Contents

[1. Agentie de turism 1](#_Toc66302977)

[2. Gestiune magazine 2](#_Toc66302978)

[3. Gestiune biblioteca 3](#_Toc66302979)

[4. Gestiune grupa studenti 4](#_Toc66302980)

[5. Gestiune flota transport 5](#_Toc66302981)

[6. Gestiune echipa sportiva 6](#_Toc66302982)

[7. Cinematograf 7](#_Toc66302983)

[8. Cabinet medic familie 9](#_Toc66302984)

# Agentie de turism

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CNP | Nume | Prenume | Adresa | Data\_nasterii | Premium |
| Numeric | Char | Char | Char | Data | logical |

Persoana

PK

Factura

|  |  |  |  |
| --- | --- | --- | --- |
| CNP | Cod\_destinatie | Perioada | Pret\_sejur |
| numeric | Char | Char (’10.06.2020 – 19.06.2020’) | numeric |

FK FK

Destinatie

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Cod\_destinatie | Locatie | Transport | Pret\_noapte | Pret\_transport |
| Char | Char | char | Numeric | numeric |

PK

**Cerinte:**

1. Pentru tabela *Factura* campul pret sejur se calculeaza in modul urmator:
   * Se extrage din campul perioada, numarul de zile si se inmulteste cu pretul pe noapte, la acesta se adauga pretul transportului, ambele informatii fiind valabile in tabela *Destinatie*
2. Se proiecteaza o intefata grafica, prin care utilizatorului i se va cere sa introduca un CNP. La apasarea butonului “afiseaza informatii client”, pe baza CNP-ului citit se va afisa fie lista tuturor facturilor incluzand urmatoarele informatii:
   * Nume, prenume, cnp, premium, locatie, perioada, transport, pret sejur
   * Fie un mesaj corespunzator daca persoana nu a fost gasita in baza de date
3. Se proiecteaza o intefata grafica, prin care utilizatorului i se va cere sa introduca un cod de destinatie. La apasarea butonului “afiseaza informatii destinatie”, pe baza codulului de destinatie citit se va afisa fie lista tuturor facturilor vandute pentru aceasta destinatie cat si urmatoarele informatii:
   * Cod destinatie locatie, perioada, transport, pret sejur
   * Fie un mesaj corespunzator daca acest cod nu a fost gasit in baza de date
   * Se va afisa un clasament al locatiilor, in ordine descrescatoare, bazat pe pretul sejurului. Daca in tabela de facturi exista mai multe destinatii cu acelasi cod, se insumeaza pretul sejurului pentru a obtine acuratetea clasamentului. - ”) - pentru usurinta, acest punct se implementeaza in SQL
4. Pentru fiecare destinatie din tabela se vor numara si afisa cate persoane au cumparat bilete si se vor afisa si numarul de zile petrecute de fiecare persoana in aceasta locatie
   * Exemplu de afisare:
   * Cod\_destinatie locatie nr\_persoane
     + Persoana cnp a petrecut x zile
5. Pentru tabela de *persoane*, campul premium este completat pe baza urmatorului algoritm:
   * Se parcuge tabela facture: se insumeaza pentru fiecare CNP pretul sejurului, daca acesta depasete 3000ron atunci el devine un client premium (default campul este trecut pe “False”) - pentru usurinta, acest punct se implementeaza in SQL

# Gestiune magazine

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CNP | Nume | Prenume | Adresa | Data\_nasterii | Tip\_client |
| Numeric | Char | Char | Char | Data | char |

Persoana

PK

Factura

|  |  |  |  |
| --- | --- | --- | --- |
| CNP | Cod\_produs | cantitate | Suma\_de\_plata |
| numeric | Char | numeric | numeric |

FK FK

produse

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Cod\_produs | denumire | categorie | Pret | Data\_expirare |
| Char | Char | char | Numeric | data |

PK

**Cerinte:**

1. camp “tip client”: clientul poate sa fie unul din urmatoarele: platinum, gold, regular. Actualizarea acestui camp se face in functie de suma totala calculata pentru fiecare cnp, folosind datele din tabela factura. Daca suma totala depasete 4000ron este platinum, daca este intre 2000 si 3999.99 este gold, iar restul intra la regular.
   1. Campul suma\_de\_plata se calculeaza: pret produs \* nr de produse vandute
2. Se proiecteaza o intefata grafica, prin care utilizatorului i se va cere sa introduca un CNP. La apasarea butonului “afiseaza informatii client”, pe baza CNP-ului citit se va afisa fie lista tuturor facturilor incluzand urmatoarele informatii:
   1. Nume, prenume, cnp, tip client, denumire produse, categorie, transport, suma de plata
   2. Fie un mesaj corespunzator daca persoana nu a fost gasita in baza de date
3. Pentru toate produsele din tabela de produse, a caror data de expirare este in urmatoarele 10zile se va aplica un discount de 15% din pret, iar pentru cele care vor expira in urmatoarele 3zile, un discount de 50%. Toate produsele a caror data de expirare a trecut se vor marca pentru stergere. De asemenea se vor afisa doar aceste produse expirate cat si suma totala pierduta de catre magazine prin nevanzarea lor.
4. Pentru fiecare categorie de produse se afisa un top cu cele mai multe produse vandute.

Afisare in urmatorul mod:

Categorie produs numar total de produse vandute - acest punct va fi rezolvat in exclusivitate in SQL

1. Se va realiza un top descrescator al clientilor in functie de suma totala platita pe toate facturile
   1. Afisare sub forma
      1. Nume client cnp suma totala platita - rezolvarea subiectului se va face in SQL

# Gestiune biblioteca

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Cod\_Cititor | Nume | Prenume | Adresa | Data\_nasterii | student |
| Numeric | Char | Char | Char | Data | logical |

Persoana

PK

fisa

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Cod\_cititor | Cod\_carte | Data\_imprumut | Data\_retur | penalizari |
| numeric | Char | data | data | numeric |

FK FK

carte

|  |  |  |  |
| --- | --- | --- | --- |
| Cod\_carte | titlu | categorie | autor |
| Char | Char | char | Numeric |

PK

1. Se proiecteaza o intefata grafica, prin care utilizatorului i se va cere sa introduca un cod\_cititor. La apasarea butonului “afiseaza informatii client”, pe baza codui citit se va afisa fie lista tuturor cartilor imprumutate incluzand urmatoarele informatii:
   * Nume, prenume, cod\_Cititor, titlu, autor, data\_imprumut
   * Fie un mesaj corespunzator daca persoana nu a fost gasita in baza de date
2. Pentru a calcula si afisa penalizarile pentru fiecare citator, se va tine cont de urmatorul algoritm: daca intre data de retur si data imprumutului au trecut mai mult de 2 saptamani, si cititorul nu este student, atunci pentru fiecare zi depasita (>14) se adauga 1ron; daca cititorul este student se adauga cate 1ron pentru o perioada mai mare de 28zile. Daca persoana are intre 18 si 14 ani, atunci ea este scutita de la penalizari (se va trece 0). Se va afisa tabela fisa indexata dupa codul citator + penalizari (“top rusinica”)
3. Se va afisa un top al celor mai mari iubitori de carte, si se va afisa acest top. In alcatuirea topului se va tine cont de urmatoare: se vor numara si afisa doar cartile care au fost deja returnate la data curenta. Afisarea va fi facuta in urmatorul mod (evident de la persoana care a citit cele mai multe carti pana la cea care a citit cele mai putine carti)
   * Cod citator, nume, prenume, nr carti citite
4. Se va afisa topul celor mai citite carti – SQL
   * Titlu | autor | de cate ori apare in tabela fisa
5. Se va afisa o statistica pentru fiecare categorie de carti, cea mai indragita categorie cat si varsta medie pentru fiecare categorie:
   * Categorie | varsta medie| nr de carti imprumutate| numar mediu de zile pentru lectura
   * Se va rezolva exclusive in SQL

# Gestiune grupa studenti

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Cod\_student | Nume | Prenume | Adresa | Data\_nasterii | bursier | medie |
| Numeric | Char | Char | Char | Data | logical | numeric |

Persoana

PK

Fisa\_note

|  |  |  |  |
| --- | --- | --- | --- |
| Cod\_student | Cod\_materie | Data\_notare | Nota |
| numeric | Char | data | numeric |

FK FK

discipline

|  |  |  |  |
| --- | --- | --- | --- |
| Cod\_materie | Denumire | Profesor | credite |
| Char | Char | char | Numeric |

PK

1. Se proiecteaza o intefata grafica, prin care utilizatorului i se va cere sa introduca un cod\_student. La apasarea butonului “afiseaza informatii student”, pe baza codui citit se va afisa fie lista tuturor notelor incluzand urmatoarele informatii:
   * Nume, prenume, cod\_student, note, materie
   * Fie un mesaj corespunzator daca persoana nu a fost gasita in baza de date
2. Campul medie este calculate pe baza notelor din fisei de note, ca medie aritmetica. Pentru campul bursier se vor tine cont de urmatoarele criterii: notele trebuie sa fie toate peste 5 (note de trecere), de asemenea media trebuie sa fie peste 8,50 iar numarul de credite obtinute sa fie minim 25. Dupa actualizarea acestor campuri, se vor afisa toti student bursieri, intr-un clasament alt mediei (indiciu: folositi indexarea/sortarea pentru aceasta afisare)
3. Se va face un clasament al celor mai greu de trecut materii (note sub 5), in fct de numarul de note sub 5 vs numarul total de note. Afisarea va fi facuta sub urmatoarea forma:
   * Materie| nr de student picati/nr total de student (exprimat ca procent) – rezolvat prin ambele metode (relationar +sql)
4. Se proiecteaza o intefata grafica, prin care utilizatorului i se va cere sa introduca un cod\_materie. La apasarea butonului “afiseaza informatii materie”, pe baza codui citit se va afisa fie lista tuturor notelor incluzand urmatoarele informatii:
   * Nume, prenume, cod\_student, note, materie, profesor, credite
   * Fie un mesaj corespunzator daca materia nu a fost gasita in baza de date
5. Se va afisa numar de student promovati/ materie la fiecare sesiune de examinare (se vor lua in considare datele trecute pentru fiecare nota), afisare in ordine descrecatoare, corelat cu nr de credite. -> exclusive in SQL.

# Gestiune flota transport

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Cod\_sofer | Nume | Prenume | Adresa | Data\_nasterii | operational |
| Numeric | Char | Char | Char | Data | logical |

Persoana

PK

Fisa\_traseu

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Cod\_sofer | Cod\_ruta | Data\_start | tarif | Data\_terminare |
| numeric | Char | data | numeric | data |

FK FK

rute

|  |  |  |  |
| --- | --- | --- | --- |
| Cod\_ruta | Localitate\_start | Localitate\_final | km |
| Char | Char | char | Numeric |

PK

1. Se proiecteaza o intefata grafica, prin care utilizatorului i se va cere sa introduca un cod\_sofer. La apasarea butonului “afiseaza informatii sofer”, pe baza codui citit se va afisa fie lista tuturor rutelor incluzand urmatoarele informatii:
   * Nume, prenume, cod\_sofer, km, destinatie start, destinatie finala, data de start, data finala
   * Fie un mesaj corespunzator daca persoana nu a fost gasita in baza de date
2. Campul operational este calculate pe baza campurilor din fisa de traseu, in functie de combinatia dintre campurile data start si data terminare. Daca aceasta este mai mare de o zi sau traseul este mai lung de 800km, campul operational va devein fals (default este true). Se vor afisa toti soferii care sunt operationali.
3. Se va face un clasament al soferilor care au condus cei mai multi KM.. Afisarea va fi facuta sub urmatoarea forma, in ordine descrescatoare:
   * Cod sofer| km (relationar +sql)
4. Se proiecteaza o intefata grafica, prin care utilizatorului i se va cere sa introduca un cod\_ruta. La apasarea butonului “afiseaza informatii ruta”, pe baza codui citit se va afisa fie lista tuturor soferilor care au condus pe aceasta ruta incluzand urmatoarele informatii:
   * Nume, prenume, cod\_ruta, km, puncta din ruta, datele traseului, nr de zile
   * Fie un mesaj corespunzator daca ruta nu a fost gasita in baza de date
5. Se va afisa varsta medie a soferilor care au condus pe fiecare ruta in parte cat si o statistica despre fiecare ruta (suma km totali, numar de parcurgeri) -> exclusive in SQL.

# Gestiune echipa sportiva

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Cod\_sportiv | Nume | Prenume | Adresa | Data\_nasterii | titular |
| Numeric | Char | Char | Char | Data | logical |

Persoana

PK

Fisa\_performanta

|  |  |  |
| --- | --- | --- |
| Cod\_sportiv | Cod\_echipa | punctaj |
| numeric | Char | numeric |

FK FK

echipa

|  |  |  |  |
| --- | --- | --- | --- |
| Cod\_echipa | Nume\_echipa | antrenor | Loc\_clasament |
| Char | Char | char | Numeric |

PK

1. Campul loc\_clasament va fi completat dupa urmatorul rationament: se aduna toate punctele obtinute de catre o echipa, iar in functie de ordinea inversa a clasamentului, se actualizeaza si campul loc\_clasamanet
2. Campul titular se actualizeaza dupa urmatorul rationament: raman titular jucatorii care reusesc sa aduca minim 20% din punctele totale obtinute de echipa
3. Se proiecteaza o intefata grafica, prin care utilizatorului i se va cere sa introduca un cod\_echipa. La apasarea butonului “afiseaza informatii echipa”, pe baza codui citit se va afisa fie lista tuturor sportivilor din aceasta echipa:
   1. Nume, prenume, cod\_echipa, nume\_Echipa, antrenor, loc\_clasament
   2. Fie un mesaj corespunzator daca codul de echipa nu a fost gasita in baza de date
4. Pentru fiecare echipa se va afisa varsta medie si numarul de sportivi din echipa si numarul de titulari. Mod de afisare:
   1. Nume echipa | varsta\_medie| nr\_jucatori | nr titular
5. Pornind de la tabela de sportivi, se va realiza un clasament individual al tutor sportivilor, in functie de numarul total de puncte. Se va afisa acest clasament, pornind de la cel mai putin performant sportiv.

# Cinematograf

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Cod\_film | Nume | Regizor | Tip\_film | Data\_premiera | Audienta |
| Numeric | Char | Char | Char | Data | Char |

Filme

PK

bilete

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Cod\_film | Cod\_sala | Data\_spectacol | Pret\_bilet | Ora\_desfasurare |
| numeric | Char | data | numeric | Char |

FK FK

Sala

|  |  |  |  |
| --- | --- | --- | --- |
| Cod\_sala | Tip\_sala | Nume | NrTotalLocuri |
| Char | Char | char | Numeric |

PK

1. Pentru fiecare film si pentru fiecare sala, se va afisa numarul de locuri ramase libere. Indicatii de rezolvare: se vor numara cate bilete sunt vandute pentru un film, o ora de desfasurare si o anumita sala, iar numarul de locuri ramase libere se considera ca fiind diferenta intre NrTotalLocuri – nr de bilete vandute (rezolvare se va face atat in maniera dbase (relationar) cat si SQL
2. Se proiecteaza o intefata grafica, prin care utilizatorului i se va cere sa introduca un cod\_film. La apasarea butonului “afiseaza informatii film”, pe baza codului citit se va afisa fie

a. Nume, regizor, tipul filmului, data premierei, date si orele la care ruleaza si numarul total de bilete vandute

b. Fie un mesaj corespunzator daca filmul nu a fost gasita in baza de date

1. C. se va realiza un clasament al filmelor in functie de numarul total de bilete vandute. (top vanzari bilete). Afisarea se va face in ordinea descrescatoare a sumelor incasate (pret\_bilet \* numar bilete vandute). Pentru usurinta aceasta cerinta va fi implementata in SQL.
2. Se va parcurge tabela de filme si se va analiza compatibilitatea campului audienta (AG/AP/+15/+18) cu orele de rulare. In cazul in care exista neconformitati se afisa un mesaj corespunzator de avertizare (stabiliti voi un orar in care ar fi adecvat de rulat fiecare tip de film).
3. Se proiecteaza o intefata grafica, prin care utilizatorului i se va cere sa introduca un tip de film. La apasarea butonului “afiseaza informatii despre tipul de film”, pe baza codului citit se va afisa
   1. Pentru fiecare tip de film (SF/Drama/etc) si in functie de anul de premiera se va afisa totalul incasarilor din bilete cat si numele filmelor
   2. In cazul in care se introduce un cod eronat, se va afisa un mesaj de eroare.

# Cabinet medic familie

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CNP | Nume | Prenume | Adresa | Data\_nasterii |
| Numeric | Char | Char | Char | Data |

Pacient

PK

Fisa medicala

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| CNP | Cod\_diagnostic | recomandari | Data\_consult | Concediu medical |
| numeric | Char | Char | data | logical |

FK FK

Diagnostice

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Cod\_ diagnostic | denumire | medicamente | Repaos\_recomandat | Nr\_zile |
| Char | Char | char | logic | numeric |

1. Se proiecteaza o intefata grafica, prin care utilizatorului i se va cere sa introduca un CNP. La apasarea butonului “afiseaza informatii pacient”, pe baza CNP-ului citit se va afisa fie lista tuturor vizitelor medicale incluzand urmatoarele informatii:
   * Nume, prenume, cnp, data\_Consult, diagnostic
   * Fie un mesaj corespunzator daca persoana nu a fost gasita in baza de date
2. Pentru tabela *Fisa medicala* campul concediu medical se verifica dupa urmatorul algoritm:
   * Daca in tabela de diagnostic campul repaos\_recomandat in corelatie cu codul de diagnostic este fals, atunci obligatoriu acest camp va fi trecut pe fals
3. Se va calcula si afisa care este diagnosticul pentru care au fost alocate cele mai multe zile de concediu pentru un an citit de la tastura. (adaugati un camp de citire + buton de interogare in interfata grafica).
   * Exemplu de afisare:
   * Cod\_destinatie locatie nr\_persoane
     + Persoana cnp a petrecut x zile
4. Pentru tabela de *persoane*, se va face un clasament cu pacientii care au avut cele mai multe vizite medicale, afisand pentru fiecare si suma totala de numar de zile de concediu medical de-a lungul timpului. - SQL
5. Pentru un pacient al carui cnp este citit de la tastatura se va afisa de cate ori s-a prezentat la control. (se considera control atunci cand diferenta dintre consultati este mai mica de 5zile lucratoare). De asemenea se va verifica daca campul cu recomandari difera pentru acelasi tip de diagnostic, dupa control.