

Campionat de sarituri cu schiurile

In acest proiect am creat o baza de date care va gestiona informații legate de organizarea și funcționarea editiei 2022-2023 a campionatului de sarituri cu schiurile.

Saritorii au un istoric de accidentari, aparținând câte unei echipe care poate fi sponsorizată de mai multe firme.

Cativa saritori pot aparține și echipei naționale, unde vor primi un număr de ordine.

Saritorii participă la mai multe etape din cadrul a mai multor competiții care se suprapun, rezultatele dintr-o etapă putând conta în mai multe competiții.

Fiecare etapă se desfășoară pe o trambulină, dintr-un oraș care se găsește într-o țară.

Fiecare rezultat al unui saritor va conta și la palmaresul saritorilor. Acolo unde respectivul se află în top 3 acesta primește o medalie.

RESTRICTII

- 1.Orice saritor poate avea mai multe accidentari sau niciuna.
- 2.Un saritor poate aparține unei singure echipe.
- 3.O echipă poate fi sponsorizată de mai mulți sponsori sau de niciunul, fiecare sponsor putând sponsoriza mai multe echipe.
- 4.Un saritor participă la competiții și la etapele acestora, majoritatea etapelor făcând parte din competiția "Cupa mondială",putând să aparțină și de alte competiții precum "Turneul celor 4 trambuline" simultan , dar o competiție poate fi și de sine statătoare.
- 5.O etapă se desfășoară pe o singură trambulină, dintr-un oraș, dintr-o țară. Într-o țară putând fi mai multe orașe în care pot fi mai multe trambuline.
- 6.Rezultatul este dependent de etapă, fiecare etapă având un podium.
- 7.Un saritor poate avea mai multe medalii sau niciuna.

ENTITATI

Pentru modelul de date ce se referă la sarituri cu schiurile, structurile SARITORI, ACCIDENTARI, COMPETITIE, ETAPA, TRAMBULINA, ORAS, TARA, REZULTAT, MEDALII, ECHIPA, SPONSOR, MEMBRU_ECHIPA_NAT reprezintă entități.

Toate entitățile care vor fi prezentate sunt independente, cu excepția entităților dependente ACCIDENTARI, MEDALII, REZULTAT și a subentității MEMBRU_ECHIPA_NAT.

SARITOR=persoana fizica, sportiv de performanta care aparține de o echipă și care participă la competițiile de sarituri cu schiurile. Cheia primară este *id_saritor*

ACCIDENTARE= entitate dependentă de SARITOR care detine informații despre accidentările acestuia. Cheia primară a entității este compusă din *id_saritor* și *id_accident*.

ECHIPA= echipa formata din mai multi saritori, de aceeasi nationalitate, care are un singur antrenor comun, ocupandu-se de pregatirea sportivilor. Cheia primara a entitatii este *id echipa*.

SPONSOR= persoana juridica care ofera bunuri si bani echipelor in schimbul promovarii. Mai multe echipe pot fi sponsorizate de mai multi sponsori. Cheia primară a acestei entități este *id_sponsor*.

ETAPA= doua manse de sarituri pe aceeasi trambulina, in aceleasi conditii, la finalul carora avem un singur rezultat. O etapa poate face parte din mai multe competitii. Cheia primara a entitatii este *id_etapa*.

COMPETITIE= concurs format din mai multe etape la care pot participa mai multi saritori. Cheia primara este *id_competitie*.

TRAMBULINA= locatia unde se tin etapele competitiei, fiecare trambulina are dimensiuni diferite si un record care poate fi depasit. Cheia primara este *id_trambulina*.

ORAS=entitate care identifica orasul in care se afla o trambulina pe care se tine etapa. Cheia primara a entitatii este *id_oras*.

TARA=exprima tara in care se gaseste orasul in care se desfasoara o etapa. Cheia primara este *id_tara*.

REZULTAT= entitate dependenta de ETAPA si de SARITOR care detine informatii despre rezultatul din finalul unei etape al unui saritor. Cheia primara a entitatii este compusa din *id_rezultat* , *id_etapa* si *id_saritor*.

MEDALII= entitate dependenta de SARITOR si de REZULTAT. In urma rezultatelor pe podium, saritorilor castigatori li se atribuie o medalie. Cheia primara a entitatii este compusa din *id_medalie*, *id_saritor* si *id_rezultat*.

MEMBRU_ECHIPA_NAT= entitate dependenta de SARITOR, reprezentand un saritor care este inscris si in echipa nationala. Cheia primara a entitatii este *id_saritor#*.

RELATII

SARITOR apartine_la ECHIPA= relatie care leaga entitatile SARITOR si ECHIPA, reflectand legatura dintre acestea (din ce echipa face parte un anumit saritor). Ea are cardinalitatea minima 1:1 (un saritor trebuie sa apartina la o echipa si la o echipa trebuie sa apartina cel putin un saritor) si cardinalitatea maxima 1:m(un saritor poate sa apartina unei singure echipe, iar de o echipa pot sa apartina mai multi saritori).

SPONSOR sponsorizeaza ECHIPA = relatie de tip many-to-many dintre entitatea ECHIPA și SPONSOR, reflectând legătura dintre acestea (ce sponsori sponsorizeaza o anumita echipa). Ea are cardinalitatea minimă 1:0 (cel puțin o echipa trebuie sa fie sponsorizata de un sponsor si niciun sponsor nu trebuie sa sponsorizeze o anumita echipa) și cardinalitatea maximă m:n(o echipa poate fi sponsorizata de mai multi sponsori si un sponsor poate sponsoriza mai multe echipe).

SARITOR participa_la ETAPA din COMPETITIE= relatie de tip 3 ce leaga entitatile SARITOR, ETAPA, COMPETITIE, reflectand ce saritor participa la ce etapa, din ce competitie. Denumirea acestei relatii va fi participa_la.

ETAPA se_desfasoara TRAMBULINA= relație dintre entitatea ETAPA și TRAMBULINA, reflectând legătura dintre acestea (ce trambulina este alocata unei etape). Ea are cardinalitatea minimă 1:1 și cardinalitatea maximă 1:n(o etapa se poate desfasura pe o singura trambulina, iar pe o trambulina se pot organiza mai multe etape).

TRAMBULINA este_in ORAS= relație dintre entitatea ORAS și TRAMBULINA, reflectând legătura dintre acestea (in ce oras este o trambulina). Ea are cardinalitatea minimă 1:1 și cardinalitatea maximă 1:n(o trambulina se poate afla intr-un singur oras in timp ce intr-un oras pot fi mai multe trambuline).

ORAS apartine_de TARA= relație dintre entitatea ORAS și TARA, reflectând legătura dintre acestea (de ce tara apartine un oras). Ea are cardinalitatea minimă 1:1 și cardinalitatea maximă 1:n(un oras se poate afla intr-o singura tara in timp ce intr-o tara pot fi mai multe oras).

ETAPA s-a obtinut REZULTAT= relație dintre entitatea ETAPA și REZULTAT, reflectând legătura dintre acestea (ce rezultat s-a obtinut la o etapa). Ea are cardinalitatea minimă 1:1 și cardinalitatea maximă 0:n(dintr-o etapa se pot obtine mai multe rezultate, dar si niciunul daca aceasta se anuleaza, iar un rezultat se poate obtine dintr-o singura etapa).

SARITOR s-a obtinut REZULTAT= relație dintre entitatea SARITOR și REZULTAT, reflectând legătura dintre acestea (ce rezultat a obtinut un saritor). Ea are cardinalitatea minimă 1:1 și cardinalitatea maximă 1:n(un saritor poate obtine mai multe rezultate dar un rezultat nu poate fi obtinut decat de un saritor).

SARITOR are MEDALIE= relatie dintre entitatea MEDALIE si SARITOR, reflectand legatura dintre acestea (ce medalie are un saritor). Ea are cardinalitatea minima 1:1 si cardinalitatea maxima 1_n (un saritor poate avea mai multe medalii, in timp ce o medalie poate apartine unui singur saritor).

REZULTAT reprezinta MEDALIE= relație care leagă REZULTAT și MEDALIE (ce rezultat reprezina o medalie). Ținând cont de restricțiile impuse modelului, relația are cardinalitatea minimă 1:1 și cea maximă 1:1.

SARITOR a_avut ACCIDENTARE= relație care leagă entitățile SARITOR si ACCIDENTARE, reflectând legătura dintre acestea (pentru o accidentare aflam saritorul care a comis-o). Relația are cardinalitatea minimă 1:1 și cardinalitatea maximă 0:n(un saritor nu trebuie sa nu aiba nicio accidentare, dar poate avea mai multe).

ATTRIBUTE

Entitatea independentă SARITOR are ca atribute:

id_saritor =variabilă de tip întreg, de lungime maximă 5, care reprezintă codul unui saritor.

nume= variabilă de tip caracter, de lungime maximă 25, care reprezintă numele saritorului.

prenume= variabilă de tip caracter, de lungime maximă 25, care reprezintă prenumele saritorului.

gen= variabilă de tip caracter, luând valorile m sau f, de lungime 1, care reprezintă genul angajatului.

varsta= variabilă de tip numeric care reprezinta varsta saritorului

greutate= variabilă de tip numeric (real) care reprezinta greutatea in kilograme a saritorului

id echipa= variabilă de tip caracter, de lungime maximă 3, care reprezintă codul echipei din care face parte saritorul respectiv. Atributul trebuie să corespundă la o valoare a cheii primare din tabelul ECHIPA.

tip= variabila de tip numeric,0 sau 1, care ne spune daca saritorul este membru al echipei nationale.

Subentitatea MEMBRU_ECHIPA_NAT are ca attribute:

id_saritor=variabilă de tip întreg, de lungime maximă 5, care reprezintă codul unui saritor.

nr_tur_saritura=variabilă de tip numeric, de lungime 1, între 1 și 4, care reprezinta turul in care fiecare saritor va performa

Entitatea dependentă ACCIDENTARE are ca attribute:

id_accident=variabila de tip intreg, de lungime maxima 3, care reprezinta codul accidentului.

id_saritor=variabilă de tip întreg, de lungime maximă 5, care reprezintă codul unui saritor.

data_accidentare=variabilă de tip dată calendaristică reprezentand ziua in care s-a produs accidentul.

Entitatea independentă ECHIPA are ca attribute:

id echipa=variabilă de tip caracter, de lungime maximă 3, care reprezintă codul echipei din care face parte saritorul respectiv.

nume echipa=variabilă de tip caracter, de lungime maximă 25, care reprezintă numele echipei.

antrenor=variabilă de tip caracter, de lungime maximă 50, care reprezintă numele antrenorului.

salariu_antrenor=variabilă de tip numeric, mai mare decat 2000, reprezentand salariul in RON al antrenorului echipei.

Relația SPONSOR sponsorizeaza ECHIPA are ca attribute id echipa, id_sponsor, durata_contract(variabila de tip numeric reprezentnd numarul de luni pe care este valabil contractul), suma(variabila de tip numeric, reprezentand suma care este data drept sponsorizare).

Entitatea independentă SPONSOR are ca attribute:

id_sponsor=variabilă de tip întreg, de lungime maximă 5, care reprezintă codul unui sponsor.

denumire=variabilă de tip caracter, de lungime maximă 40, care reprezintă numele sponsorului(persoana_juridica).

Relația SARITOR participa_la ETAPA din COMPETITIE are ca attribute: id_saritor, id_competitie, id_etpa, descriere_pieptar(variabilă de tip caracter, de lungime maximă 40, reprezentand modelul pe care il va avea pieptarul purtat de saritor, diferit in functie de competitive si de etapa).

Entitatea independentă COMPETITIE are ca attribute:

id_competitie=variabilă de tip întreg, de lungime maximă 7, care reprezintă codul unui sponsor.

nume= variabilă de tip caracter, de lungime maximă 45, care reprezintă numele saritorului.

data_inceput=variabilă de tip dată calendaristică reprezentand ziua in care incepe competitia

data_final=variabilă de tip dată calendaristică reprezentand ziua in care se termina competitia, luna trebuie sa fie cel tarziu aprilie.

numar_etape=variabilă de tip numeric, reprezentand numarul de etape dintr-o competitie.

Entitatea independentă ETAPA are ca attribute:

id_etapa=variabilă de tip întreg, care reprezintă codul etapei.

Data_etapa=variabilă de tip dată calendaristică reprezentand ziua in care are loc etapa

tip=variabilă de tip caracter, luând valorile "saritura" sau "zbor", care reprezintă tipul de etapa

zi/noapte=variabilă de tip caracter, luând valorile "noapte" sau "zi", care reprezintă timpul desfasurarii etapei.

id_trambulina=variabilă de tip întreg, care reprezintă codul trambulinei.

Entitatea independentă TRAMBULINA are ca attribute: id_trambulina, nume, pct_de_constructie, record, dimensiune, id_oras.

Entitatea independentă ORAS are ca attribute: id_oras,nume, id_tara;

Entitatea independentă TARA are ca attribute: id_tara,nume;

Entitatea dependentă REZULTAT are ca attribute:

id_saritor =variabilă de tip întreg, de lungime maximă 5, care reprezintă codul unui saritor.

id_rezultat=variabilă de tip întreg, de lungime maximă 5, care reprezintă codul unui rezultat.

id_etapa=variabilă de tip întreg, care reprezintă codul etapei.

lungime1=variabilă de tip numeric, care reprezinta lungimea primei sarituri

lungime2=variabilă de tip numeric, care reprezinta lungimea celei de-a doua sarituri

Entitatea dependentă MEDALIE are ca atribute:

id_medalie=variabilă de tip întreg, de lungime maximă 5, care reprezintă codul unei medalii.

id_rezultat=variabilă de tip întreg, de lungime maximă 5, care reprezintă codul unui rezultat.

id_saritor=variabilă de tip întreg, de lungime maximă 5, care reprezintă codul unui saritor.

tip_medalie=variabilă de tip caracter, luând valorile "aur", "argint" sau "bronz", care reprezintă tipul de medalie detinuta de un saritor

DIAGRAMA ENTITATE-RELATIE

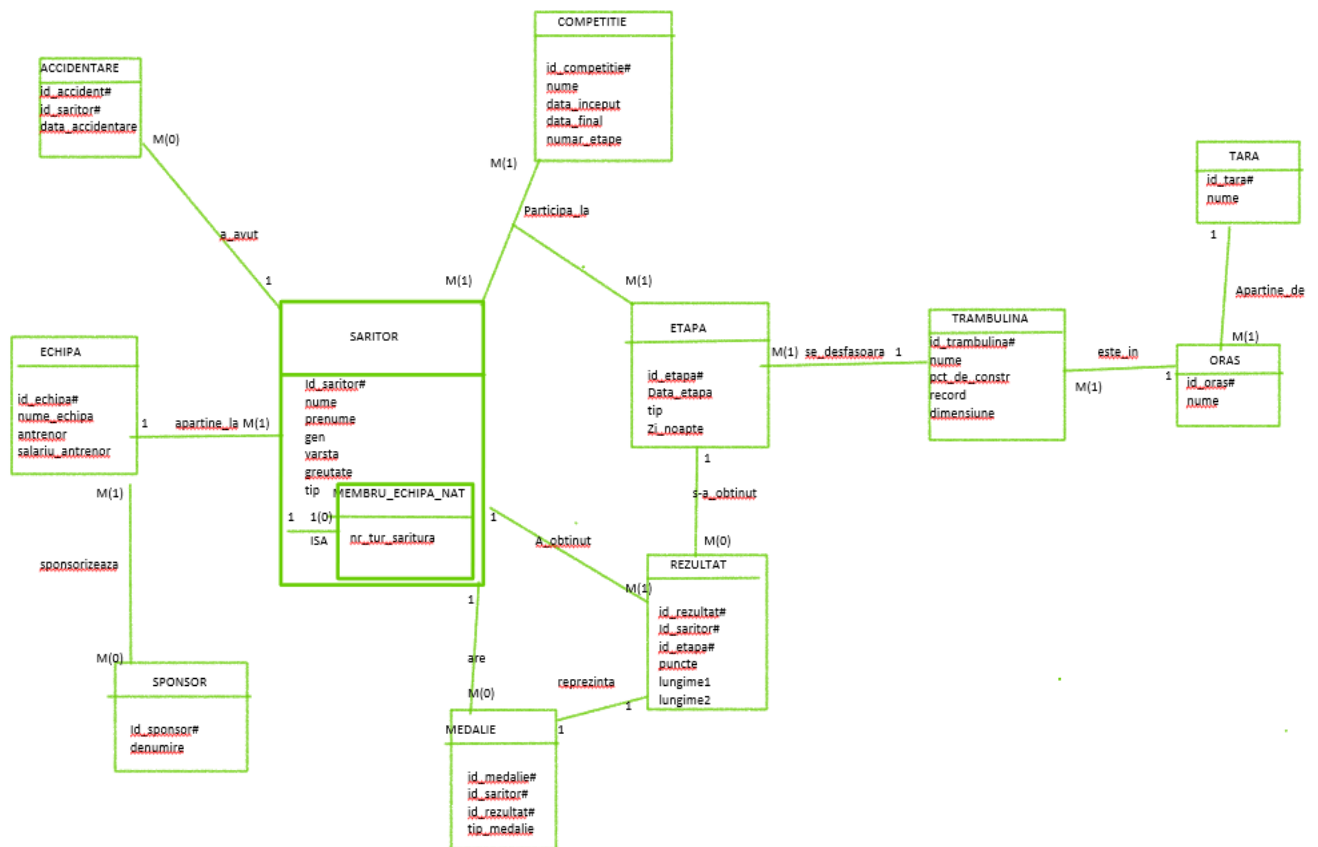
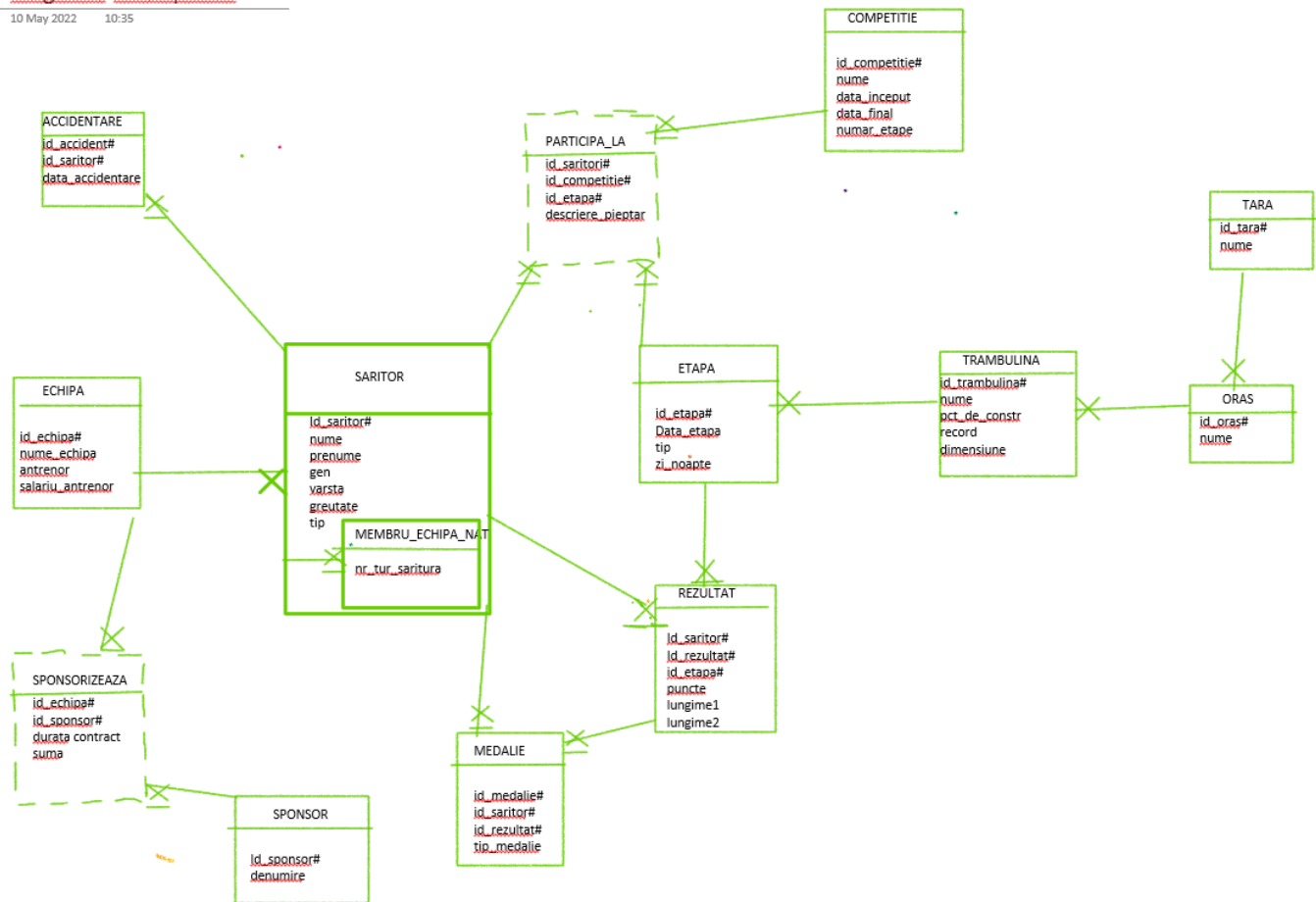


DIAGRAMA CONCEPTUALA

Diagrama conceptuala

10 May 2022 10:35



SCHEMELE RELATIONALE

SARITOR(id_saritor#, nume, prenume, gen, varsta, greutate, tip, id_echipa)

MEMBRU_ECHIPA_NAT(id_saritor#, nr_tur_saritura)

ACCIDENTARE(id_accidentare#, id_saritor#, data_accidentare)

ECHIPA(id_echipa#, nume_echipa, antrenor, salariu_antrenor)

SPONSORIZEAZA(id_echipa#, id_sportiv#, durata_contract, suma)

SPONSOR(id_sponsor#, denumire)

MEDALIE(id_medalie#, id_saritor#, id_rezultat#, tip_medalie)

PARTICIPA_LA(id_saritor#, id_competitie#, id_etapa#, descriere_pieptar)

COMPETITIE(id_competitie#, nume, data_inceput, data_final, numar_etape)

ETAPA(id_etapa#, data_etapa, tip, zi/noapte, id_trambulina)

TRAMBULINA(id_trambulina, nume, pct_de_constr, record, dimensiune, id_oras)

ORAS(id_oras#, nume, id_tara)

TARA(id_tara#,nume)

REZULTAT(id_rezultat#, id_etapa#,id_saritor#, lungime1, lungime2)

FORMA NORMALA 1

O relatie se afla in FN1 daca fiecarui atribut care o compune ii corespunde o valoare indivizibila, adica, o relatie se afla in FN1 daca exista un identificatory unic.

Un EXEMPLU de relatie care nu e FN1 :

Presupunem ca atributul antrenor din relatia ECHIPA nu este unic, putand lua mai multe valori.

Id echipa#	Nume echipa	antrenor	Salariu antrenor
POL	Polonia	x	4500
SLO	Slovenia	Y,yy,yyy	3000
GER	Germania	Z,zz	3400

Relatia de mai sus nu se afla in FN1, pentru ca atributului antrenor nu ii corespunde o valoare indivizibila.

Id echipa#	Nume echipa	antrenor	Salariu antrenor
POL	Polonia	x	4500
SLO	Slovenia	Y	3000
GER	Germania	Z,zz	3400
SLO	Slovenia	yy	3000
SLO	Slovenia	yyy	3000
GER	Germania	zz	3400

Se poate observa ca dupa normalizare fiecarui atribut ii corespunde o valoare indivizibila.

In relatia ACCIDENTARE exista o cheie primara compusa formata din: id_saritor, id_accidentare. Aceasta cheie primara este un identificator unic, avand o valoare indivizibila=>relatia este in FN1. In plus, fiecare accidentare poate avea loc intr-o singura zi, atributul data_accidentare fiind si el indivizibil.

In relatia SARITOR avem cheia primara id_saritor, ceea ce inseamna ca avem un identificator unic.In acest fel, atributul id_saritor fiind o valoare indivizibila, relatia se afla in FN1. Celelalte atribute: nume, prenume,gen, varsta,greutate,tip au valori indivizibile.

In relatia MEMBRU_ECHIPA_NAT avem cheia primara id_saritor, ceea ce inseamna ca avem un identificator unic.In acest fel, atributul id_saritor fiind o valoare indivizibila, relatia se afla in FN1. Celelalte atribute: nume, prenume,gen, varsta,greutate,tip, nr_tur_saritura au valori indivizibile.

In relatia ECHIPA avem cheia primara id echipa, ceea ce inseamna ca avem un identificator unic.In acest fel, atributul id echipa fiind o valoare indivizibila, relatia se afla in FN1. Celelalte atribute: nume echipa, antrenor, salariu_antrenor au valori indivizibile.

In relatia SPONSORIZEAZA exista o cheie primara compusa formata din: id echipa, id_sponsor. Aceasta cheie primara este un identificator unic, avand o valoare indivizibila=>relatia este in FN1. In plus, durata unui contract si suma acestuia sunt valori indivizibile.

In relatia SPONSOR avem cheia primara id_sponsor, ceea ce inseamna ca avem un identificator unic. In acest fel, atributul id_sponsor fiind o valoare indivizibila, relatia se afla in FN1. In plus, denumirea nu poate lua mai multe valori, fiind indivizibila.

In relatia PARTICIPA_LA exista o cheie primara compusa formata din: id_saritor, id_competitie, id_etapa. Aceasta cheie primara este un identificator unic, avand o valoare indivizibila=>relatia este in FN1. In plus, descriere_pieptar este un atribut indivizibil.

In relatia COMPETITIE avem cheia primara id_competitie, ceea ce inseamna ca avem un identificator unic. In acest fel, atributul id_competitie fiind o valoare indivizibila, relatia se afla in FN1. Celelalte attribute: nume, data_inceput, data_final, numar_etape au valori indivizibile.

In relatia ETAPA avem cheia primara id_etapa, ceea ce inseamna ca avem un identificator unic. In acest fel, atributul id_etapa fiind o valoare indivizibila, relatia se afla in FN1. Celelalte attribute: data, tip, zi/noapte au valori indivizibile.

In relatia TRAMBULINA avem cheia primara id_trambulina, ceea ce inseamna ca avem un identificator unic. In acest fel, atributul id_trambulina fiind o valoare indivizibila, relatia se afla in FN1. Celelalte attribute: nume, pct_de_constr, record, dimensiune au valori indivizibile.

In relatia ORAS avem cheia primara id_oras, ceea ce inseamna ca avem un identificator unic. In acest fel, atributul id_oras fiind o valoare indivizibila, relatia se afla in FN1. In plus, numele este un atribut indivizibil.

In relatia TARA avem cheia primara id_tara, ceea ce inseamna ca avem un identificator unic. In acest fel, atributul id_tara fiind o valoare indivizibila, relatia se afla in FN1. In plus, numele este un atribut indivizibil.

In relatia REZULTAT exista o cheie primara compusa formata din: id_rezultat, id_etapa, id_saritor. Aceasta cheie primara este un identificator unic, avand o valoare indivizibila=>relatia este in FN1. Celelalte attribute: puncte, lungime1, lungime2 au valori indivizibile.

In relatia MEDALIE exista o cheie primara compusa formata din: id_medalie, id_saritor. Aceasta cheie primara este un identificator unic, avand o valoare indivizibila=>relatia este in FN1. In plus, tipul medaliei are valoare indivizibila.

FORMA NORMALA 2

O relatie se afla in FN2 daca si numai daca aceasta se afla in fn1 si daca fiecare atribut care nu face parte din cheie este dependent de intreaga cheie primara. Presupunem ca exista relatiile TARA_ORAS, ORAS_TRAMBULINA si TRAMBULINA_ETAPA:

- Relatia TARA_ORAS

Id_tara#	Id_oras#	Nume_tara	Nume_oras
1	11	Austria	Innsbruck
2	22	Polonia	Zakopane
3	33	Germania	Oberstdorf
4	44	Slovenia	Planica

Relatia se afla in FN1 pentru ca exista un identificator unic pentru toate intrarile din tabel.

Pentru a fi FN2, attributele nume_tara si nume_oras trebuie sa depinda de intreaga cheie primara compusa: id_oras#, id_rezultat#. Acest lucru nu se intampla, din cauza dependentei directe dintre id_oras si nume_oras. De aceea, relatia nu se afla in FN2.

Dependentele sunt:

{id_tara#}->{nume_tara}

{id_oras#}->{nume_oras}

Transformare in FN2=> 2 tabele

TARA

Id_tara#	Nume_tara
1	Austria
2	Polonia
3	Germania
4	Slovenia

ORAS

Id_oras#	Nume_oras
11	Innsbruck
22	Zakopane
33	Oberstdorf
44	Planica

- Relatia ORAS-TRAMBULINA

Id_oras #	Nume_oras	Id_trambulina #	nume	Pct_de_const r	record	dimensiune
11	Innsbruck	1	Bergisel	120	144	128
22	Zakopane	2	Wielka Krokiew	125	150	140
33	Oberstdorf	3	Schattenbergschante	120	143.5	137
44	Planica	4	Letalnica	185	239	215

Relatia se afla in FN1 pentru ca exista un identificator unic pentru toate intrarile din tabel.

Dependentele sunt:

{id_trambulina}->{nume, pct_de_constr, record, dimensiune}

{id_oras#}->{nume_oras}

Din aceste dependente reiese faptul ca relatia nu se afla in FN2.

ORAS

Id_oras#	Nume_oras
11	Innsbruck
22	Zakopane
33	Oberstdorf
44	Planica

TRAMBULINA

Id_trambulina#	nume	Pct_de_constr	record	dimensiune
1	Bergisel	120	144	128
2	Wielka Krokiew	125	150	140
3	Schattenbergschante	120	143.5	137
4	Letalnica	185	239	215

- Relatia TRAMBULINA-ETAPA#

Id_etapa#	data	tip	Zi/noapte	Id_trambulina#	nume	Pct_de_constr	record	dimensiune
11	22.12.2022	sarit	Zi	1	Bergisel	120	144	128
22	25.12.2022	sarit	Zi	2	Wielka Krokiew	125	150	140
33	03.01.2023	sarit	Noapte	3	Schattenbergschante	120	143.5	137
44	08.01.2022	zbor	noapte	4	Letalnica	185	239	215

Relatia se afla in FN1 pentru ca exista un identificator unic pentru toate intrarile din tabel.

Dependentele sunt:

{id_trambulina}->{nume, pct_de_constr, record, dimensiune}

{id_etapa#}->{data,tip,zi/noapte}

Din aceste dependente reiese faptul ca relatia nu se afla in FN2.

TRAMBULINA

Zi/noapte	Id_trambulina#	nume	Pct_de_constr	record	dimensiune
Zi	1	Bergisel	120	144	128
Zi	2	Wielka Krokiew	125	150	140
Noapte	3	Schattenbergschante	120	143.5	137
noapte	4	Letalnica	185	239	215

ETAPA

Id_etapa#	data	tip	Zi/noapte
11	22.12.2	sarit	Zi

	022		
22	25.12.2 022	sarit	Zi
33	03.01.2 023	sarit	Noapte
44	08.01.2 022	zbor	noapte

- Relatia:REZULTAT

Aceasta relatie se afla in FN1 pentru ca exista un identificatory unic pentru toate intrarile din tabel.

Se poate remarca faptul ca attributele pozitie_clasament, puncta, lungime1, lungime2 care nu sunt in componenta cheii primare compuse sunt dependente de intreaga cheie.=>relatia se afla in FN2.

- Relatia: MEDALIE

Aceasta relatie se afla in FN1 pentru ca exista un identificator unic pentru toate intrarile din tabel.

Se poate remarca faptul ca atributul tip_medalie care nu este in componenta cheii primare compuse este dependent de intreaga cheie.=>relatia se afla in FN2.

- Relatia:ECHIPA

Aceasta relatie se afla in FN1 pentru ca exista un identificator unic pentru toate intrarile din tabel.

Se poate remarca faptul ca attributele nume_echipa, antrenor, salariu_antrenor care nu sunt in componenta cheii primare sunt dependente de intreaga cheie.=>relatia se afla in FN2.

- Relatie: ACCIDENTARE

Aceasta relatie se afla in FN1 pentru ca exista un identificator unic pentru toate intrarile din tabel.

Se poate remarca faptul ca atributul data_accidentare care nu este in componenta cheii primare compuse este dependent de intreaga cheie. =>relatia se afla, in FN2.

- Relatie: SPONSOR

Aceasta relatie se afla in FN1 pentru ca exista un identificator unic pentru toate intrarile din tabel.

Se poate remarca faptul ca atributul denumire care nu este in componenta cheii primare este dependent de intreaga cheie. =>relatia se afla, in FN2.

- Relatia:COMPETITIE

Aceasta relatie se afla in FN1 pentru ca exista un identificator unic pentru toate intrarile din tabel.

Se poate remarca faptul ca attributele nume, data_inceput, data_final, numar_etape care nu sunt in componenta cheii primare sunt dependente de intreaga cheie.=>relatia se afla in FN2.

Un EXEMPLU in care o relatie nu s- ar fi aflat in FN2 este:

Presupunem ca in relatia PARTICIPA_LA , pe langa cheia compusa si atributul descriere_pieptar,avem ca atribut data_inscriere, referindu-se la data de inscriere a unui saritor la o competitie. In acest caz, data_inscriere este dependenta doar de id_saritor# si de id_competitie#, deci relatia nu se afla in FN2, fiind necesare doua tabele. Unul cu attributele id_saritor#, nume,

prenume, gen ,varsta,greutate,tip, id_competitie#, nume, data_inceput, data_final, numar_etape, data_inscriere si unul cu attributele id_saritor#, nume, prenume, gen ,varsta,greutate,tip, id_competitie#, nume, data_inceput, data_final, numar_etape, data_inscriere, id_etapa#, data, tip, zi/noapte, descriere_pieptar.

FORMA NORMALA 3

Definitie. O relatie se afla in FN3 daca si numai daca fiecare atribut care nu este cheie, depinde direct de cheia primara si numai de ea.

Presupunem ca in tabelul REZULTAT_MEDALIE avem atributul pozitie_clasament care depinde de cheia primara id_rezultat#,id_etapa# prin atributul puncte.=> relatia nu este in FN3.

Id_rezultat#	Id_etapa#	puncte	Lungime1	Lungime2	Poz_clasament
1	11	213	145	140	2
2	22	222	149	139	1

Relatiile de dependenta:

{id_rezultat#,id_etapa#}->{puncte,lungime1,lungime2}

{id_rezultat#, id_etapa#}->{puncte}->{pozitie_clasament}

Pentru a rezolva aceasta problema am avem nevoie de doua tabele: REZULTAT si POZITIE.

Id_rezultat#	Id_etapa#	puncte	Lungime1	Lungime2
1	11	213	145	140
2	22	222	149	139

POZITIE

puncte	Pozitie_clasament
213	2
222	1

CREARE TABELE (am adaugat cgh la finalul numelui pentru a le diferentia de alte tabele din server)

create table SARITORCGH

(

id_saritor number(5,0) not null primary key,

nume varchar(30) not null,

prenume varchar(30) not null,

gen varchar(1),

varsta number(2,0) not null,

greutate number(2,1) not null,

```
tip number(1,0) not null,  
id echipa varchar(3) not null,  
foreign key id echipa references ECHIPACGH(id echipa) on delete cascade  
);
```

```
create table ECHIPACGH(  
id echipa varchar(3) not null primary key,  
nume echipa varchar(25) not null,  
antrenor varchar(50) not null,  
salariu_antrenor numeric(4,0) check(salariu_antrenor>2000)  
);
```

```
create table ACCIDENTARECGH(  
id_accident number(3,0) not null primary key,  
id_saritor number(5,0) not null,  
data_accidentare DATE,  
foreign key (id_saritor) references SARITORCGH(id_saritor) on delete cascade  
);
```

```
create table SPONSORIZEAZACGH(  
id echipa varchar(5) not null,  
id_sponsor number(5,0) not null,  
durata_contract number(2,0) not null,  
suma number(4,0) not null,  
constraint pk_sponsorizeaza primary key(id_sponsor,id echipa)  
);
```

```
create table SPONSORCGH(  
id_sponsor number(5,0) not null primary key,  
denumire varchar(40) not null  
);
```

```
create table COMPETITIECGH(  
id_competitie number(7,0) not null primary key,  
nume varchar(45) not null,  
data_inceput DATE,  
data_final DATE,  
numar_etape number(1,0) not null  
);
```

```
create table TARACGH(  
id_tara number(2,0) not null primary key,  
nume varchar(15) not null  
);
```

```
create table ORASCGH(  
id_oras number(3,0) not null primary key,  
id_tara number(2,0) not null,  
nume varchar(15) not null,  
foreign key (id_tara) references TARACGH(id_tara) on delete cascade  
);
```

```
create table TRAMBULINACGH(  
id_trambulina number(3,0) not null primary key,  
id_oras number(3,0) not null,  
nume varchar(40) not null,  
pct_de_constr number(3,0) not null,  
record number(3,1),  
dimensiune number(3,0) not null,  
foreign key (id_oras) references ORASCGH(id_oras) on delete cascade  
);
```

```
create table ETAPACGH (  
id_trambulina number(3,0) not null,  
id_etapa number(3,0) not null primary key,  
data_etapa DATE,
```

```

tip varchar(9) not null,
zi_noapte varchar(7) not null,
foreign key (id_trambulina) references TRAMBULINACGH(id_trambulina) on delete cascade
);
commit;
create table PARTICIPA_LACGH (
id_etapa number(3,0) not null,
id_competitie number(7,0) not null,
id_saritor number(5,0) not null,
descriere_pieptar varchar(50),
constraint pk_participa_la primary key(id_competitie,id_saritor,id_etapa)
);
create table MEMBRU_ECHIPA_NATCGH
(
id_saritor number(5,0) not null references SARITORCGH(id_saritor),
nr_tur_saritura number(1,0) not null,
primary key (id_saritor)
);
create table REZULTATCGH(
id_saritor number(5,0) not null,
id_etapa number(3,0) not null,
id_rezultat number(5,0) not null,
puncte number(5,2)not null,
lungime1 number(5,1),
lungime2 number(5,1),
constraint pk_rezultat primary key(id_etapa,id_rezultat,id_saritor)
);
create table MEDALIECGH(
id_medalie number(5,0) not null,
id_saritor number(5,0) not null,
id_rezultat number(4,0) not null,

```



```
tip_medalie varchar(10) not null,  
constraint pk_medalie primary key(id_rezultat,id_medalie,id_saritor)  
);
```

INSERARI

- voi atasa cate un exemplu de inserare pentru fiecare tabel(am adaugat cgh la finalul numelui pentru a le diferentia de alte tabele din server)

```
insert into SPONSORIZEAZACGH values
```

```
('SLO',246,18,4700);
```

```
insert into SARITORCGH values
```

```
('501','Kraft','Stefan','m',29,56,1,'AUT');
```

```
insert into COMPETITIECGH values
```

```
(45678, 'Planica 3','21-FEB-2022','05-MAR-2023',3);
```

```
insert into TARACGH values
```

```
(1,'Polonia');
```

```
insert into ORASCGH values
```

```
(11,1,'Zakopane');
```

```
insert into TRAMBULINACGH values
```

```
(121,12,'Adam Malysz',120,116,134);
```

```
insert into ETAPACGH values
```

```
(321,11,'03-JAN-2023','sarit','zi');
```

```
insert into PARTICIPA_LACGH values
```

```
(1,23456,301,'5 galben milka');
```

```
insert into REZULTATCGH values
```

```
(100,1,1001,222,152,150);
```

```
insert into MEDALIECGH values
```

```
(100,20,10020,'argint');
```

```
insert into ACCIDENTARECGH values
```

```
(111,100,'25-MAR-21');
```

```
insert into ECHIPACGH values
```

```
('NOV','Norvegia','Roar Ljokelsoy',2300);
```

insert into SPONSORCGH values

(456,'4F');

insert into MEMBRU_ECHIPA_NATCGH values

(404,4);

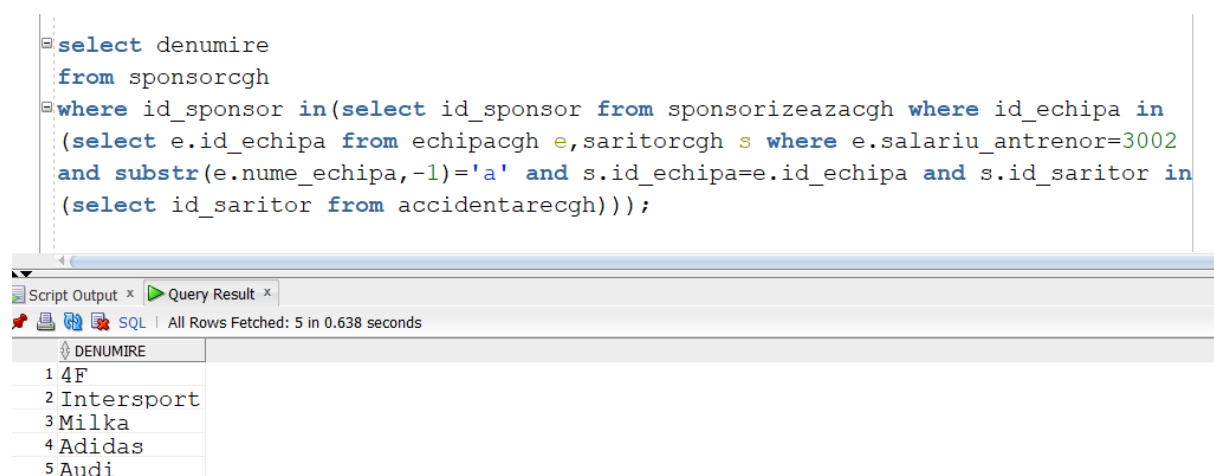
5 CERERI SQL COMPLEXE

1. Afisati denumirea sponsorului care sponsorizeaza o echipa in care salariul antrenorului este 3002, al carei nume se termina in 'A' si care continue sportive ce au suferit o accidentare.(subcerere nesincronizata, functie pe sir de caractere)

select denumire

from sponsorcgh

where id_sponsor in(select id_sponsor from sponsorizeazacgh where id echipa in (select e.id echipa from echipacgh e,saritorcgh s where e.salariu_antrenor=3002 and substr(e.num_echipa,-1)='a' and s.id echipa=e.id echipa and s.id_saritor in (select id_saritor from accidentarecgh)));



The screenshot shows a SQL query editor with a query window and a results window. The query window contains the following SQL code:

```
select denumire
from sponsorcgh
where id_sponsor in(select id_sponsor from sponsorizeazacgh where id echipa in
(select e.id echipa from echipacgh e,saritorcgh s where e.salariu_antrenor=3002
and substr(e.num_echipa,-1)='a' and s.id echipa=e.id echipa and s.id_saritor in
(select id_saritor from accidentarecgh)));
```

The results window shows the output of the query, which is a list of sponsor names. The results are as follows:

DENUMIRE
1 4F
2 Intersport
3 Milka
4 Adidas
5 Audi

2. Sa se afiseze greutatea medie a saritorilor dintr-o echipa care participa la competitile 'Turneul celor 4 trambuline', 'Planica 3', 'Cupa mondiala'. Pentru fiecare competitive se vor lua in calcul doar saritorii care participa la aceasta, nu toata echipa. Se vor afisa doar echipele in care salariul antrenorului este mai mare de 2000. (group by, order by, filtrare la nivel de linii, nvl, decode)

select e.num_echipa,

nvl(avg(decode(p.id_competitie,12345,s.greutate)),0) as "Turneul celor 4 trambuline",

nvl(avg(decode(p.id_competitie,45678,s.greutate)),0) as "Planica 3",

nvl(avg(decode(p.id_competitie,23456,s.greutate)),0) as "Cupa mondiala"

```
select e.ume_echipa,
nvl(avg(decode( p.id_competitie,12345,s.greutate)),0) as "Turneul celor 4 trambuline",
nvl(avg(decode( p.id_competitie,45678,s.greutate)),0) as "Planica 3",
nvl(avg(decode( p.id_competitie,23456,s.greutate)),0) as "Cupa mondiala"
from saritorcgh s, participa_lacgh p, echipacgh e
where s.id_saritor=p.id_saritor and s.id_echipa=e.id_echipa
group by e.ume_echipa
having avg(salariu_antrenor)>2000
order by e.ume_echipa;
```

```

3. Sa se afiseze saritorii care participa la o competitie mai lunga de o luna si au mai multe puncta
decat media totala a punctelor.(with, functie pe data calenadristica, siruri de caractere(concat),join)

with val_medie as (select avg(puncte) medie from rezultatcgh)

select concat(concat(s.nume, ' '),s.prenume)

from saritorcgh s

join participa_lacgh p on(s.id_saritor=p.id_saritor)

join competitiecgh c on(c.id_competitie=p.id_competitie)

join rezultatcgh r on (s.id_saritor=r.id_saritor)

where months_between(c.data_final,c.data_inceput)>1 and r.puncte>(select medie from val_medie)

;

```

```
with val_medie as (select avg(puncte) medie from rezultatcgh)
```

from saritorcgh s

```
join competitivcgh c on(c.id_competitie=p.id_competitie)
```

```
where months_between(c.data_final,c.data_inceput)>1 and r.puncte>(select medie from val_medie)
;
```

```

with val_medie as (select avg(puncte) medie from rezultatcgh)
select distinct concat(concat(s.ume, ' '),s.prenume)
from saritorcgh s
join participa_lacgh p on(s.id_saritor=p.id_saritor)
join competitiecgh c on(c.id_competitie=p.id_competitie)
join rezultatcgh r on (s.id_saritor=r.id_saritor)
where months_between(c.data_final,c.data_inceput)>1 and r.puncte>(select medie from val_medie) ;

```

Query Result x

SQL | All Rows Fetched: 4 in 0.065 seconds

CONCAT(CONCAT(S.UME,""),S.PRENAME)
1 Tande Daniel-Andre
2 Granerud Halvor-Eqner
3 Eisenbichler Markus
4 Kos Lovro

4. Sa se afiseze echipele care au cel putin doi saritori care au cel putin 4 medalii.(cerere sincronizata)

Select e.ume_echipa

From echipacgh

Where (select count(*) from saritorcgh s where e.id_echipa=s.id_echipa and (select count(*) from medaliecgh where id_saritor=s.sariotr_id)>=4)>=2;

```

select e.ume_echipa
from echipacgh e
where (select count(*) from saritorcgh s where e.id_echipa=s.id_echipa and
(select count(*) from medaliecgh m where m.id_saritor=s.id_saritor )>=4)>=2;

```

Script Output x Query Result x

SQL | All Rows Fetched: 1 in 0.917 seconds

NUME_ECHIPA
1 Norvegia

5. Afisati echipa de care apartin saritorii de 29 de ani care s-au accidentat si ultima zi a lunii in care s-au accidentat acestia. (case, functie pe data calendaristica)

select e.id_echipa, last_day(a.data_accidentare)

from echipacgh e, saritorcgh s, accidentarecgh a

where s.id_echipa=e.id_echipa and s.id_saritor=a.id_saritor and s.id_saritor=case s.varsta when 29 then s.id_saritor else null end;

```

select e.id echipa, last_day(a.data_accidentare)
from echipacgh e, saritorcgh s, accidentarecgh a
where s.id echipa=e.id echipa and s.id_saritor=a.id_saritor and
s.id_saritor=case s.varsta when 29 then s.id_saritor else null end;

```

Script Output x Query Result x

SQL | All Rows Fetched: 2 in 0.815 seconds

ID_ECHIPA	LAST_DAY(A.DATA_ACCIDENTARE)
1 GER	31-MAR-21
2 SLO	31-JAN-22

SUPRIMARE ACTUALIZARE

--crestem salariu antrenor daca echipa are mai mult de 3 saritori

update echipacgh

set salariu_antrenor=salariu_antrenor*1.5

where id echipa in(select e.id echipa

from echipacgh e

where (select count(*) from saritorcgh s

where s.id echipa=e.id echipa)>=3);

cript Output x Query Result x

SQL | All Rows Fetched: 5 in 0.008 seconds

ID_ECHIPA	NUME_ECHIPA	ANTRENOR	SALARIU_ANTRENOR
1 NOV	Norvegia	Roar Ljokelsoy	2300
2 POL	Polonia	Stefan Hula	3002
3 GER	Germania	Stefan Hornqacher	5000
4 SLO	Slovenia	Riqa Helar	3002
5 AUT	Austria	Andreas Widhoelzl	4300

```

update echipacgh
set salariu_antrenor=salariu_antrenor*1.5
where id echipa in(select e.id echipa
                    from echipacgh e
                    where (select count(*) from saritorcgh s
                          where s.id echipa=e.id echipa)>=3);

select*from echipacgh;
rollback;

```

Script Output x Query Result x

SQL | All Rows Fetched: 5 in 0.01 seconds

ID_ECHIPA	NUME_ECHIPA	ANTRENOR	SALARIU_ANTRENOR
1 NOV	Norvegia	Roar Ljokelsoy	3450
2 POL	Polonia	Stefan Hula	3002
3 GER	Germania	Stefan Hornqacher	5000
4 SLO	Slovenia	Riqa Helar	4503
5 AUT	Austria	Andreas Widhoelzl	4300

--stergem rezultatele de la etapele de pe trambulina Berg Isel

delete

from rezultatcgh

where id_etapa in(select id_etapa

from etapacgh e, trambulinacgh t

where e.id_trambulina= t.id_trambulina and t.nume='Berg Isel');

```
select * from rezultatcgh;
```

ID_SARITOR	ID_ETAPA	ID_REZULTAT	PUNCTE	LUNGIME1	LUNGIME2
3	100	2	1002	212	142
4	101	2	1012	200	137
5	102	2	1022	210	143
6	100	3	1003	212	132
7	101	3	1013	200	117
8	401	3	4013	210	123
9	100	4	1004	212	132
10	201	4	2014	200	117
11	401	4	4014	210	123
12	100	5	1005	212	132
13	301	5	3015	220	127
14	201	5	2015	210	123
15	100	6	1006	212	132
16	301	6	3016	230	137
17	501	6	5016	205	133

```
delete
from rezultatcgh
where id_etapa in(select id_etapa
from etapacgh e, trambulinacgh t
where e.id_trambulina= t.id_trambulina and t.nume='Berg Isel');

rollback;
select * from rezultatcgh;
```

ID_SARITOR	ID_ETAPA	ID_REZULTAT	PUNCTE	LUNGIME1	LUNGIME2
1	101	1	1011	220	150
2	102	1	1021	210	143
3	100	2	1002	212	142
4	101	2	1012	200	137
5	102	2	1022	210	143
6	100	3	1003	212	132
7	101	3	1013	200	117
8	401	3	4013	210	123
9	100	6	1006	212	132
10	301	6	3016	230	137
11	501	6	5016	205	133
12	100	7	1007	212	132
13	401	7	4017	230	137
14	501	7	5017	205	133
15	100	8	1008	270	222

--sa se creasca varsta saritorilor care au greutatea mai mare de 60kg si care s-au accidentat

update saritorcgh

set varsta=varsta+1

where greutate>60 and id_saritor in

(select id_saritor from accidentarecgh);

ID_SARITOR	NUME	PRENUME	GEN	VARSTA	GREUTATE	TIP	ID_ECHIPA
1	100 Tande	Daniel-Andre	m	28	65.5	1 NOV	
2	101 Granerud	Halvor-Eqner	m	25	67	1 NOV	
3	102 Forfanq	Johann	m	26	72.5	1 NOV	
4	201 Stoch	Kamil	m	34	55	1 POL	
5	301 Eisenbichler	Markus	m	31	59	1 GER	
6	302 Geiger	Karl	m	29	70.3	1 GER	
7	401 Kos	Lovro	m	22	62	1 SLO	
8	402 Prevc	Peter	m	29	64	1 SLO	
9	403 Prevc	Cene	m	26	63.5	1 SLO	
10	404 Prevc	Domen	m	22	61	1 SLO	
11	501 Kraft	Stefan	m	29	56	1 AUT	

```

update saritorcgh
set varsta=varsta+1
where greutate>60 and id_saritor in
(select id_saritor from accidentarecgh);

```

ID_SARITOR	NUME	PRENUME	GEN	VARSTA	GREUTATE	TIP	ID_ECHIPA
1	100 Tande	Daniel-Andre	m	29	65.5	1 NOV	
2	101 Granerud	Halvor-Eqner	m	26	67	1 NOV	
3	102 Forfanq	Johann	m	27	72.5	1 NOV	
4	201 Stoch	Kamil	m	34	55	1 POL	
5	301 Eisenbichler	Markus	m	31	59	1 GER	
6	302 Geiger	Karl	m	30	70.3	1 GER	
7	401 Kos	Lovro	m	22	62	1 SLO	
8	402 Prevc	Peter	m	30	64	1 SLO	
9	403 Prevc	Cene	m	26	63.5	1 SLO	
10	404 Prevc	Domen	m	22	61	1 SLO	
11	501 Kraft	Stefan	m	29	56	1 AUT	

OUTER JOIN

select s.num, s.prenume c.num, e.data_etapa

from saritorcgh s

left outer join participa_lacgh p on(s.id_saritor=p.id_saritor)

right outer join competitiecgh c on(p.id_competitie=c.id_competitie)

right outer join etapacgh e on (p.id_etapa=e.id_etapa);

```

select s.num, s.prenume, c.num, e.data_etapa
from saritorcgh s
left outer join participa_lacgh p on(s.id_saritor=p.id_saritor)
right outer join competitiecgh c on(p.id_competitie=c.id_competitie)
right outer join etapacgh e on (p.id_etapa=e.id_etapa);

```

NUME	PRENUME	NUME_1	DATA_ETAPA
250 Eisenbichler	Markus	Zbor cu schiurile	02-DEC-22
251 Eisenbichler	Markus	Zbor cu schiurile	05-DEC-22
252 Geiger	Karl	Zbor cu schiurile	02-DEC-22
253 Geiger	Karl	Zbor cu schiurile	05-DEC-22
254 Kos	Lovro	Zbor cu schiurile	02-DEC-22
255 Kos	Lovro	Zbor cu schiurile	05-DEC-22
256 Prevc	Peter	Zbor cu schiurile	02-DEC-22
257 Prevc	Peter	Zbor cu schiurile	05-DEC-22
258 Prevc	Cene	Zbor cu schiurile	02-DEC-22
259 Prevc	Cene	Zbor cu schiurile	05-DEC-22
260 Prevc	Domen	Zbor cu schiurile	02-DEC-22
261 Prevc	Domen	Zbor cu schiurile	05-DEC-22
262 Kraft	Stefan	Zbor cu schiurile	02-DEC-22
263 Kraft	Stefan	Zbor cu schiurile	05-DEC-22
264 (null)	(null)	(null)	16-MAR-23
265 (null)	(null)	(null)	10-MAR-23
266 (null)	(null)	(null)	13-MAR-23
267 (null)	(null)	(null)	09-MAR-23
268 (null)	(null)	(null)	17-MAR-23

DIVISION

--numele saritorilor care participa la toate competitile in afara de planica 3 si numarul acestora

select s.num, count(distinct p.id_competitie)

from participa_lacgh p, saritorcgh s

where s.id_saritor=p.id_saritor and p.id_competitie in(select id_competitie from competitiecgh where nume!='Planica 3')

group by s.num

having count(distinct p.id_competitie)=(select count(*) from competitiecgh where nume!='Planica 3');

```

select s.num, count(distinct p.id_competitie)
from participa_lacgh p, saritorcgh s
where s.id_saritor=p.id_saritor and p.id_competitie in(select id_competitie from competitiecgh where nume!='Planica 3')
group by s.num
having count(distinct p.id_competitie)=(select count(*) from competitiecgh where nume!='Planica 3');

```

NUME	COUNT(DISTINCTP.ID_COMPETITIE)
1 Forfang	4
2 Granerud	4
3 Tande	4

--echipele care au toti sponsorii norvegiei

select e.ume_echipa

from echipacgh e, sponsorizeazacgh s

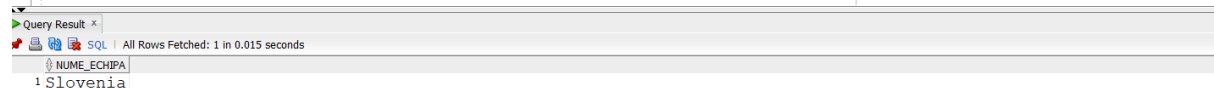
where e.id_echipa=s.id_echipa and s.id_sponsor in (select id_sponsor from sponsorizeazacgh where id_echipa='NOV')

and e.id_echipa!='NOV'

group by e.ume_echipa

having count (id_sponsor)=(select count(*) from sponsorizeazacgh where id_echipa='NOV');

```
select e.ume_echipa
from echipacgh e, sponsorizeazacgh s
where e.id_echipa=s.id_echipa and s.id_sponsor in (select id_sponsor from sponsorizeazacgh where id_echipa='NOV')
and e.id_echipa!='NOV'
group by e.ume_echipa
having count (id_sponsor)=(select count(*) from sponsorizeazacgh where id_echipa='NOV');
```



UME_ECHIPA
Slovenia

BCNF

Def. O relatie se afla in forma normala BCNF daca si numai daca fiecare determinant este o cheie candidat.

Pentru exemplificarea normalizarii de tip BCNF se va presupune ca atributul descriere_pieptar din participa_la este unic si determina functional atributul #id_etapa care face parte din cheia primara.

Astfel, intee attributele relatiei participa_la (id_sportiv#, id_competitie#, id_etapa#, descriere_pieptar) exista dependentele:

{descriere_pieptar}->{id_etapa#}

{ id_sportiv#, id_competitie#, id_etapa#, }->{descriere_pieptar}

Se poate remarca faptul ca aributul descriere_pieptar este o cheie candidat si cheia id_etapa depinde de atributul descriere_pieptar.

Aplicam regula Casey-Delobel

Participa_la1 (descriere_pieptar#, id_etapa)

Participa_la2 (id_sportiv#, id_competitie#, descriere_pieptar)

FORMA NORMALA 4

Def. O relatie se afla in FN4 daca si numai daca se afla in BCNF si nu continue relatii m:n independente.

Fie relatie ECHIPA(id echipa#, id_saritor#, id_sponsor#) si presupunem ca o echipa poate avea mai multi sponsori si mai multi saritori.

Intre attributele relatiei exista multidependentele:

Id echipa# → id_saritor

Id echipa# → id_sponsor

Relatia ECHIPA este in BCNF. Pentru a aduce relatia in FN4 o vom descompune prin proiectie in doua relatii:

ECHIPA_1(id echipa#, id_sponsor#), ECHIPA_2(id echipa#, id_saritor#)

OPTIMIZAREA UNEI CERERI

Cererea aleasa:

```
select concat(concat(s.nume, ' '), s.prenume)
from saritorcgh s
join participa_lacgh p on (s.id_saritor=p.id_saritor)
join competitiecgh c on (c.id_competitie=p.id_competitie)
join rezultatcgh r on (s.id_saritor=r.id_saritor)
where months_between(c.data_final, c.data_inceput) > 1 and r.puncte > 100;
```

R1=SELECT(rezultat, puncte > 100)

R2=PROJECT(R1, id_saritor)

R3=PROJECT(SARITOR, nume, prenume, id_saritor)

R4=JOIN(R2, R3, id_saritor)

R5=SELECT(competitie, months_between(c.data_final, c.data_inceput) > 1)

R6=PROJECT(R5, id_competitie)

R7=PROJECT(PARTICIPA_LA, id_competitie, id_saritor)

R8=JOIN(R6, R7, id_competitie)

R9=JOIN(R3, R8, id_saritor)

R10=PROJECT(R9, nume, prenume)

Cererea este una optima, deoarece cand a fost create s-a tinut cont de urmatoarele reguli de optimizare:

- Selectiile se executa cat mai devreme posibil
- Produsele carteziene se inlocuiesc cu join-uri
- Proiectiile se executa la inceput pt a indeparta attributele nefolositoare

Proprietatile utilizate sunt :

- Comutarea selectiei cu proiectia => mai intai aplicam selectie pt a elimina datele care nu sunt necesare si apoi aplicam project pt a lua coloanele dorite
- Comutarea proiectiei cu operatia join => prima oara se face proiectia si dupa join-ul pt a nu trebui sa se faca join pe toate coloanele din relatie

