilor in timpul testarii

-Clasele FullCourse si PartialCourse se implementeaza asemanator cu clasa Course, in plus avnd metoda promovare pe care am implementat-o urmarind criteriile de promovare cerute in enunt (partial >= si examen >= 2 sau total >= 5)

-pentru observe am decalcat parintele sa implementeze observer, iar Catalog sa implementeze Subject, am implementat cele 2 sabloane, am creat o lista de observare si una de notificari, iar in metoda notifyObserver

parcurng toti observatorii din lista (care sunt parintii adaugati anterior in clasa de test), iar daca parintele exista il notific despre existenta unei note

-am implementat clasa ScoreVisitor urmarind sablonul, declarand un Tuple generic si implementand metoda visit atat pentru asistent cat si pentru profesor: pentru asistent am parcurs cursurile si pentru fiecare curs am luat toti asistentii de la acel curs si am verificat daca vreunul este identic cu parametrul dat metodei. Pentru un astfel de asistent gasit, iteram printre tuplurile din partialScore si fiecarui student ii adaugam nota la partial. Pentru teacher am luat la fel fiecare curs la rand, iar in momentul in care gaseam exact cursul profesorului titular pe care il cautam, iteram prin tuplurile

din examScore, cream o nota noua pentru fiecare student gasit, ii setam nota in examen si apoi adaugam

nota in catalog. Pentru fiecare student ii notificam si parintii despre primirea unei noi note, daca acestia exista.

-pentru testarea aplicatiei eu am ales introducea de mana a tuturor valorilor din Json, iar apoi adaugarea cores[unzatoare a cursurilor, studentilor, profesorilor, grupelor si asistentilor si am afisat o paret din functiile importante, inclusiv apelarea strategiilor pentru a afla cel mai bun student de la un anumit curs

-La StudentPage am in ceput prin a imi crea spatiul de lucru cu un panou si un frame care imi dau posibilitatea

sa plasez unde vreau diverse label-uri, am schimbat imaginea de start a aplicatiei din coltul stanga sus pentru a se incadra in tema interfetei, iar apoi am afisat studentul iar pentru a-i afisa cursurile la care este inscris am interat printre acestea si am verificat daca il gasim printre studenti, am realizat butonul ce ofera informatii suplimentare si contine, numele profesorului, asistentii de la curs, asistentul elevului (am parcurs grupele si daca respectiva grupa in continea pe student afisam asistentul acelei grupe), nota din examenul parial si final(am iterat prin lista de grades, iar la gasirea studentului pe care il cautam

i-am afisat notele). Pentru fiecare buton, panou, frame sau label adaugat am ales o culoare de fundal, o culoare

a textului si o dimensiune corespunzatoare.

-la TeacherPage parcurg fiecare curs pana gasesc cursul la care predă profesorul respectiv, apoi pein apasarea

butonului urmeaza sa se apeleze visit-ul pentru a adauga notele in catalog si le afisam ulterior intr un alt panou, prin parcurgerea listei de grades

-la AssistentPage in mare parte am facut acelasi lucru doar ca pentru fiecare curs am parcurs toti asistentii

pana l-am gasit pe cel pe care il cautam, am apelat visit pentru asistent si am introdus notele intr-un panou

la fel ca la teacher

-pentru Log in, creez 2 TextFields in care urmeaza sa se introduca username-ul si parola pe care prin apasarea

butonului Submit se vor compara cu argumentele date ca parametru, iar daca acestea sunt identice se deschide

fereastra corespunzatoare tipului de user