

PROIECT BAZE DE DATE

1. Descrierea modelului real, a utilității acestuia și a regulilor de funcționare.

Modelul de date va furniza informații despre modul în care decurg prezentările în cadrul evenimentului ”Fashion week”, cât și detalii ce țin de brandurile care iau parte și defilări organizate, precum ce ținută este prezentată, de care fotomodel este aceasta prezentă, dar și în cadrul cărui eveniment. Modelul de date se aplică pentru prezentările din lumea modei și este util pentru organizatori, deoarece trebuie să aibe o astfel de evidență.

Fashion week este un eveniment mult așteptat în lumea modei, acesta se organizează în orașele unei țări.

La un astfel de eveniment iau parte brand-uri care își expun colecțiile. O colecție este formată din mai multe ținute care sunt purtate de fotomodele în timpul defilărilor.

Participanții la fashion week își rezerva locuri, care pot fi simple (un loc oarecare), sau vip care se deosebește de cele simple prin faptul că află în proximitatea podiumului, astfel participatul are poziție mai bună.

De asemenea, la un astfel de eveniment ia parte și presa care promovează evenimentul la posturi de televiziune.

2. Prezentarea constrângerilor (restricții, reguli) impuse asupra modelului.

- Presa promovează evenimentul Fashion week la mai multe posturi de televiziune
- Un fotomodel poate purta mai multe ținute din cadrul mai multor colecții

- Un fotomodel prezintă mai multe ținute în cadrul mai multor defilări
- Un fotomodel este de o anumită naționalitate, iar o naționalitate poate fi avută de mai multe fotomodele
- Un participant poate cumpăra ținuta care îi place, sau chiar mai multe ținute, sau chiar nici una.
- Aceeași ținută poate fi cumpărată de mai mulți participanți.
- Un participant ia parte la mai multe evenimente fashion week (cel puțin la unul, dar nu neapărat la toate).

3. Descrierea entităților, incluzând precizarea cheii primare.

ENTITATE	CHEIE PRIMARĂ	OBSERVAȚII
Post_tv	id_post	Informații despre posturile tv care promoveaza evenimentul
Presa	id_jurnalist	Informații despre presa care se ocupa cu promovarea
Fashion week	id_eveniment	Informații generale despre desfasurarea evenimentului “Fashion week”
Oras	id_oras	Informații despre orsul unde se desfasoara fashion week
Tara	id_tara	Informații despre tara unde se desfasoara fashion week
Participanti	id_participant	Informații despre persoanele care iau parte la eveniment
Locuri	id_loc	Informații despre locuri

Locuri_vip	id_loc	Informații specific despre locurile vip, subentitate a entitatii locuri
Brand	id_brand	Informații despre brandurile care isi expun colectiile la fashion week
Colectii	id_colectie	Informații despre colectiile prezentate de brand-uri
Tinute	id_tinuta	Informații despre fiecare tinuta care este prezentata
Defilare	id_defilare	Informații generale despre defilarile din cadrul fashion week
Fotomodel	id_fotomodel	Informații generale despre fotomodelele care prezinta tinutele in cadrul defilarilor
Nationalitate	id_nationalitate	informații despre nationalitatea fiecarui fotomodel

4. Descrierea relațiilor, incluzând precizarea cardinalității acestora.

RELATIE	CARDINALITATE	OBSERVAȚII
promovează	Post_tv (fashion_week, presa) one-to-many Fashion_week (post_tv, presa) one-to-many Presa (fashion_week, post_tv,) one-to-many	Presa promovează evenimentul fashion week la mai multe posturi tv
are loc	Oras – Fashion week one-to-many	Fashion week poate are loc intr-un oraș Intru-un oraș pot avea loc mai multe evenimente fashion week

se află	Oras – Tara one-to-many	Un oraș se află într-o țară Într-o țară se află mai multe orașe
participă	Fashion week – Participanti many-to-many	La un fashion week participă mai mulți participanți Un participant poate lua parte la mai multe evenimente fashion week
rezervă	Participanți – Locuri one-to-many	Un participant poate rezerva mai multe locuri Un loc poate fi rezervat de un singur participant
cumpără	Participanți – Ținute many-to-many	Un participant poate cumpăra mai multe ținute O ținuta poate fi cumpărată de mai mulți participanți
iau parte	Brand – Fashion week many-to-many	La un fashion week pot lua parte mai multe brand-uri Un brand poate lua parte la mai multe evenimente fashion week
expun	Brand – Colecții one-to-many	Un brand poate expune mai multe colecții, cel puțin una Aceași colecție nu poate fi conținută de mai multe brand-uri
conțin	Ținute – Colecții one-to-many	O colecție conține mai multe ținute, cel puțin una Aceași ținuta nu poate fi conținută de mai multe colecții
e format din	Defilare – Fashion week one-to-many	Un fashion week e format din mai multe defilări Aceași defilare nu poate fi la mai multe evenimente fashion week

prezintă	<p>Ținute (Fotomodele, Defilări) one-to-many</p> <p>Defilări (Fotomodele, Ținute) one-to-many</p> <p>Fotomodele (Defilări, Ținute) one-to-many</p>	Un fotomodel poartă mai multe ținute în cadrul mai multor defilări
au	<p>Fotomodele –</p> <p>Nationalitate one-to-many</p>	<p>Un fotomodel are o singura naționalitate</p> <p>O naționalitate poate fi avută de mai multe fotomodel sau nici unul</p>

5. Descrierea atributelor, incluzând tipul de date și eventualele constrângeri, valori implicite, valori posibile ale atributelor.

ENTITATE: FASHION WEEK

ATRIBUT	TIP	DIMENSIUNE /PRECIZIE	VALORI POSIBILE/DEFAULT	OBSERVATII
Data_inceput	data			
Data_final	data			

ENTITATE: BRAND

ATRIBUT	TIP	DIMENSIUNE /PRECIZIE	VALORI POSIBILE/DEFAULT	OBSERVATII
Denumire	string	20		
sediu	string	20		

ENTITATE: CUMPARA

ATRIBUT	TIP	DIMENSIUNE /PRECIZIE	VALORI POSIBILE/DEFAULT	OBSERVATII
marime	string	2	Poate fi doar {'XS', 'S', 'M', 'L', 'XL'}	
discount	number		Poate fi 'null' in cazul cand nu se aplica vreo reducere	

ENTITATE: LOCURI

ATRIBUT	TIP	DIMENSIUNE /PRECIZIE	VALORI POSIBILE/DEFAULT	OBSERVATII
pret	Number			
disponibilitate	number	1	Poate fi 0, 1 sau null	

ENTITATE: LOC_VIP

ATRIBUT	TIP	DIMENSIUNE /PRECIZIE	VALORI POSIBILE/DEFAULT	OBSERVATII
pret	Number			
disponibilitate	number	1	Poate fi 0, 1 sau null	
pozitie	string	20	Are valoarea 'zona 0'	

ENTITATE: FOTOMODEL

ATRIBUT	TIP	DIMENSIUNE /PRECIZIE	VALORI POSIBILE/DEFAULT	OBSERVATII
Nume	string	25		
prenume	string	25		

Data_nastere	data			
salariu	number			

ENTITATE: PARTICIPANTI

ATRIBUT	TIP	DIMENSIUNE /PRECIZIE	VALORI POSIBILE/DEFAULT	OBSERVATII
nume	string			
prenume	string			

ENTITATE: TINUTE

ATRIBUT	TIP	DIMENSIUNE /PRECIZIE	VALORI POSIBILE/DEFAULT	OBSERVATII
material	string			
pret	number			

ENTITATE: PARTICIPA

ATRIBUT	TIP	DIMENSIUNE /PRECIZIE	VALORI POSIBILE/DEFAULT	OBSERVATII
discount	number	double	Poate fi 'null' in cazul cand nu se aplica vreo reducere	Discount care se aplica la taxa de participare

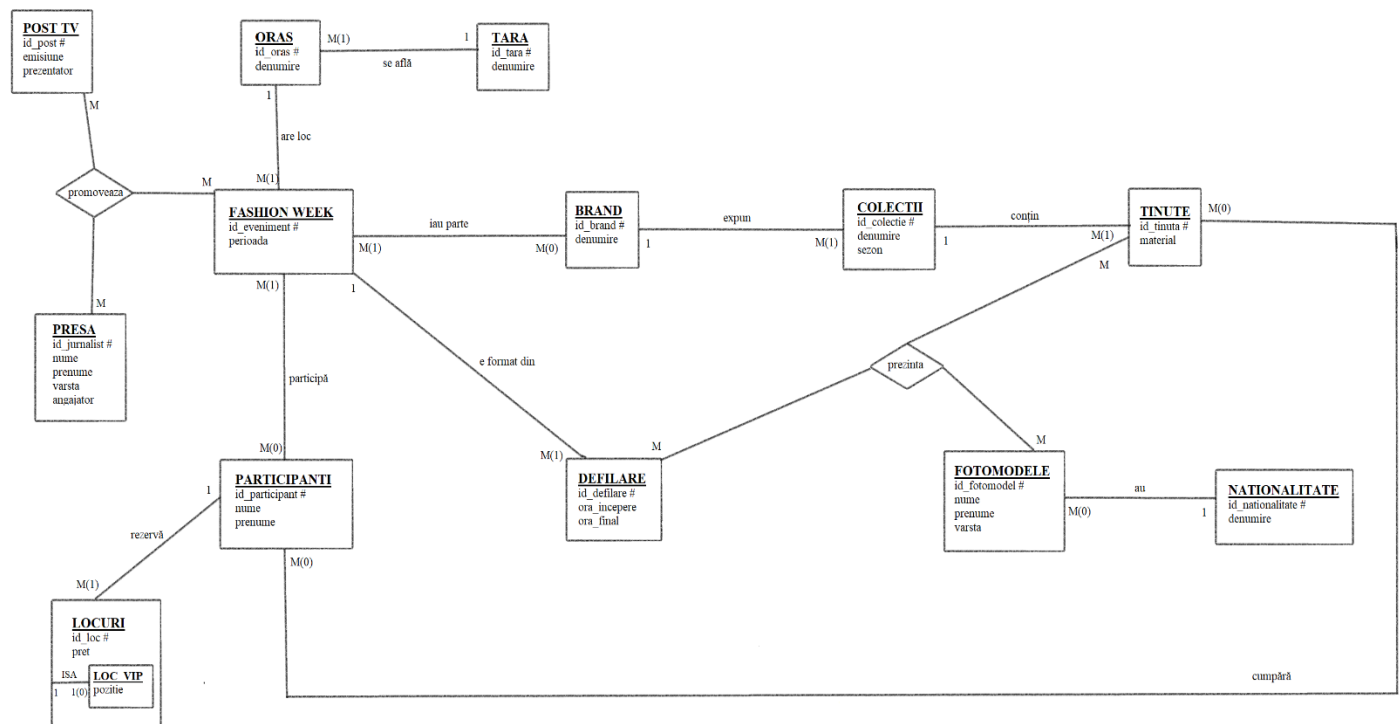
ENTITATE: IA_PARTE

ATRIBUT	TIP	DIMENSIUNE /PRECIZIE	VALORI POSIBILE/DEFAULT	OBSERVATII
Taxa_participare	number			

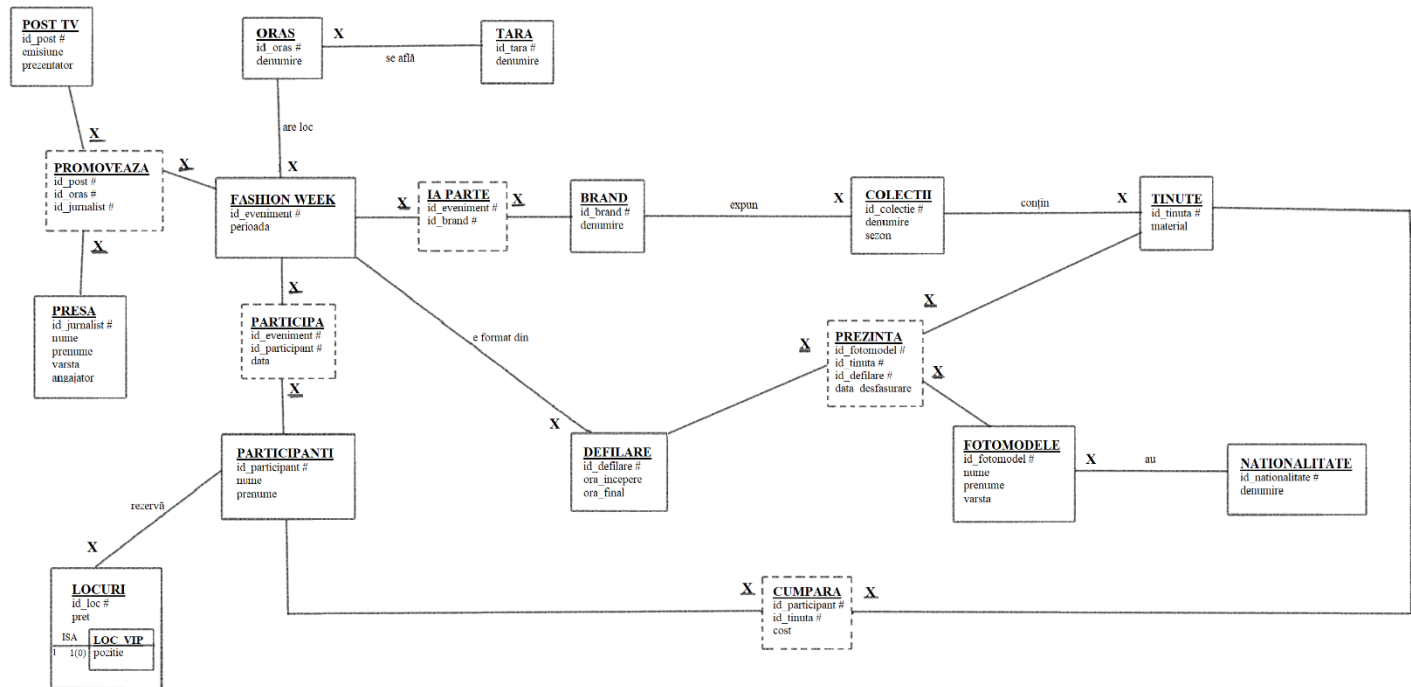
ENTITATE: COLECTIE

ATRIBUT	TIP	DIMENSIUNE /PRECIZIE	VALORI POSIBILE/DEFAULT	OBSERVATII
temarica	string	25		
sezon	string	25	De preferat ‘toamna- iarna’ sau ‘primavara- vara’	

6. Realizarea diagramei entitate-relație corespunzătoare descrierii de la punctele 3-5.



7. Realizarea diagramei conceptuale corespunzătoare diagramei entitate-relație proiectate la punctul 6.



8. Enumerarea schemelor relaționale corespunzătoare diagramei conceptuale proiectate la punctul 7.

FASHION WEEK (id_eveniment #, data_inceput, data_final, id_oras)

ORAS (id_oras #, denumire, id_tara)

TARA (id_tara, denumire)

BRAND (id_brand #, denumire, sediu)

IA PARTE (id_eveniment #, id_brand #, taxa_participare)

COLECTII (id_colectie #, sezon, tematica, id_brand)

TINUTE (id_tinuta #, material, pret, id_colectie)

DEFILARE (id_defilare #, ora_inceput, ora_final, id_eveniment)

FOTOMODEL (id_fotomodel #, nume, prenume, varsta, id_nationalitate, salariu)

ISTORIC DEFILARI (id_defilare #, id_fotomodel, id_defilare, data_desfasurare)

PREZINTA (id_fotomodel #, id_tinuta #, id_defilare #, data_desfasurare)

NATIONALITATE (id_nationalitate, denumire)

PARTICIPA (id_eveniment #, id_participant #, discount)

PARTICIPANT (id_participant #, nume, prenume)

LOCURI (id_loc #, pret, disponibilitate(bool) id_participant)

LOC_VIP (id_loc #, pret, disponibilitate, id_participant, pozitie)

CUMPARA (id_participant #, id_tinuta #, marime_tinuta, discount)

POST TV (id_post #, denumire, emisiune)

PRESA (id_jurnalist #, nume, prenume, angajator)

PROMOVEAZA (id_eveniment #, id_post #, id_jurnalist #, durata(min), vizualizari)

9. Realizarea normalizării până la forma normală 3 (FN1-FN3).

FN1 (FORMA NORMALA 1)

Def: O relație se află în FN1, dacă fiecărui atribut care o compune îi corespunde o valoare indivizibila. O relație se află în FN1 dacă există un identificator unic.

Pentru a demostra că relațiile avute în diagram sunt în FN1 vom da câteva exemple.

Exemplu 1:

În relația **POST_TV** avem cheia primară, *id_post*, ceea ce înseamnă că avem un identificator unic. În acest fel, atributul *id_post* fiind o valoare indivizibilă, relația se află în FN1.

Exemplu 2:

Un alt exemplu este relația **PROMOVEAZA** care are o cheie primară compusă formată din: *id_post*, *id_jurnalist* și *id_eveniment*. Această cheie primară este un identificator unic pentru relația mai sus menționată. În acest fel, attributele *id_post*, *id_jurnalist* și *id_eveniment* fiind împreună o valoare indivizibilă, relația se află în formă normal 1 (există identificator unic pentru toate intrările din table)

Exemple 3:

Pentru a exemplifica cum se aplică această normalizare se va crea un exemplu imaginar, mai exact vom presupune că pentru participarea la un fashion week există mai multe tipuri de taxe de participare (*taxa_participare*)

Așa arată inițial *ia_parte_nonFN1*

id_eveniment#	id_brand#	taxa_participare
101	100	200
203	200	200, 250
304	400	150, 100, 200

Astfel, se poate remarca faptul că relația de mai sus nu este în FN1, deoarece atributului *taxa_participare* nu îi corespunde o valoare indivizibilă.

După ce aplicăm FN1, relația va fi următoarea

id_eveniment#	id_brand#	taxa_participare #
101	100	200
203	200	200
203	200	250
304	400	150
304	400	100
304	400	200

Acum, se poate observa că după normalizarea fiecărui atribut care compune relația îi corespunde o valoare indivizibilă.

FN2 (FORMA NORMALA 2)

Def: O relație se află în FN2, dacă și numai dacă se află în prima formă normal (FN1) și dacă fiecare atribut care nu face parte din cheie e dependent de întreaga cheie primară.

Pentru a exemplifica această formă normală se va arăta un exemplu care pornește de la o relație care inițial nu era în FN2 și se va transforma într-o relație din forma normal

Inițial relația era ca mai jos:

colectii_si_tinute

Id_tinuta#	material	pret	id_colectie#	tematica
10	casmir	500	1	Space theme
11	lână	180	1	Space theme

12	voal	2700	7	Indian style
13	pânză	3000	5	Underwater escapism
14	mătase	1750	2	Escalator runway
18	în	5000	5	Underwater escapism

Relația de mai sus se află în forma normală 1, deoarece există identificator unic pentru toate intrările din table.

Pentru a fi în FN2, atributul *tematică*, trebuie să depindă de întreaga cheie primară: *id_tinuta* și *id_colecție*, dar se poate observa că acestea nu depinde direct de întreaga cheia primară. Acest fapt se explică prin existența dependenței directe dintre *id_colecție* și *tematică*, de aceea relația nu se află în FN2.

Dependențele sunt:

{id_colecție#} -> {tematica}

{id_tinuta#, id_colecție} -> {material, pret}

Transformarea în FN2 va rezulta prin împărțirea relației *colecții_si_tinute* în două relații după cum urmează:

Relația 1: *colecții*

Id_colecție#	tematica
1	Space theme
2	Escalator runway
5	Underwater escapism
7	Indian style

Relatia 2: *tinute*

Id_tinuta#	material	pret	id_colectie#
10	casmir	500	1
11	lână	180	1
12	voal	2700	7
13	pânză	3000	5
14	mătase	1750	2
18	in	5000	5

FN3 (FORMA NORMALA 3)

Def: O relație se află în FN3, dacă și numai dacă se află în a doua formă normal (FN2) și dacă fiecare atribut care nu e cheie depinde direct de cheia primară și numai de ea.

Pentru exemplificare acestei forme normale am decis să realizez un exemplu fictiv pentru relația **FOTOMODEL**, vom presupune că atributul *salariu* depinde de vârsta fotomodelului (calculată după *data_nasteri*), ceea ce nu adevărat în cazul diagramei construite.

Astfel avem relația ***fotomodel_nonFN3***

id_fotomodel#	nume	prenume	data_nasterii	salariu	id_nationalitate
20	Rodriguez	Sasha	01-01-2000	17000	3
21	Smith	Bella	15-10-1995	25000	1

22	Marshal	Martha	03-04-2000	17000	1
23	Jones	Suellen	09-04-1997	21000	2
24	Kennedy	Melanie	20-06-1995	25000	1

Această relație este în prima formă, deoarece există identificator unic pentru toate intrările din tabel, anume *id_fotomodel#*.

De asemenea, relația se află și în FN2, pentru că se poate remarca faptul că attributele: *nume*, *prenume*, *data_nastere*, *salariu* care nu sunt în componența cheii primare sunt dependente de întreaga cheie primară *id_fotomodel#*

În acest exemplu se observă că atributul *salariu* depinde tranzitiv de cheia primară *id_fotomodel#* prin intermediu atributului *data_nastere*.

Astfel, putem spune că relația nu se află în forma normală 3.

Dependențele sunt:

$\{id_fotomodel\# \} \rightarrow \{nume, prenume, data_naster\}$

$\{id_fotomodel\# \} \rightarrow \{data_nastere\} \rightarrow \{salariu\}$

Transformarea în FN3 va rezulta prin împărțirea relației *fotomodel_nonFN3* după cum urmează:

Relatie 1: *fotomodel_FN3*

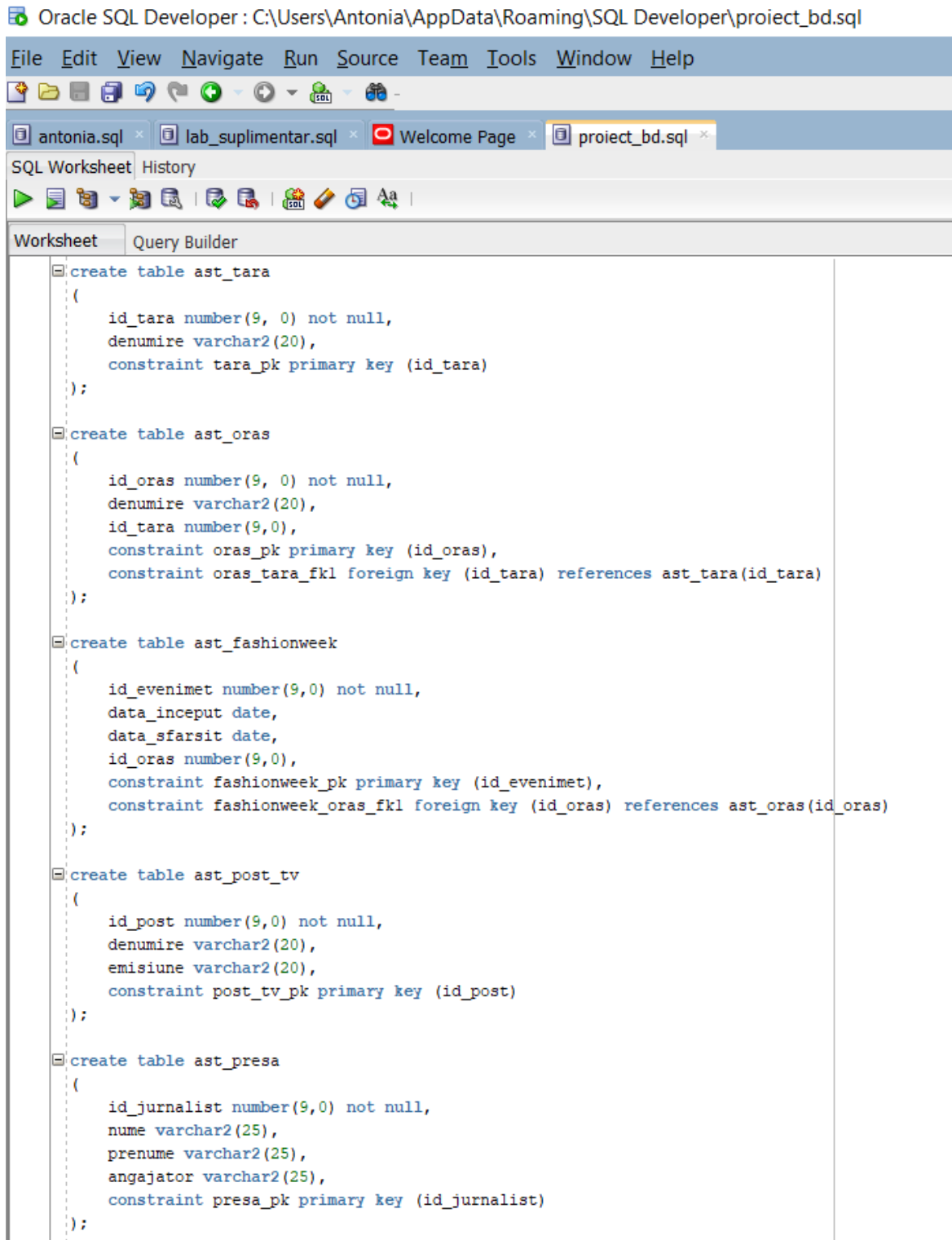
id_fotomodel#	nume	prenume	data_nasterii	id_nationalitate
20	Rodriguez	Sasha	01-01-2000	3
21	Smith	Bella	15-10-1995	1
22	Marshal	Martha	03-04-2000	1
23	Jones	Suellen	09-04-1997	2
24	Kennedy	Melanie	20-06-1995	1

Relatie 2: *data_nasteri_salariu*

data_nasterii	salariu
01-01-2000	17000
15-10-1995	25000
03-04-2000	17000
09-04-1997	21000
20-06-1995	25000

Acum, relația *fotomodel_FN3* se află în forma normală 3, deoarece fiecare atribut: *nume*, *prenume*, *data_nastere* sunt dependente de întreaga cheie primară *id_fotomodel#*, fără a mai fi dependenete între ele.

10. Crearea tabelelor în SQL și inserarea de date coerente în fiecare dintre acestea (minimum 5 înregistrări în fiecare tabel neasociativ; minimum 10 înregistrări în tabelele asociative).



Oracle SQL Developer : C:\Users\Antonia\AppData\Roaming\SQL Developer\proiect_bd.sql

File Edit View Navigate Run Source Team Tools Window Help

antonia.sql lab_suplimentar.sql Welcome Page proiect_bd.sql

SQL Worksheet History

Worksheet Query Builder

```
create table ast_tara
(
    id_tara number(9, 0) not null,
    denumire varchar2(20),
    constraint tara_pk primary key (id_tara)
);

create table ast_oras
(
    id_oras number(9, 0) not null,
    denumire varchar2(20),
    id_tara number(9,0),
    constraint oras_pk primary key (id_oras),
    constraint oras_tara_fkl foreign key (id_tara) references ast_tara(id_tara)
);

create table ast_fashionweek
(
    id_evenimet number(9,0) not null,
    data_inceput date,
    data_sfarsit date,
    id_oras number(9,0),
    constraint fashionweek_pk primary key (id_evenimet),
    constraint fashionweek_oras_fkl foreign key (id_oras) references ast_oras(id_oras)
);

create table ast_post_tv
(
    id_post number(9,0) not null,
    denumire varchar2(20),
    emisiune varchar2(20),
    constraint post_tv_pk primary key (id_post)
);

create table ast_presa
(
    id_jurnalist number(9,0) not null,
    nume varchar2(25),
    prenume varchar2(25),
    angajator varchar2(25),
    constraint presa_pk primary key (id_jurnalist)
);
```

Oracle SQL Developer : C:\Users\Antonia\AppData\Roaming\SQL Developer\proiect_bd.sql

File Edit View Navigate Run Source Team Tools Window Help

antonia.sql lab_suplimentar.sql Welcome Page proiect_bd.sql

SQL Worksheet History

Worksheet Query Builder

```
create table ast_promoveaza
(
    id_eveniment number(9,0) not null,
    id_post number(9,0) not null,
    id_jurnalist number(9,0) not null,
    durata number(3),
    vizualizari number(9),
    constraint promoveaza_pk primary key (id_eveniment, id_post, id_jurnalist)
);

alter table ast_promoveaza
add constraint promoveaza_fashionweek_fkl foreign key (id_eveniment) references ast_fashionweek(id_evenimet);

alter table ast_promoveaza
add constraint promoveaza_post_tv_fkl foreign key (id_post) references ast_post_tv(id_post);

alter table ast_promoveaza
add constraint promoveaza_presa_fkl foreign key (id_jurnalist) references ast_presa(id_jurnalist);

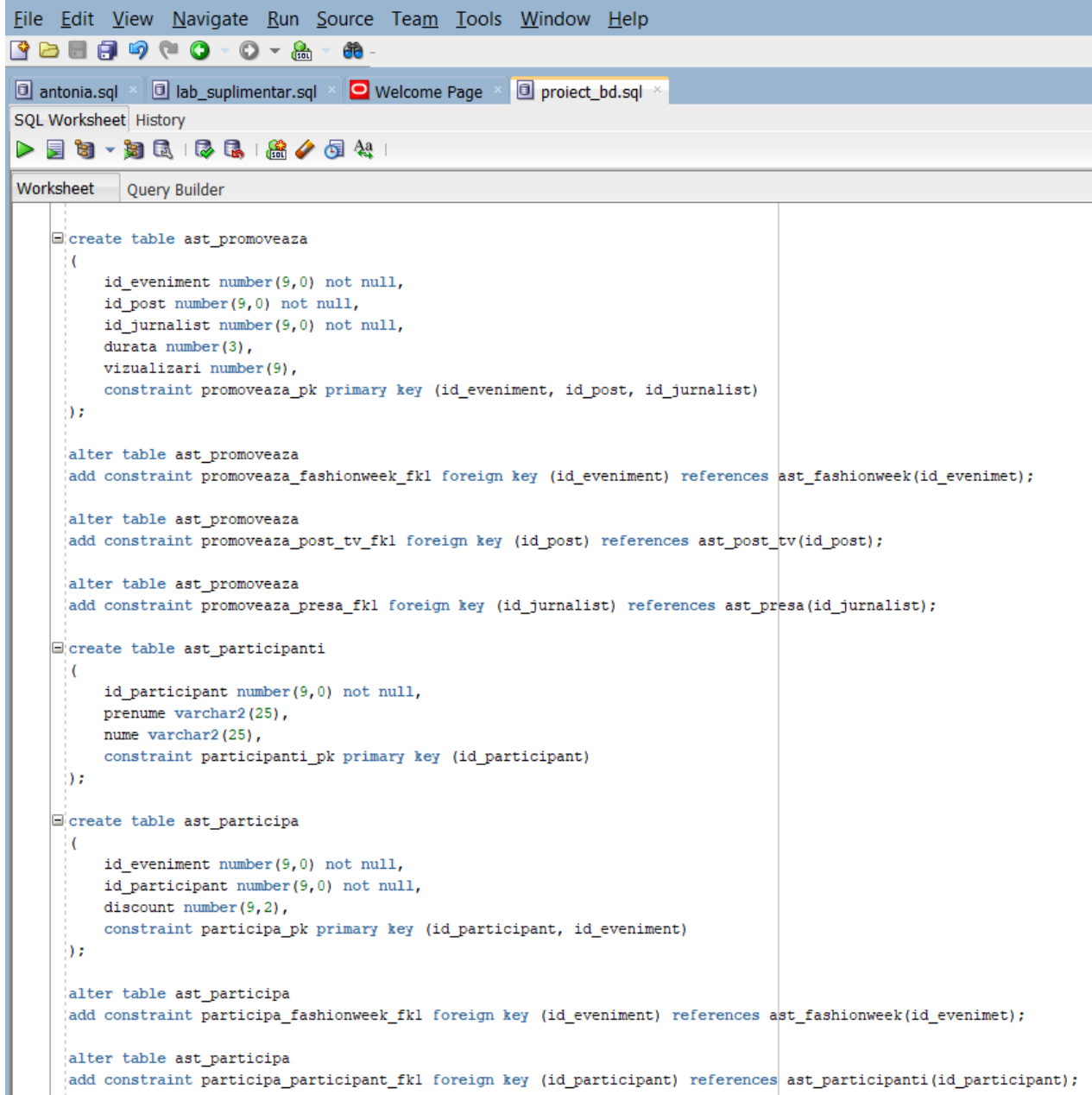
create table ast_participanti
(
    id_participant number(9,0) not null,
    prenume varchar2(25),
    nume varchar2(25),
    constraint participanti_pk primary key (id_participant)
);

create table ast_participa
(
    id_eveniment number(9,0) not null,
    id_participant number(9,0) not null,
    discount number(9,2),
    constraint participa_pk primary key (id_participant, id_eveniment)
);

alter table ast_participa
add constraint participa_fashionweek_fkl foreign key (id_eveniment) references ast_fashionweek(id_evenimet);

alter table ast_participa
add constraint participa_participant_fkl foreign key (id_participant) references ast_participanti(id_participant);
```

Oracle SQL Developer : C:\Users\Antonia\AppData\Roaming\SQL Developer\proiect_bd.sql



The screenshot shows the Oracle SQL Developer interface. The title bar indicates the file path: C:\Users\Antonia\AppData\Roaming\SQL Developer\proiect_bd.sql. The menu bar includes File, Edit, View, Navigate, Run, Source, Team, Tools, Window, and Help. The toolbar contains icons for file operations, execution, and formatting. The 'Worksheet' tab is active, displaying the following SQL code:

```
create table ast_promoveaza
(
    id_eveniment number(9,0) not null,
    id_post number(9,0) not null,
    id_jurnalist number(9,0) not null,
    durata number(3),
    vizualizari number(9),
    constraint promoveaza_pk primary key (id_eveniment, id_post, id_jurnalist)
);

alter table ast_promoveaza
add constraint promoveaza_fashionweek_fkl foreign key (id_eveniment) references ast_fashionweek(id_evenimet);

alter table ast_promoveaza
add constraint promoveaza_post_tv_fkl foreign key (id_post) references ast_post_tv(id_post);

alter table ast_promoveaza
add constraint promoveaza_presa_fkl foreign key (id_jurnalist) references ast_presa(id_jurnalist);

create table ast_participanti
(
    id_participant number(9,0) not null,
    prenume varchar2(25),
    nume varchar2(25),
    constraint participanti_pk primary key (id_participant)
);

create table ast_participa
(
    id_eveniment number(9,0) not null,
    id_participant number(9,0) not null,
    discount number(9,2),
    constraint participa_pk primary key (id_participant, id_eveniment)
);

alter table ast_participa
add constraint participa_fashionweek_fkl foreign key (id_eveniment) references ast_fashionweek(id_evenimet);

alter table ast_participa
add constraint participa_participant_fkl foreign key (id_participant) references ast_participanti(id_participant);
```

File Edit View Navigate Run Source Team Tools Window Help

antonia.sql lab_suplimentar.sql Welcome Page proiect_bd.sql

SQL Worksheet History

Worksheet Query Builder

```

create table ast_locuri
(
    id_loc number(9,0) not null,
    id_participant number(9,0),
    pret number(9) not null,
    disponibilitate number(1),
    constraint disponibilitate_ck check ( disponibilitate in (1,0) ),
    constraint locuri_pk primary key (id_loc),
    constraint locuri_participanti_fkl foreign key (id_participant) references ast_participanti(id_participant)
);

create table ast_loc_vip
(
    id_loc number(9,0) not null,
    id_participant number(9,0),
    pret number(9) not null,
    disponibilitate number(1),
    pozitie varchar2(10),
    constraint disponibilitate_vip_ck check ( disponibilitate in (1,0) ),
    constraint loc_vip_pk primary key (id_loc),
    constraint loc_vip_participanti_fkl foreign key (id_participant) references ast_participanti(id_participant)
);

create table ast_defilare
(
    id_defilare number(9,0) not null,
    id_eveniment number(9,0) not null,
    ora_inceput varchar2(10),
    ora_final varchar2(10),
    constraint defilare_pk primary key (id_defilare),
    constraint defilare_fashionweek_fkl foreign key (id_eveniment) references ast_fashionweek(id_evenimet)
);

create table ast_nationalitate
(
    id_nationalitate number(9,0) not null,
    denumire varchar2(20),
    constraint nationalitate_pk primary key (id_nationalitate)

```

Oracle SQL Developer : C:\Users\Antonia\AppData\Roaming\SQL Developer\project_bd.sql

File Edit View Navigate Run Source Team Tools Window Help

antonia.sql lab_suplimentar.sql Welcome Page project_bd.sql

SQL Worksheet History

Worksheet Query Builder

```
create table ast_prezinta
(
    id_fotomodel number(9,0) not null,
    id_defilare number(9,0) not null,
    id_tinuta number(9,0) not null,
    data_prezentare date,
    constraint prezinta_pk primary key (id_fotomodel, id_defilare, id_tinuta),
    constraint prezinta_fotomodel_fkl foreign key (id_fotomodel) references ast_fotomodel(id_fotomodel),
    constraint prezinta_defilare_fkl foreign key (id_defilare) references ast_defilare(id_defilare),
    constraint prezinta_tinuta_fkl foreign key (id_tinuta) references ast_tinute(id_tinuta)
);

insert into ast_tara (id_tara, denumire) values (10, 'Italia');
insert into ast_tara (id_tara, denumire) values (11, 'SUA');
insert into ast_tara (id_tara, denumire) values (12, 'Romania');
insert into ast_tara (id_tara, denumire) values (13, 'Singapore');
insert into ast_tara (id_tara, denumire) values (14, 'Franta');

insert into ast_oras (id_oras, denumire, id_tara) values (101, 'Milano', 10);
insert into ast_oras (id_oras, denumire, id_tara) values (102, 'Singapore', 13);
insert into ast_oras (id_oras, denumire, id_tara) values (103, 'New York', 11);
insert into ast_oras (id_oras, denumire, id_tara) values (104, 'Paris', 14);
insert into ast_oras (id_oras, denumire, id_tara) values (105, 'Bucuresti', 12);

insert into ast_fashionweek (id_evenimet, data_inceput, data_sfarsit, id_oras) values (101, to_timestamp('01-aug-2017','dd-mon-rr hh.mi.sxfff am'), to_timestamp('06-aug-2017','dd-mon-rr hh.mi.sxfff am'), 102);
insert into ast_fashionweek (id_evenimet, data_inceput, data_sfarsit, id_oras) values (105, to_timestamp('02-jul-2014','dd-mon-rr hh.mi.sxfff am'), to_timestamp('09-jul-2014','dd-mon-rr hh.mi.sxfff am'), 101);
insert into ast_fashionweek (id_evenimet, data_inceput, data_sfarsit, id_oras) values (203, to_timestamp('7-oct-2013','dd-mon-rr hh.mi.sxfff am'), to_timestamp('15-oct-2013','dd-mon-rr hh.mi.sxfff am'), 104);
insert into ast_fashionweek (id_evenimet, data_inceput, data_sfarsit, id_oras) values (304, to_timestamp('03-nov-2015','dd-mon-rr hh.mi.sxfff am'), to_timestamp('08-nov-2015','dd-mon-rr hh.mi.sxfff am'), 103);
insert into ast_fashionweek (id_evenimet, data_inceput, data_sfarsit, id_oras) values (406, to_timestamp('17-jun-2016','dd-mon-rr hh.mi.sxfff am'), to_timestamp('25-jun-2016','dd-mon-rr hh.mi.sxfff am'), 103);

insert into ast_post_tv (id_post, denumire, emisiune) values (1, 'CNN', 'Fashion time');
insert into ast_post_tv (id_post, denumire) values (2, 'TV');
insert into ast_post_tv (id_post, denumire, emisiune) values (3, 'Fashion tv', 'Fashion time show');
insert into ast_post_tv (id_post, denumire) values (4, 'Antena3');
insert into ast_post_tv (id_post, denumire, emisiune) values (5, 'Antena Star', 'Star matinal');

insert into ast_presa (id_jurnalist, nume, prenume) values (20, 'Chris', 'Stuard');
insert into ast_presa (id_jurnalist, nume, prenume, angajator) values (21, 'Anna', 'Collins', 'Vogue');
insert into ast_presa (id_jurnalist, nume, prenume, angajator) values (22, 'Elleh', 'Hopkins', 'CNN');
insert into ast_presa (id_jurnalist, nume, prenume, angajator) values (23, 'Alexandra', 'Morar', 'Antena');
insert into ast_presa (id_jurnalist, nume, prenume) values (24, 'Emily', 'Jones');
```

File Edit View Navigate Run Source Team Tools Window Help

antonia.sql lab_suplimentar.sql Welcome Page project_bd.sql

SQL Worksheet History

Worksheet Query Builder

```

insert into ast_promoveaza (id_eveniment, id_post, id_jurnalst, durata, vizualizari) values (101,1,22,15,150000);
insert into ast_promoveaza (id_eveniment, id_post, id_jurnalst, durata, vizualizari) values (101,2,24,30,1000);
insert into ast_promoveaza (id_eveniment, id_post, id_jurnalst, durata, vizualizari) values (105,3,20,10,12000);
insert into ast_promoveaza (id_eveniment, id_post, id_jurnalst, durata, vizualizari) values (105,6,23,3,8000);
insert into ast_promoveaza (id_eveniment, id_post, id_jurnalst, durata, vizualizari) values (203,4,21,2,1000);
insert into ast_promoveaza (id_eveniment, id_post, id_jurnalst, durata, vizualizari) values (203,2,21,5,9000);
insert into ast_promoveaza (id_eveniment, id_post, id_jurnalst, durata, vizualizari) values (304,1,22,10,130000);
insert into ast_promoveaza (id_eveniment, id_post, id_jurnalst, durata, vizualizari) values (304,5,23,3,3000);
insert into ast_promoveaza (id_eveniment, id_post, id_jurnalst, durata, vizualizari) values (406,2,24,5,1200);
insert into ast_promoveaza (id_eveniment, id_post, id_jurnalst, durata, vizualizari) values (406,4,20,3,8800);

insert into ast_participanti (id_participant, prenume, nume) values (151,'Sandu','Aurel');
insert into ast_participanti (id_participant, prenume, nume) values (161,'Carmen','Diaconescu');
insert into ast_participanti (id_participant, prenume, nume) values (207,'Jane','Stuard');
insert into ast_participanti (id_participant, prenume, nume) values (306,'Michele','Marsh');
insert into ast_participanti (id_participant, prenume, nume) values (450,'Robin','Smith');
insert into ast_participanti (id_participant, prenume, nume) values (351,'Sandy','Butler');

insert into ast_participa (id_eveniment, id_participant, discount) values (101,151,0.3);
insert into ast_participa (id_eveniment, id_participant) values (101,306);
insert into ast_participa (id_eveniment, id_participant, discount) values (203,161,0.1);
insert into ast_participa (id_eveniment, id_participant) values (203,450);
insert into ast_participa (id_eveniment, id_participant) values (105,161);
insert into ast_participa (id_eveniment, id_participant) values (105,450);
insert into ast_participa (id_eveniment, id_participant, discount) values (105,207,0.2);
insert into ast_participa (id_eveniment, id_participant, discount) values (406,306,0.2);
insert into ast_participa (id_eveniment, id_participant) values (406,151);
insert into ast_participa (id_eveniment, id_participant) values (304,306);

insert into ast_locuri (id_loc, id_participant, pret, disponibilitate) values (35,151, 150,1);
insert into ast_locuri (id_loc, pret, disponibilitate) values (36, 150,0);
insert into ast_locuri (id_loc, id_participant, pret, disponibilitate) values (37,207, 150,1);
insert into ast_locuri (id_loc, id_participant, pret, disponibilitate) values (38,207, 150,1);
insert into ast_locuri (id_loc, pret, disponibilitate) values (39, 150,0);
insert into ast_locuri (id_loc, id_participant, pret, disponibilitate) values (40,161, 150,1);
insert into ast_locuri (id_loc, id_participant, pret, disponibilitate) values (41,161, 150,1);

insert into ast_loc_vip (id_loc, id_participant, pret, disponibilitate, pozitie) values (1,450,350,1,'zona 0');
insert into ast_loc_vip (id_loc, id_participant, pret, disponibilitate, pozitie) values (2,450,350,1,'zona 0');
insert into ast_loc_vip (id_loc, pret, disponibilitate, pozitie) values (3,350,0,'zona 0');
insert into ast_loc_vip (id_loc, id_participant, pret, disponibilitate, pozitie) values (4,351,350,1,'zona 0');
insert into ast_loc_vip (id_loc, id_participant, pret, disponibilitate, pozitie) values (8,306,350,1,'zona 0');

```

Script Output * Query Result * Query Result 1 *

Oracle SQL Developer : C:\Users\Antonia\AppData\Roaming\SQL Developer\project_bd.sql

File Edit View Navigate Run Source Team Tools Window Help

antonia.sql lab_suplimentar.sql Welcome Page project_bd.sql

SQL Worksheet History

Worksheet Query Builder

```
insert into ast colectii (id colectie, id_brand, sezon, tematica) values (ast_seq colectii.nextval,500,'toamna-iarna','New color');
insert into ast colectii (id colectie, id_brand, sezon, tematica) values (ast_seq colectii.nextval,500,'primavara-vara','Indian style');
insert into ast colectii (id colectie, id_brand, sezon) values (ast_seq colectii.nextval,300,'toamna-iarna');
insert into ast colectii (id colectie, id_brand, sezon, tematica) values (ast_seq colectii.nextval,400,'toamna-iarna','Country side style');
insert into ast colectii (id colectie, id_brand, sezon) values (ast_seq colectii.nextval,200,'primavara-vara');

create sequence ast_seq_tinute
increment by 1
start with 10
maxvalue 1000
nocycle;

insert into ast_tinute (id_tinuta, id colectie, pret, material) values (ast_seq_tinute.nextval, 1,500,'casmir');
insert into ast_tinute (id_tinuta, id colectie, pret, material) values (ast_seq_tinute.nextval, 1,180.0,'lana');
insert into ast_tinute (id_tinuta, id colectie, pret, material) values (ast_seq_tinute.nextval, 7,2700,'voal');
insert into ast_tinute (id_tinuta, id colectie, pret, material) values (ast_seq_tinute.nextval, 5,3000,'panza');
insert into ast_tinute (id_tinuta, id colectie, pret, material) values (ast_seq_tinute.nextval, 2,1750,'matase');
insert into ast_tinute (id_tinuta, id colectie, pret, material) values (ast_seq_tinute.nextval, 6,3600,'bumbac');
insert into ast_tinute (id_tinuta, id colectie, pret, material) values (ast_seq_tinute.nextval, 9,9900,'piele');
insert into ast_tinute (id_tinuta, id colectie, pret, material) values (ast_seq_tinute.nextval, 3,2000,'lana');
insert into ast_tinute (id_tinuta, id colectie, pret, material) values (ast_seq_tinute.nextval, 5,5000,'in');

insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (101, 100,100);
insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (101, 300,100);
insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (203, 200,200);
insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (203, 300,200);
insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (105, 100,150);
insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (105, 500,150);
insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (304, 200,100);
insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (304, 400,100);
insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (406, 400,220);
insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (406, 500,220);

insert into ast_cumpara (id_participant, id_tinuta, marime) values (151,10,'S');
insert into ast_cumpara (id_participant, id_tinuta, marime) values (151,11,'S');
insert into ast_cumpara (id_participant, id_tinuta, marime, discount) values (207,17,'M',5);
insert into ast_cumpara (id_participant, id_tinuta, marime) values (306,16,'L');
insert into ast_cumpara (id_participant, id_tinuta, marime) values (306,12,'L');
insert into ast_cumpara (id_participant, id_tinuta, marime) values (306,10,'L');
insert into ast_cumpara (id_participant, id_tinuta, marime, discount) values (450,17,'XS',2);
insert into ast_cumpara (id_participant, id_tinuta, marime, discount) values (161,17,'M',2);
insert into ast_cumpara (id_participant, id_tinuta, marime) values (161,15,'M');
insert into ast_cumpara (id_participant, id_tinuta, marime) values (450,12,'S');
```

Script Output x Query Result x Query Result 1 x

Oracle SQL Developer : C:\Users\Antonia\AppData\Roaming\SQL Developer\proiect_bd.sql

File Edit View Navigate Run Source Team Tools Window Help



antonia.sql lab_suplimentar.sql Welcome Page proiect_bd.sql

SQL Worksheet History



Worksheet Query Builder

```
insert into ast_tinute (id_tinuta, id_colectie, pret, material) values (ast_seq_tinute.nextval, 5,3000,'panza');
insert into ast_tinute (id_tinuta, id_colectie, pret, material) values (ast_seq_tinute.nextval, 2,1750,'matase');
insert into ast_tinute (id_tinuta, id_colectie, pret, material) values (ast_seq_tinute.nextval, 6,3600,'bumbac');
insert into ast_tinute (id_tinuta, id_colectie, pret, material) values (ast_seq_tinute.nextval, 9,9900,'piele');
insert into ast_tinute (id_tinuta, id_colectie, pret, material) values (ast_seq_tinute.nextval, 3,2000,'lana');
insert into ast_tinute (id_tinuta, id_colectie, pret, material) values (ast_seq_tinute.nextval, 5,5000,'in');

insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (101, 100,100);
insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (101, 300,100);
insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (203, 200,200);
insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (203, 300,200);
insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (105, 100,150);
insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (105, 500,150);
insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (304, 200,100);
insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (304, 400,100);
insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (406, 400,220);
insert into ast_ia_parte (id_eveniment, id_brand, taxa_participare) values (406, 500,220);

insert into ast_cumpara (id_participant, id_tinuta, marime) values (151,10,'S');
insert into ast_cumpara (id_participant, id_tinuta, marime) values (151,11,'S');
insert into ast_cumpara (id_participant, id_tinuta, marime, discount) values (207,17,'M',5);
insert into ast_cumpara (id_participant, id_tinuta, marime) values (306,16,'L');
insert into ast_cumpara (id_participant, id_tinuta, marime) values (306,12,'L');
insert into ast_cumpara (id_participant, id_tinuta, marime) values (306,10,'L');
insert into ast_cumpara (id_participant, id_tinuta, marime, discount) values (450,17,'XS',2);
insert into ast_cumpara (id_participant, id_tinuta, marime, discount) values (161,17,'M',2);
insert into ast_cumpara (id_participant, id_tinuta, marime) values (161,15,'M');
insert into ast_cumpara (id_participant, id_tinuta, marime) values (450,12,'S');
insert into ast_cumpara (id_participant, id_tinuta, marime) values (207,12,'XS');

insert into ast_prezinta (id_fotomodel, id_defilare, id_tinuta, data_prezentare) values (20,1,10,to_timestamp('04-aug-2017','dd-mon-rr hh.mi.ssxiff am'));
insert into ast_prezinta (id_fotomodel, id_defilare, id_tinuta, data_prezentare) values (21,1,11,to_timestamp('04-aug-2017','dd-mon-rr hh.mi.ssxiff am'));
insert into ast_prezinta (id_fotomodel, id_defilare, id_tinuta, data_prezentare) values (24,1,17,to_timestamp('05-aug-2017','dd-mon-rr hh.mi.ssxiff am'));
insert into ast_prezinta (id_fotomodel, id_defilare, id_tinuta, data_prezentare) values (22,3,12,to_timestamp('08-oct-2013','dd-mon-rr hh.mi.ssxiff am'));
insert into ast_prezinta (id_fotomodel, id_defilare, id_tinuta, data_prezentare) values (23,3,14,to_timestamp('09-oct-2013','dd-mon-rr hh.mi.ssxiff am'));
insert into ast_prezinta (id_fotomodel, id_defilare, id_tinuta, data_prezentare) values (20,2,15,to_timestamp('08-jul-2014','dd-mon-rr hh.mi.ssxiff am'));
insert into ast_prezinta (id_fotomodel, id_defilare, id_tinuta, data_prezentare) values (24,4,18,to_timestamp('07-nov-2015','dd-mon-rr hh.mi.ssxiff am'));
insert into ast_prezinta (id_fotomodel, id_defilare, id_tinuta, data_prezentare) values (20,4,16,to_timestamp('07-nov-2015','dd-mon-rr hh.mi.ssxiff am'));
insert into ast_prezinta (id_fotomodel, id_defilare, id_tinuta, data_prezentare) values (23,5,13,to_timestamp('25-jun-2016','dd-mon-rr hh.mi.ssxiff am'));
insert into ast_prezinta (id_fotomodel, id_defilare, id_tinuta, data_prezentare) values (22,5,15,to_timestamp('18-jun-2016','dd-mon-rr hh.mi.ssxiff am'));

commit;
```

Script Output x Query Result x Query Result 1 x

11. Formulați în limbaj natural și implementați 5 cereri SQL complexe

```
--11. Formulați în limbaj natural și implementați 5 cereri SQL complexe

--1. Sa se afiseze tarile ordonate dupa nume in care a calatorit participantul cu id-ul 161 pentru a participa la prezentarile din fashion week
-- Se va folosi: subcerere nesincronizata + ordonare
select ast_tara.id_tara, ast_tara.denumire
from ast_tara
where id_tara in(select id_tara
                  from ast_oras
                  where id_oras in(select id_oras
                                   from ast_fashionweek
                                   where id_evenimet in(select id_evenimet
                                                         from ast_participa
                                                         where ast_participa.id_eveniment = ast_fashionweek.id_evenimet and id_participant = 161 and ast_participa.id_eveniment = ast_fashionweek.id_evenimet
                                                         in(select id_participant
                                                            from ast_participanti
                                                            where id_participant = 161)
                                                         )
                                   )
                  )
order by ast_tara.denumire;
```

Script Output x Query Result x Query Result 1 x

SQL | All Rows Fetched: 2 in 0.185 seconds

ID_TARA	DENUMIRE
14	Franta
10	Italia

```
--2.1. Pentru fiecare participant care contine 'ha' in prenume sa se afiseze suma totala a tinutele pe care le-a cumparat si intr-o coloana 'status pret' sa se
-- afiseze un mesaj care indica statusul cheltuielilor (mic, 3000<mediu <5000, mare>=5000)
-- Se foloseste: clauza with + case + group by + ordonare
with pret as (select ast_cumpara.id_participant, sum(pret) as total
              from ast_tinute
              join ast_cumpara on (ast_tinute.id_tinuta = ast_cumpara.id_tinuta)
              join ast_participanti on (ast_cumpara.id_participant = ast_participanti.id_participant)
              where ast_participanti.prenume like ('%ha%')
              group by ast_cumpara.id_participant
              order by id_participant)

select id_participant, total,
       case when total > 5000 then 'Mare'
            when total < 5000 and total >=2000 then 'Mediu'
            else 'Mic'
       end as "status cheltuieli"
from pret;
```

Script Output x Query Result x Query Result 1 x

SQL | All Rows Fetched: 1 in 0.063 seconds

ID_PARTICIPANT	TOTAL	status cheltuieli
1	151	680 Mic

Oracle SQL Developer : C:\Users\Antonia\AppData\Roaming\SQL Developer\proiect_bd.sql

File Edit View Navigate Run Source Team Tools Window Help

antonia.sql lab_suplimentar.sql Welcome Page proiect_bd.sql

SQL Worksheet History

Worksheet Query Builder

```
from pret;

--2.2 Pentru fiecare fotomodel care contine 'ha' in prenume sa se afiseze suma totala a tinutele pe care le-a cumparat si intr-o coloana 'status salarii' sa se
-- afiseze un mesaj care indica statusul cheltuielilor (mic, 20000<salariu mediu <50000, mare>=50000)
with salariu as(select id_fotomodel, salariu, prenume
                  from ast_fotomodel
                  where ast_fotomodel.prenume like ('%ha%')
                  group by id_fotomodel, salariu, prenume
                  order by id_fotomodel)
select id_fotomodel, salariu, prenume,
case when salariu >= 50000 then 'salariu mare'
      when salariu < 50000 and salariu >= 20000 then 'salariu mediu'
      else 'salariu mic'
end as "status salarii"
from salariu;
```

Script Output x Query Result x Query Result 1 x

SQL | All Rows Fetched: 2 in 0.111 seconds

ID_FOTOMODEL	SALARIU	PRENUME	status salarii
1	20	(null) Samantha	salariu mic
2	22	(null) Martha	salariu mic

```
--3. Verificati daca un loc e rezervat si afisati intr-o 'Rezervat', respectiv 'Nerezervat' intr-o coloana denumita corespunzator (daca
-- disponibilitate = (null) => nu e rezervat
-- Se foloseste: decode + nvl
-- verf daca exisya o agentie cu doul 30 folosind DECODE
select id_loc,
decode(nvl(disponibilitate, 0), 1, 'Rezervat', 'Nerezervat') as rezervari
from ast_loc_vip;
```

Script Output x Query Result x Query Result 1 x

SQL | All Rows Fetched: 6 in 0.054 seconds

ID_LOC	REZERVARI
1	1 Rezervat
2	2 Rezervat
3	3 Nerezervat
4	4 Rezervat
5	8 Rezervat
6	9 Rezervat

```
--4. Pentru brandurile care contin tinute a caror pret maxim este >= 4500 si pretul minim <= 3500 , sa se obtina id-ul, numele si pretul maxim si minim
-- al unei tinute a brandului respectiv
-- Se foloseste: grupari de date + functii grup(min, max) + filtrare la nivel de grup
select ast_brand.id_brand, denumire, max(pret) as "pret maxim", min(pret) as "pret minim"
from ast_brand join ast_colectii on (ast_brand.id_brand = ast_colectii.id_brand)
              join ast_tinute on (ast_colectii.id_colectie = ast_tinute.id_colectie)
group by ast_brand.id_brand, denumire
having max(pret) >= 4500 and min(pret) <= 3500;
```

Script Output x Query Result x Query Result 1 x

SQL | All Rows Fetched: 1 in 0.082 seconds

ID_BRAND	DENUMIRE	pret maxim	pret minim
1	400Versace	9900	3000

Oracle SQL Developer : C:\Users\Antonia\AppData\Roaming\SQL Developer\proiect_bd.sql

File Edit View Navigate Run Source Team Tools Window Help

antonia.sql lab_suplimentar.sql Welcome Page proiect_bd.sql

SQL Worksheet History

Worksheet Query Builder

```
group by ast_brand.id_brand, denumire
having max(pret) >= 4500 and min(pret) <= 3500;
```

```
--5. Sa se afiseze detalii despre fotomodelele care au participat la fashion-week-ul care contine 'style' in tematica si a avut loc mai acum mai bine
-- de 80 de luni si (fotomodelele) au prezentat tinute pt brandul care are sediu intr-un oras care contine 'New', cu numele si prenumele concatenat
-- as 'Nume'
```

```
-- Se foloseste: functie sir caracter + functie date calendaristice
```

```
select ast_fotomodel.id_fotomodel, concat(ast_fotomodel.num, ' ' || ast_fotomodel.prenume as "nume fotomodel", ast_colectii.tematica, ast_fashionweek.id_evenimet, ast_brand.id_brand
from ast_fotomodel join ast_prezinta on (ast_fotomodel.id_fotomodel = ast_prezinta.id_fotomodel)
              join ast_tinute on (ast_prezinta.id_tinuta = ast_tinute.id_tinuta)
              join ast_colectii on (ast_tinute.id_colectie = ast_colectii.id_colectie)
              join ast_brand on (ast_colectii.id_brand = ast_brand.id_brand)
              join ast_ia_parte on (ast_brand.id_brand = ast_ia_parte.id_brand)
              join ast_fashionweek on (ast_ia_parte.id_eveniment = ast_fashionweek.id_evenimet)
where ast_colectii.tematica like ('%style%') and ast_brand.sediu like ('%New%') and months_between(to_timestamp(sysdate, 'dd-mon-rr'), data_sfarsit) > 80;
```

Script Output x Query Result x

SQL | All Rows Fetched: 1 in 0.087 seconds

ID_FOTOMODEL	nume fotomodel	TEMATICA	ID_EVENIMET	ID_BRAND
1	22Marshal Martha	Indian style	105	500

12. Implementarea a 3 operații de actualizare sau suprimare a datelor utilizând subcereri.

Oracle SQL Developer : C:\Users\Antonia\AppData\Roaming\SQL Developer\proiect_bd.sql

The screenshot displays the Oracle SQL Developer interface. The main window shows a SQL script in the 'Worksheet' tab. The script includes a comment about implementing three operations (update or delete) using subqueries, followed by an update statement for the 'ast_tinute' table. The update statement increases the price by 10% for items belonging to the 'Chanel' brand. Below the script, the 'Query Result' tab shows the execution results, displaying a table with 9 rows and 4 columns: ID_TINUTA, ID_COLECTIE, PRET, and MATERIAL. The table contains data for various items, including 'casmir', 'lana', 'voal', 'panza', 'matase', 'bumbac', 'piele', 'lana', and 'in'.

```
--12. Implementarea a 3 operatii de actualizare sau suprimare a datelor utilizand subcereri

--1. Sa se mareasca cu 10% pretul tinutelor expuse de brand-ul 'Chanel'
update ast_tinute
set pret = pret*1.1
where ast_tinute.id_colectie in(select ast_colectii.id_colectie
                                from ast_colectii
                                where ast_colectii.id_brand in (select ast_brand.id_brand
                                                                from ast_brand
                                                                where upper(ast_brand.denumire) = upper('Chanel'))
                                );
select * from ast_tinute;
rollback;
```

Script Output x Query Result x

SQL | All Rows Fetched: 9 in 0.071 seconds

ID_TINUTA	ID_COLECTIE	PRET	MATERIAL
1	10	1	550 casmir
2	11	1	198 lana
3	12	7	2700 voal
4	13	5	3000 panza
5	14	2	1750 matase
6	15	6	3600 bumbac
7	16	9	9900 piele
8	17	3	2200 lana
9	18	5	5000 in

Oracle SQL Developer : C:\Users\Antonia\AppData\Roaming\SQL Developer\proiect_bd

File Edit View Navigate Run Source Team Tools Window Help



antonia.sql x lab_suplimentar.sql x Welcome Page x proiect_bd.sql x

SQL Worksheet History

0.49000001 seconds

Worksheet Query Builder

```
rollback;

--2. Sa se majoreze cu 5% taxa de participare a brandului cu sediul in 'Paris'
update ast_ia_parte
set taxa_participare = taxa_participare*0.5
where ast_ia_parte.id_brand in (select ast_brand.id_brand
                                from ast_brand
                                where lower(ast_brand.sediu) = lower('Paris'));
select *from ast_ia_parte;
rollback;
```

Script Output x Query Result 1 x

SQL | All Rows Fetched: 10 in 0.06 seconds

	ID_EVENIMENT	ID_BRAND	TAXA_PARTICIPARE
1	101	100	100
2	105	100	150
3	203	200	200
4	304	200	100
5	101	300	50
6	203	300	100
7	304	400	100
8	406	400	220
9	105	500	150
10	406	500	220

Oracle SQL Developer : C:\Users\Antonia\AppData\Roaming\SQL Developer

File Edit View Navigate Run Source Team Tools Window Help



antonia.sql x lab_suplimentar.sql x Welcome Page x proiect_bd.sql

SQL Worksheet History



Worksheet Query Builder

```
rollback;

--3. Sa se elimine tinutele care nu au un cumparator aferent
delete from ast_tinute
where ast_tinute.id_tinuta not in (select unique id_tinuta
                                   from ast_cumpara);

select * from ast_cumpara;
rollback ;
```

Script Output x Query Result 2 x

SQL | All Rows Fetched: 6 in 0.071 seconds

	ID_PARTICIPANT	ID_TINUTA	MARIME	DISCOUNT
1	151	10	S	(null)
2	151	11	S	(null)
3	207	17	M	5
4	306	16	L	(null)
5	306	12	L	(null)
6	306	10	L	(null)

13. Crearea unei secvențe ce va fi utilizată în inserarea înregistrărilor în tabele (punctul 10)

Oracle SQL Developer : C:\Users\Antonia\AppData\Roaming\SQL Developer\proiect_bd.sql

File Edit View Navigate Run Source Team Tools Window Help

antonia.sql lab_suplimentar.sql Welcome Page proiect_bd.sql

SQL Worksheet History

Worksheet Query Builder

```
--ex 13 creare secv
create sequence ast_seq_brand
increment by 100
start with 100
maxvalue 10000
nocycle;

insert into ast_brand (id_brand, denumire, sediu) values (ast_seq_brand.nextval, 'Chanel', 'New York');
insert into ast_brand (id_brand, denumire, sediu) values (ast_seq_brand.nextval, 'Dior', 'Boston');
insert into ast_brand (id_brand, denumire, sediu) values (ast_seq_brand.nextval, 'Luis Vuitton', 'Paris');
insert into ast_brand (id_brand, denumire, sediu) values (ast_seq_brand.nextval, 'Versace', 'Milano');
insert into ast_brand (id_brand, denumire, sediu) values (ast_seq_brand.nextval, 'Prada', 'New Jersey');

create sequence ast_seq_colectii
increment by 1
start with 1
maxvalue 1000
nocycle;

insert into ast_colectii (id_colectie, id_brand, sezon, tematica) values (ast_seq_colectii.nextval, 100, 'toamna-iarna', 'Space theme');
insert into ast_colectii (id_colectie, id_brand, sezon, tematica) values (ast_seq_colectii.nextval, 300, 'primavara-vara', 'Escalator runway');
insert into ast_colectii (id_colectie, id_brand, sezon, tematica) values (ast_seq_colectii.nextval, 100, 'toamna-iarna', 'Grocery store');
insert into ast_colectii (id_colectie, id_brand, sezon, tematica) values (ast_seq_colectii.nextval, 200, 'primavara-vara');
insert into ast_colectii (id_colectie, id_brand, sezon, tematica) values (ast_seq_colectii.nextval, 400, 'primavara-vara', 'Underwater escapism');
insert into ast_colectii (id_colectie, id_brand, sezon, tematica) values (ast_seq_colectii.nextval, 500, 'toamna-iarna', 'New color');
insert into ast_colectii (id_colectie, id_brand, sezon, tematica) values (ast_seq_colectii.nextval, 500, 'primavara-vara', 'Indian style');
insert into ast_colectii (id_colectie, id_brand, sezon, tematica) values (ast_seq_colectii.nextval, 300, 'toamna-iarna');
insert into ast_colectii (id_colectie, id_brand, sezon, tematica) values (ast_seq_colectii.nextval, 400, 'toamna-iarna', 'Country side style');
insert into ast_colectii (id_colectie, id_brand, sezon, tematica) values (ast_seq_colectii.nextval, 200, 'primavara-vara');
```

Script Output Query Result 2

Oracle SQL Developer : C:\Users\Antonia\AppData\Roaming\SQL Developer\proiect_bd.sql

File Edit View Navigate Run Source Team Tools Window Help

antonia.sql lab_suplimentar.sql Welcome Page proiect_bd.sql

SQL Worksheet History

Worksheet Query Builder

```
create sequence ast_seq_tinute
increment by 1
start with 10
maxvalue 1000
nocycle;

insert into ast_tinute (id_tinuta, id_colectie, pret, material) values (ast_seq_tinute.nextval, 1, 500, 'casmir');
insert into ast_tinute (id_tinuta, id_colectie, pret, material) values (ast_seq_tinute.nextval, 1, 180.0, 'lana');
insert into ast_tinute (id_tinuta, id_colectie, pret, material) values (ast_seq_tinute.nextval, 7, 2700, 'voal');
insert into ast_tinute (id_tinuta, id_colectie, pret, material) values (ast_seq_tinute.nextval, 5, 3000, 'panza');
insert into ast_tinute (id_tinuta, id_colectie, pret, material) values (ast_seq_tinute.nextval, 2, 1750, 'matase');
insert into ast_tinute (id_tinuta, id_colectie, pret, material) values (ast_seq_tinute.nextval, 6, 3600, 'bumbac');
insert into ast_tinute (id_tinuta, id_colectie, pret, material) values (ast_seq_tinute.nextval, 9, 9900, 'piele');
insert into ast_tinute (id_tinuta, id_colectie, pret, material) values (ast_seq_tinute.nextval, 3, 2000, 'lana');
insert into ast_tinute (id_tinuta, id_colectie, pret, material) values (ast_seq_tinute.nextval, 5, 5000, 'in');
```

14. Crearea unei vizualizări compuse. Dați un exemplu de operație LMD permisă pe vizualizarea respectivă și un exemplu de operație LMD nepermisă.

```
--14. Crearea unei vizualizări compuse. Dați un exemplu de operație LMD permisă pe
-- vizualizarea respectivă și un exemplu de operație LMD nepermisă
create view ast_shopping_overview
as select ast_participanti.numel||' '||ast_participanti.prenume as "nume complet", ast_cumpara.marime, ast_tinute.pret, ast_tinute.material, ast_colectii.tematica
from ast_participanti inner join ast_cumpara on (ast_participanti.id_participant = ast_cumpara.id_participant)
inner join ast_tinute on (ast_cumpara.id_tinuta = ast_tinute.id_tinuta)
inner join ast_colectii on (ast_tinute.id_colectie = ast_colectii.id_colectie);
-- select este un exemplu de o comanda permisa pt o vizualizare compusa
select * from ast_shopping_overview;

-- comenzile LMD: insert/delete/update nu sunt pot fi aplicate direct pe vizualizarea compusa deoarece aceasta e formata
-- prin join la mai multe tabele
```

Script Output x Query Result x

SQL | All Rows Fetched: 6 in 0.774 seconds

	nume complet	MARIME	PRET	MATERIAL	TEMATICA
1	Aurel Sandu	S	180	lana	Space theme
2	Aurel Sandu	S	500	casmir	Space theme
3	Marsh Michele	L	500	casmir	Space theme
4	Stuard Jane	M	2000	lana	Grocery store
5	Marsh Michele	L	2700	voal	Indian style
6	Marsh Michele	L	9900	piele	Country side style

15. Crearea unui index care să optimizeze o cerere de tip căutare cu 2 criterii. Specificați cererea.

Oracle SQL Developer : C:\Users\Antonia\AppData\Roaming\SQL Developer\proiect_bd.sql

File Edit View Navigate Run Source Team Tools Window Help

antonia.sql x lab_suplimentar.sql x proiect_bd.sql x Welcome Page x antonia x

SQL Worksheet History

Worksheet Query Builder

```
--from ast_participanti inner join ast_cumpara on (ast_participanti.id_participant = ast_cumpara.id_participant)
--inner join ast_tinute on (ast_cumpara.id_tinuta = ast_tinute.id_tinuta)
--inner join ast_colectii on (ast_tinute.id_colectie = ast_colectii.id_colectie);
-- select este un exemplu de o comanda permisa pt o vizualizare compusa
select * from ast_shopping_overview;

-- comenzile LMD: insert/delete/update nu sunt pot fi aplicate direct pe vizualizarea compusa deoarece aceasta e formata
-- prin join la mai multe tabele

--15. Crearea unui index care să optimizeze o cerere de tip căutare cu 2 criterii. Specificați cererea

--1. Sa se gaseasca fotomodelele cu prenumele 'Melanie' si salariu > 30000
-- folosim index pt optimizarea timpului de cautare
create index ast_prenume_salariu
on ast_fotomodel(prenume, salariu);

select prenume, salariu
from ast_fotomodel
where prenume = 'Melanie' and salariu > 3000;
```

Script Output x Query Result x

SQL | All Rows Fetched: 1 in 0.078 seconds

	PRENUME	SALARIU
1	Melanie	45000

16. Formulați în limbaj natural și implementați în SQL: o cerere ce utilizează operația outerjoin pe minimum 4 tabele și două cereri ce utilizează operația division.

DEVISION

Oracle SQL Developer : C:\Users\Antonia\AppData\Roaming\SQL Developer\proiect_bd.sql

File Edit View Navigate Run Source Team Tools Window Help

antonia.sql lab_suplimentar.sql proiect_bd.sql Welcome Page antonia

SQL Worksheet History

Worksheet Query Builder

```
--16. Formulați in limbaj natural si implementati in SQL: o cerere ce utilizeaza operatia outer-join pe minimum 4 tabele si doua
--cereri ce utilizeaza operatia division

--DEVISION
--1. Afisati toate fotomodelele care participa la aceleasi fashion week-uri cu fotomodelul cu id-ul 22
-- si au pprezentat un nr egal de tinute
select ast_prezinta.id_fotomodel, nume||' '||prenume as "Nume complet"
from ast_prezinta join ast_fotomodel on (ast_prezinta.id_fotomodel = ast_fotomodel.id_fotomodel)
where id_defilare in (select id_defilare
                     from ast_prezinta
                     where ast_prezinta.id_fotomodel = 23)
and ast_prezinta.id_fotomodel != 23
group by ast_prezinta.id_fotomodel, nume, prenume
having count(*) = (select count(id_defilare)
                  from ast_prezinta
                  where ast_prezinta.id_fotomodel = 23);

--2. Afisati toti participantii care au cumparat aceleasi tinute cu participantul cu id-ul 207
select ast_cumpara.id_participant, nume||' '||prenume as "Nume complet"
from ast_cumpara join ast_participanti on (ast_cumpara.id_participant = ast_participanti.id_participant)
where id_tinuta in (select ast_cumpara.id_tinuta
                   from ast_cumpara
                   where ast_cumpara.id_participant = 207)
and ast_cumpara.id_participant != 207
group by ast_cumpara.id_participant, nume, prenume
having count(*) = (select count(id_tinuta)
                  from ast_cumpara
                  where ast_cumpara.id_participant = 207);
```

Script Output x Query Result x

Oracle SQL Developer : C:\Users\Antonia\AppData\Roaming\SQL Developer\proiect_bd.sql

File Edit View Navigate Run Source Team Tools Window Help

antonia.sql lab_suplimentar.sql Welcome Page proiect_bd.sql

SQL Worksheet History

Worksheet Query Builder

```
--16. Formulați in limbaj natural si implementati in SQL: o cerere ce utilizeaza operatia outer-join pe minimum 4 tabele si doua
--cereri ce utilizeaza operatia division

--DEVISION
--1. Afisati toate fotomodelele care participa la aceleasi fashion week-uri cu fotomodelul cu id-ul 22
-- si au pprezentat un nr egal de tinute
select ast_prezinta.id_fotomodel, nume||' '||prenume as "Nume complet"
from ast_prezinta join ast_fotomodel on (ast_prezinta.id_fotomodel = ast_fotomodel.id_fotomodel)
where id_defilare in (select id_defilare
                     from ast_prezinta
                     where ast_prezinta.id_fotomodel = 23)
and ast_prezinta.id_fotomodel != 23
group by ast_prezinta.id_fotomodel, nume, prenume
having count(*) = (select count(id_defilare)
                  from ast_prezinta
                  where ast_prezinta.id_fotomodel = 23);
```

Script Output x Query Result x

SQL | All Rows Fetched: 1 in 0.064 seconds

ID_FOTOMODEL	Nume complet
1	22 Marshal Martha

OUTER JOIN

```
--OUTER JOIN
--1. Sa se afiseze numele, prenumele tuturor participantilor. Pt fiecare participant in parte se va afisa fashion week-ul
-- la care a participat dar si tinutele cumparate
select ast_participanti.nume, ast_participanti.prenume, ast_tinute.id_tinuta, ast_fashionweek.id_evenimet
from ast_participanti full outer join ast_participa on (ast_participanti.id_participant = ast_participa.id_participant)
full outer join ast_cumpara on (ast_participanti.id_participant = ast_cumpara.id_participant)
full outer join ast_tinute on (ast_cumpara.id_tinuta = ast_tinute.id_tinuta)
full outer join ast_fashionweek on (ast_participa.id_eveniment = ast_fashionweek.id_evenimet);
```

14. Creeaza unoi vizualizări comune. Dăți un exemplu de operație IMD pentru ea

Script Output x Query Result x Query Result 1 x

SQL | All Rows Fetched: 23 in 0.106 seconds

	NUME	PRENUME	ID_TINUTA	ID_EVENIMET
1	Aurel	Sandu	11	101
2	Aurel	Sandu	10	101
3	Aurel	Sandu	11	406
4	Aurel	Sandu	10	406
5	Diaconescu	Carmen	(null)	105
6	Diaconescu	Carmen	(null)	203
7	Stuard	Jane	17	105
8	Marsh	Michele	16	101
9	Marsh	Michele	12	101
10	Marsh	Michele	10	101
11	Marsh	Michele	16	304
12	Marsh	Michele	12	304
13	Marsh	Michele	10	304
14	Marsh	Michele	16	406
15	Marsh	Michele	12	406
16	Marsh	Michele	10	406
17	Smith	Robin	(null)	105
18	Smith	Robin	(null)	203
19	Butler	Sandy	(null)	(null)
20	(null)	(null)	13	(null)
21	(null)	(null)	15	(null)
22	(null)	(null)	18	(null)
23	(null)	(null)	14	(null)

17. Optimizarea unei cereri, aplicând regulile de optimizare ce derivă din proprietățile operatorilor algebrei relaționale. Cererea va fi exprimată prin expresie algebrică, arbore algebric și limbaj (SQL).

Cerinta: Să se afișeze numele, prenumele, salariul, pretul, materialul și tematica ținutelor pentru fotomodelele care au salariu : 30000 sau a căror ținute conțin ”style” în tematică.

De asemenea, numele și prenumele trebuie concatenate, având o denumire corespunzătoare.

Mai mult, datele trebuie ordonate după numele fotomodelului.

```
--17. Cerinta:
-- Sa se afiseze numele, prenumele, salariul, pretul, materialul si tematica tinutelor
-- pentru fotomodelele care au salariu > 30000 sau a caror tinute contin 'style' in tematica
select ast_fotomodel.nume||' '||ast_fotomodel.prenume as "Nume complet", salariu, material, pret, tematica
from ast_fotomodel join ast_prezinta on (ast_fotomodel.id_fotomodel = ast_prezinta.id_fotomodel)
                join ast_tinute on (ast_prezinta.id_tinuta = ast_tinute.id_tinuta)
                join ast_colectii on (ast_tinute.id_colectie = ast_colectii.id_colectie)
                --join ast_brand on (ast_colectii.id_brand = ast_brand.id_brand)
where salariu > 30000 or tematica like '%style%'
order by "Nume complet";
```

Query Result x

SQL | All Rows Fetched: 5 in 0.108 seconds

	Nume complet	SALARIU	MATERIAL	PRET	TEMATICA
1	Kennedy Melanie	45000	in	5000	Underwater escapism
2	Kennedy Melanie	45000	lana	2000	Grocery store
3	Marshal Martha	21000	voal	2700	Indian style
4	Rodriguez Samantha	17000	piele	9900	Country side style
5	Smith Bella	70000	lana	180	Space theme

CERERE EXPRIMATĂ PRIN EXPRESII ALGEBRICE

R1 = SELECT (FOTOMODEL, salariu > 30 000)

R2 = PROJECT (R1, id_fotomodel, nume, prenume, salariu)

R3 = PROJECT (PREZINTA, id_fotomodel, id_tinuta)

R4 = JOIN (R2, R3, id_fotomodel)

R5 = SELECT (COLECTIE, tematica like ('%style%'))

R6 = PROJECT (R5, tematica, id_colectie)

R7 = JOIN (R4, TINUTE, id_tinuta)

R8 = JOIN (R6, R7, id_colectie)

Rezultat = PROJECT (R8, nume, prenume, salariu, material, pret, tematica)

Cererea este una optimă, deoarece când a fost creată s-a ținut cont de următoarele reguli de optimizare:

- Selectiile se executa cât mai devreme posibil
- Produsele carteziane se înlocuiesc cu join-uri
- Proiecțiile se execută la început pentru a îndepărta attributele nefolositoare

Proprietățile utilizate sunt:

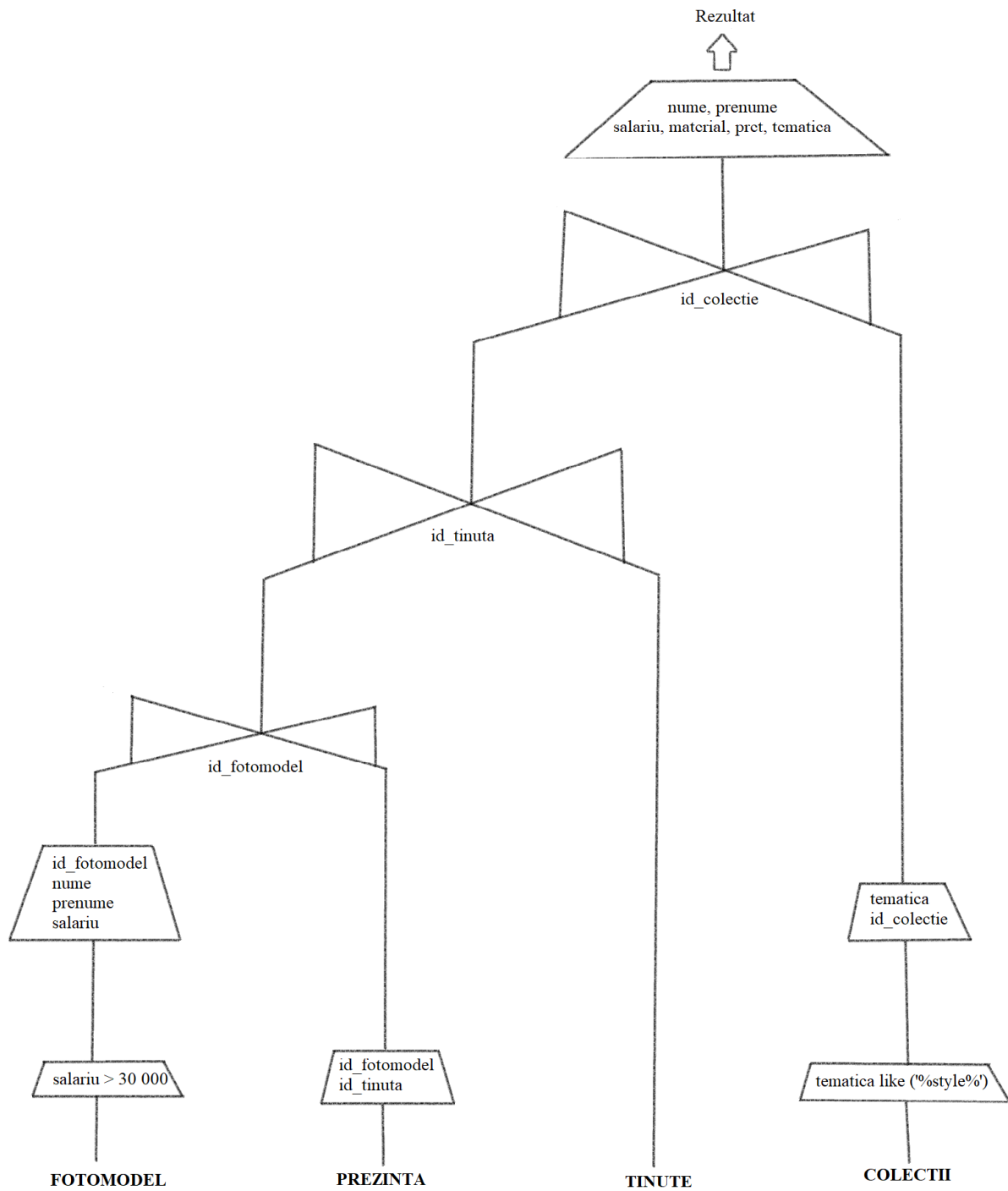
- Comutarea selecției cu proiecția

Mai întâi aplicăm selecția pentru a elimina datele care nu sunt necesare și apoi aplicăm proiect pentru a lua coloanele dorite

- Comutarea proiectiei cu operația join

Prima oară se face proiecție și după join-ul pentru a nu trebui să se facă join pe toate coloanele din relație

ARBORELE ALGEBRIC



18. a. Realizarea normalizării BCNF, FN4, FN5.

BCNF (FORMA NORMALĂ BOYCE CODD)

Def: O relație se află în BCNF, dacă și numai dacă fiecare determinant este o cheie candidat unică.

Pentru a exemplifica o normalizare de tipul BCNF se va presupune că atributul *data_prezentare* din relația **PREZINTA**, este unic și determină funcțional *id_defilare#* care face parte din cheia primară.

În acest mod între attributele relației **PREZINTA** (*id_fotomodel#*, *id_defilare#*, *id_tinuta#*, *data_prezentare*) există dependențele:

$\{data_prezentare\} \rightarrow \{id_defilare\# \}$

$\{id_fotomodel\#, id_defilare\#, id_tinuta\# \} \rightarrow \{data_prezentare\}$

Se poate remarca faptul că atributul *data_prezentare* e o cheie candidat (pentru că este unică) și cheia *id_defilare#* depinde de atributul *data_prezentare*.

Aplicăm regula Casey-Delobel din care rezultă următoarele două relații:

Prezenatre_defilare (*id_defilare*, *data_prezentare#*)

Prezentare_tinute (*id_fotomodel#*, *id_tinuta#*, *data_prezentare#*)

- Atributul *data_prezentare* face parte din cheia primară compusă, pentru a putea avea mai multe date pentru aceeași ținută și fotomodel, atributul *data_prezentare* este singur cheie externă.

FN4 (FORMA NORMALĂ 4)

Def: O relație se află în FN4, dacă și numai dacă se află în BCNF și nu conține relații many-to-many independente (orice dependența multivaloare este o dependență funcțională)

Pentru a exemplifica această relație vom avea următorul exemplu:

Un fotomodel are mai multe ținute și participă la mai multe defilări. Astfel, dacă se va dori să se introducă un nou fotomodel va trebui să se introducă și defilarea și ținuta purtată.

Relația: ***prezentare_nonFN4*** (*id_fotomodel#*, *id_defilare#*, *id_ținută#*)

Este în BCNF.

Id_fotomodel ->-> *id_defilare*

Id_fotomodel ->-> *id_ținuta*

Astfel relațiile rezultate în urma aplicării formei normal 4 sunt:

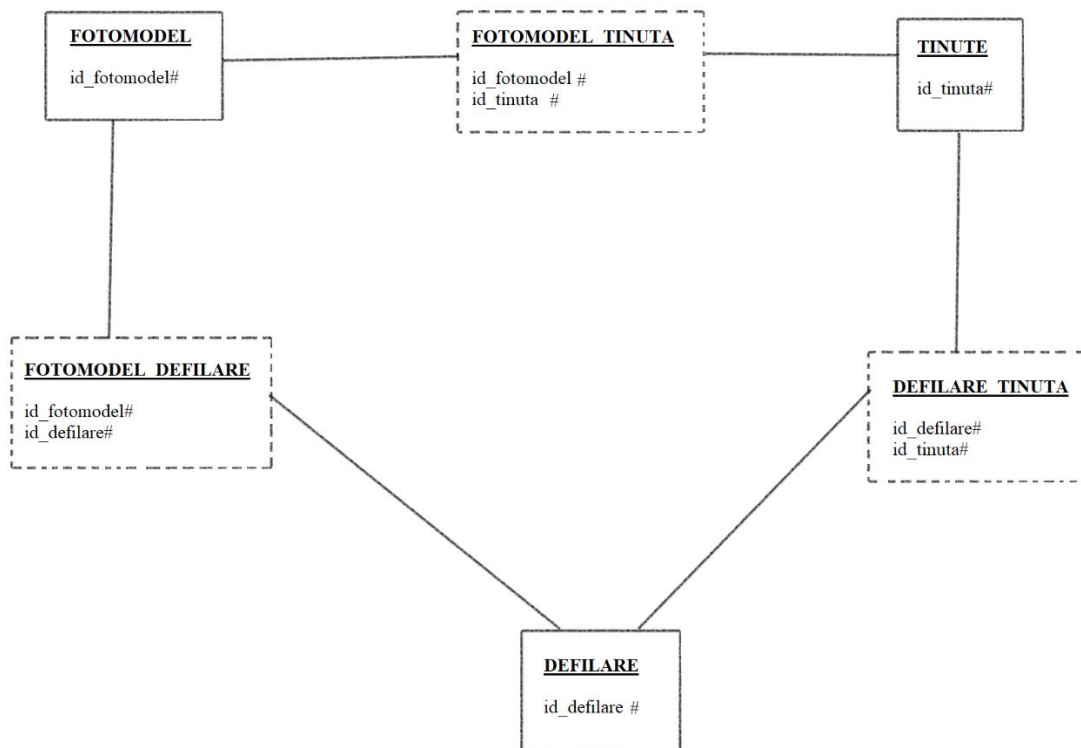
Prezentare_1 (*id_fotomodel#*, *id_ținuta#*)

Prezentare_2 (*id_fotomodel#*, *id_defilare#*)

FN5 (FORMA NORMALĂ 5)

Def: O relație se află în FN5, dacă și numai dacă se află în FN4 și nu conține dependențe ciclice.

Pentru exemplificare se va pleca de la cum arăta o parte din diagramă înainte să fie normalizată.



Se poate remarca faptul că cele 3 relații de tip 2 au compus o diagramă care conține dependențe ciclice, ceea ce înseamnă că relația de mai sus se află în FN5. Dar relația fiind ciclică aceasta va avea o relație de tip 3 echivalentă.

După normalizare se obține relația de tip 3 (cea din diagram).

18.b. Aplicarea denormalizării, justificând necesitatea acesteia.

DENORMALIZAREA

Denormalizare este o tehnică de optimizare a bazei de date în care adăugăm date redundante la unul sau mai multe tabele. Acest lucru ajută la evitarea asocierilor costisitoare într-o bază de date relațională.

Este o tehnică de optimizare care se aplică după normalizare pt reducerea numărului de join-uri care trebuie efectuate pentru rezolvarea unei interogări.

Exemplu: **FOTOMODEL** -> **NATIONALITATE**

În urma denormalizării, atributul *nume_nationalitate* va fi plasat în tabelul **FOTOMODEL** pentru că în acest fel se elimină o operație de join. Astfel, diagrama este mai eficientă.

Id_fotomodel	Nume_nationalitate	nume	prenume	Data_nastere	salariu
20	Spaniolă	Rodriguez	Samanth	2000-01-01	17000
21	Americană	Smith	Bella	1995-10-15	70000
22	americană	Marshal	Martha	1999-06-27	21000

Denormalizarea a fost necesară pentru că nu era eficient ca atributul *nationalitate* să fie într-un tabel separat față de tabelul **FOTOMODEL**, având în vedere că această poziționare implică un join suplimentar pentru a accesa datele din tabelul **NATIONALITATE**.