

PROIECT BAZE DE DATE

Serviciu de gestionare al Jocului de Dungeons and Dragons

ROŞCA Teodora-Maia
grupa 143

1. Descrierea modelului real, a utilității acestuia și a regulilor de funcționare.
2. Prezentarea constrângerilor (restricții, reguli) impuse asupra modelului.
3. Descrierea entităților, incluzând precizarea cheii primare.
4. Descrierea relațiilor, incluzând precizarea cardinalității acestora.
5. Descrierea atributelor, incluzând tipul de date și eventualele constrângeri, valori implicate, valori posibile ale atributelor.
6. Realizarea diagramei entitate-relație corespunzătoare descrierii de la punctele 3-5.
7. Realizarea diagramei conceptuale corespunzătoare diagramei entitate-relație proiectate la punctul 6. Diagrama conceptuală obținută trebuie să conțină minimum 7 tabele (fără considerarea subentităților), dintre care cel puțin un tabel asociativ.
8. Enumerarea schemelor relationale corespunzătoare diagramei conceptuale proiectate la punctul 7.
9. Realizarea normalizării până la forma normală 3 (FN1-FN3)
10. Crearea unei secvențe ce va fi utilizată în inserarea înregistrărilor în tabele.
11. 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 associative; maxim 30 de înregistrări în fiecare tabel).

CREAREA TABELELOR:

1. CLASA
2. UTILIZATOR
3. CAMPANIE:
4. SEDINTA:
5. NEAM:
6. BONUS_ABILITATI:
7. ARBORE_DE_ABILITATI:
8. PERSONAJ:
9. PERSONAJ_CLASA:
10. PERSONAJ_CAMPANIE:
11. PARTICIPARE:
12. ARHETIP:
13. SUBNEAM:
14. ATRIBUT:
15. ECHIPAMENT:
16. VRAJA:
17. ITEM_NEAM:
18. ITEM_SUBNEAM:
19. ITEM_CLASA:
20. ITEM_ARHETIP:
21. COMPONENTA:
22. COMPONENTA_VRAJA:

INSERAREA ÎN TABELE:

1. CLASA:
2. UTILIZATOR:
3. CAMPANIE:
4. NEAM:

5. BONUS ABILITATI: (exact 30 de linii; regulile astăzi chiar sunt din D&D totuși)

9. PERSONAJ_CAMPANIE:

10. ARHETIP:

11. SUBNEAM:

12. ATRIBUT:

13. ECHIPAMENT:

14. VRAJA:

15. ITEM_NEAM:

16. ITEM_SUBNEAM:

17. ITEM_CLASA:

18. ITEM_ARHETIP:

19. COMPONENTA:

20. COMPONENTA_VRAJA:

21. SEDINTĂ:

22. PARTICIPARE:

12. Formulati în limbaj natural și implementați 5 cereri SQL complexe ce vor utiliza, în ansamblul lor, următoarele elemente:

I. Prima Cerere

(exercițiu din feedback, îndeplinește subpunctele a și b)

Enunt:

Pentru fiecare personaj din clasa ‘vrajitor’ care a participat la ședinte din campania ‘Werewolves’ să se afișeze lista atributelor de tip vrajă pe care le are.

Rezolvare în cod sql:

Output:

II. A Doua Cerere

Enunt:

Rezolvare în cod sql:

Output:

III. A Treia Cerere

Enunt:

Rezolvare în cod sql:

Output:

IV. A Patra Cerere

Enunt:

Rezolvare în cod sql:

Output:

V. A Cincea Cerere

Enunt:

Rezolvare în cod sql:

Output:

13. Implementarea a 3 operații de actualizare și de suprimare a datelor utilizând subcereri.

I. Prima Operatie:

II. A doua Operatie:

PrintScreen:

III. A treia Operatie:

14. Crearea unei vizualizări complexe. Dați un exemplu de operație LMD permisă pe vizualizarea respectivă și un exemplu de operație LMD nepermisă.
15. Formulați în limbaj natural și implementați în SQL: o cerere ce utilizează operația outer-join pe minimum 4 tabele, o cerere ce utilizează operația division și o cerere care implementează analiza top-n.
- I. Prima Cerere (utilizează operația outer join)
- II. A doua Cerere (utilizează division)
- III. A treia Cerere (utilizând analiza top n)
16. Optimizarea unei cereri, aplicând regulile de optimizare ce derivă din proprietățile operatorilor algebrei relationale. Cererea va fi exprimată prin expresie algebraică, arbore algebraic și limbaj (SQL), atât anterior cât și ulterior optimizării.
17. a. Realizarea normalizării BCNF, FN4, FN5.
- b. Aplicarea denormalizării, justificând necesitatea acesteia.

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

Dungeons and Dragons este un joc de aventură ce se joacă cu o foaie și un pix împreună cu un grup de prieteni. În acesta fiecare persoană are nevoie de un personaj și de abilitățile lui pentru a putea participa. Acest proiect este menit să ofere o metodă mai ușoară atât de structurare a tuturor atributelor mai multor personaje într-o singură bază de date, cât și de organizare a ședințelor în care se joacă propriu-zis D&D.

Fiecare *jucător* va avea minim un personaj care, în funcție de regulile jocului, va avea mai multe vrăji, abilități, bucăți de echipament. Fiecare vrajă are propriile ei atribute, la fel și fiecare abilitate și aşa mai departe.

Pentru a înregistra un personaj nou (activ în paralel cu un alt personaj) un *jucător* trebuie să participe la o nouă *campanie* (adică un nou joc). Fiecare *campanie* trebuie să aibă mai mulți *jucători* și obligatoriu un singur *dungeon master* (practic un *organizator* al acelei povești/ al aceluia joc). Deși un *jucător* poate avea un singur *personaj activ* într-o campanie, el poate avea mai multe *personaje (inactive)* per total în aceeași *campanie*. *Organizatorul* necesită mai multe *personaje active* într-o singură campanie deoarece *personajele jucătorilor* interacționează cu mai multe *personaje ale organizatorului* într-o singură *campanie* sau chiar într-o singură *ședință* (dat fiindcă jocul de D&D reprezintă o mare poveste în care *personajele jucătorilor* sunt protagoniștii, iar *personajele organizatorului* sunt personajele secundare).

Fiecare *jucător* poate deveni *organizator* pentru o altă campanie (în cazul uneia deja existente *organizatorul* inițial ar trebui să se retragă sau să devină *jucător*), iar în timp ce o campanie este în desfășurare se pot organiza ședințe în care toți jucătorii se strâng fie să joace propriu-zis fie să își detalieze attributele personajelor (un personaj va avea câteva câmpuri ce nu sunt obligatorii astfel încât un jucător să poată apartine unei campanii fără ca toate detaliile personajului să fie specificate).

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

Un jucător poate avea mai multe personaje-jucător (player-characters).

Un personaj-jucător poate apartine unui singur jucător.

O campanie are mai multe personaje-jucător.

Un personaj-jucător poate apartine mai multor campanii.

O campanie are mai mulți jucători.

Un jucător poate avea mai multe campanii.

O campanie poate avea mai multe ședințe.

O ședință apartine unei unice campanii.

Un personaj poate participa la mai multe ședințe.

O ședință are mai multe personaje.

Un personaj are mai multe atribute.

Un atribut poate fi deținut de mai multe personaje.

Un personaj poate avea mai multe clase.

O clasă poate apartine mai multor personaje.

Un personaj are doar un singur neam.

Un neam poate apartine mai multor personaje.

O clasă poate avea mai multe atribute.

Un atribut poate apartine mai multor clase.

Un neam poate avea mai multe atribute.

Un atribut poate apartine mai multor neamuri.

Un personaj-organizator apartine unui singur organizator.

Un organizator poate avea mai multe personaje-organizator.

Un personaj are un singur arbore de abilități.

Un arbore de abilități apartine unui singur personaj.

Un neam poate avea mai multe sub-neamuri.

Un sub-neam trebuie să fie al unui singur neam.

O clasă poate avea mai multe arhetipuri.

Un arhetip trebuie să fie al unei unice clase.

Un arbore de abilități are o singură tabelă de bonus abilități.

O tabelă de bonus abilități are mai mulți arbori de abilități.

O componentă are mai multe vrăji.

O vrajă are mai multe componente

3. Descrierea entităților, incluzând precizarea cheii primare.

ENTITATE	CHEIE PRIMARĂ	OBSERVAȚII
utilizator/ user	utilizator_id	Informații despre useri în general.
jucător/ player	jucător_id	Informații detaliate despre un subtip de useri.
organizator/ dungeon master	dungeon_master_id	informații detaliate despre un subtip de useri.
personaj/ character	character_id	informații generale despre personaj.
personaj al jucătorului/ player-character	player_character_id	Subentitate a personajelor. Tabelă ce se referă la personajele jucate și create de useri. Reprezintă tabela de organizare a tuturor atributelor personajului.
personaj al organizatorului/ (non-playable character)	organizer_character_id	Subentitate a personajelor. Acestea sunt create exclusiv de dungeon master pentru a participa la povestea campaniei, nu pot fi jucate de către un player.
campanie/ campaign	campaña_id	Informații despre jocul la care participă userii.
personaj-campanie	campaign_character_id	Tabelă asociativă ce permite înscrierea mai multor personaje-jucător într-o singură campanie
personaj-clasă	character_class_id	Tabelă asociativă ce facilitează apariția multiclassing-ului (mod de construire al personajului în D&D).
ședință/ session	session_id, date	Şedințele propriu-zise (cu data, locație etc.) ale utilizatorilor.
atribut/ attribute	attribute_id	Informații generale despre atrbute
vrajă/ spell	spell_id	Un subtip al unui atribut detaliat, ce conține informații specifice.
echipament/ equipment	equipment_id	Un subtip al unui atribut detaliat, ce conține informații specifice.
specialitatea personajului/ class	class_id	Tabelă menită să păstreze toate detaliiile unei personajului. Clasa unui personaj determină abilitățile și echipamentul personajului.
neam/ race	race_id	Tabelă menită să păstreze toate detaliiile

		neamului personajului. Neamul personajului va avea anumite bonusuri la arborele de abilități sau la cunoștințele acestuia. (de exemplu un elf va ști limba sa natală)
arbore de abilități/ skill tree	abilitati_id	tabelă ce reține valorile numerice specifice skill-urilor/abilităților personajului. (de exemplu în D&D există abilitatea de magie, de dexteritate etc.)
arhetip	hetip_id	tabelă ce păstrează specializarea clasei personajului. (de exemplu un vrăjitor se poate specializa pe anumite tipuri de vrăjitori) în intermediul acestei tabele se adaugă attribute specifice arhetipului personajului.
subneam	bneam_id	tabelă similară cu cea de arhetipuri, însă pentru neamuri. Fiecare neam are mai multe tipuri de subneamuri (de exemplu un elf poate fi atât de pădure cât și provenit din alte zone). În funcție de subneam personajul poate primii attribute în plus, diferite față de celealte subneamuri ale aceluiași neam.
item clasă/ class item	asă_id, atribut_id	tabelă asociativă ce apare din relația many-to-many dintre clasă și atribut
item neam/ race item	neam_id, atribut_id	tabelă asociativă ce apare din relația many-to-many dintre neam și atribut
item arhetip	hetip_id, atribut_id	tabelă asociativă ce apare din relația many-to-many dintre arhetip și atribut
item subneam	bneam_id, atribut_id	tabelă asociativă ce apare din relația many-to-many dintre subneam și atribut
componentă	componenta_id	Fiecare vrăjă are nevoie de una sau mai multe componente specifice.
componentă vrăjă	componenta_id, atribut_id	tabelă asociativă pentru a facilita prezența a mai multor componente într-o vrăjă.
bonus abilități	bonus_id	lookup table făcut pentru a putea determina bonusul în funcție de scorul aflat în tabela de arbore de abilități

4. Descrierea relațiilor, incluzând precizarea cardinalității acestora.

RELAȚIE	CARDINALITATE	OBSERVAȚII
are	Jucător - Personaj Jucător one-to-many, Organizator - Personaj Organizator one-to-many, Personaj - ArboreDeAbilități one-to-one Campanie - Ședință one-to-many, Personaj - Neam many-to-one Personaj - Clasă many-to-many Clasă - Arhetip one-to-many Neam - Subneam one-to-many Componentă - Vrajă many-to-many Bonus pentru Abilități - Arbore de abilități one-to-many	Un dungeon master poate include în campaniile sale orice NPC, chiar și dacă nu îi aparțin (i.e. NPC creat de alt DM, deținut în baza de date de acel DM). Un Personaj va avea atribuții specifice determinate de clasă, neam și nivel.
organizează	Organizator - Campanie one-to-many	Un dungeon master poate crea mai multe campanii, să o campanie nu are decât un singur dungeon master organizator).
apartine	Personaj Jucător - Campanie many-to-many Atribut - (Clasă, Neam, Arhetip, Subneam) many-to-many	Un atribut aparține mai multor clase sau neamuri. O clasă sau neam are mai multe atribuții specifice.
participă	Personaj - Ședință many-to-many	Pentru a înregistra toate personajele care participă la ședință.

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

UTILIZATOR

Atribut	Tip	Valori posibile și valori default	Constrângeri	Observații
utilizator_id	INT		PK	
tip_utilizator	VARCHAR(20)	jucător/organizator	NOT NULL	Determină dacă utilizatorul este organizator sau jucător. Organizatorul primește True, iar jucătorul normal False.
nume	VARCHAR(255)			
prenume	VARCHAR(255)			
username	VARCHAR(255)		NOT NULL, UNIQUE	
parola	VARCHAR(255)		NOT NULL, lungime minimă de 5 caractere	
data_înregistrare	DATE			
email	VARCHAR(255)		NOT NULL	
număr_telefon	NUMBER(10)			Optional

CAMPANIE

Atribut	Tip	Valori posibile și valori default	Constrângeri	Observații
campanie_id	INT		PK	
utilizator_id	INT		FK	
nume	VARCHAR(255)			
data_inceput	DATE		NOT NULL	Teoretic ar trebui completat în mod automat la crearea unei noi campanii
data_final	DATE			

PERSONAJ

Atribut	Tip	Valori posibile și valori default	Const rângere	Observații
personaj_id	INT		PK	
utilizator_id	INT		FK	
abilitati_id	INT		FK	
neam_id	INT		FK	
nume	VARCHAR(255)		NOT NULL	
tip_personalitate	VARCHAR(255)	legiuit-bun (lawful good), legiut-neutru (lawful neutral), legiuit-malefic (lawful evil), neutru-bun (neutral good), complet-neutru (true neutral), neutru-malefic haotic-bun (chaotic good), haotic-neutru (chaotic neutral), haotic-malefic(chaotic evil)		Am tradus numele din engleză în română și, deși sună ciudat, teoretic sunt termenii specific jocului

PERSONAJ_CLASĂ

Atribut	Tip	Valori posibile și valori default	Constrângeri	Observații
personaj_id	INT		PK, FK	
clasa_id	INT		PK, FK	
nivel	INT	Valori de la 1 la 20	NOT NULL	Un personaj poate avea mai multe clase, iar fiecare clasă are un nivel separat al său (pentru a facilita multiclassing-ul)

PERSONAJ_CAMPANIE

Atribut	Tip	Valori posibile și valori default	Constrângeri	Observații
personaj_id	INT		PK, FK	
campanie_id	INT		PK, FK	

ŞEDINȚĂ

Atribut	Tip	Valori posibile și valori default	Constrângeri	Observații
sedinta_id	INT		PK	
campanie_id	INT		FK	A doua jumătate din primary key-ul compus
data_inceput	DATE		Poate exista doar o singura sedinta a unei campanii intr-o zi. Nu vor fi doua sedinte cu campania x in aceeasi zi	Prima jumătate din primary key-ul compus
data_final	DATE			Se completează doar atunci când o ședință durează peste ora 23:59 a datei de început
ora_inceput	VARCHAR(5)			Voi folosi o sintaxă de genul următor pentru a extrage ora exactă: TO_CHAR(SYSDATE, 'HH24:MI');
ora_final	VARCHAR(5)			

PARTICIPARE

Atribut	Tip	Valori posibile și valori default	Constrângeri	Observații
personaj_id	INT		PK, FK	
sedinta_id	INT		PK, FK	
prezență	VARCHAR(2)	da/nu	NOT NULL	

CLASĂ

Atribut	Tip	Valori posibile și valori default	Constrângeri	Observații
clasa_id	INT		PK	
nume	VARCHAR(255)		NOT NULL, UNIQUE	
descriere	CLOB (medium/long text)			
dificultate	VARCHAR(12)	Începător, Experimentat, Expert	NOT NULL	De obicei clasele care conțin mai multă magie sunt mai dificile pentru jucătorii noi

NEAM

Atribut	Tip	Valori posibile și valori default	Constrângeri	Observații
neam_id	INT		PK	
nume	VARCHAR(255)		NOT NULL	
descriere	CLOB			
tip	VARCHAR(255)	uman, gigant, elemental, plantă, aberație, diavol, monstruozitate, dragon, strigoi, zână,	NOT NULL	

ARBORE DE ABILITĂȚI

Atribut	Tip	Valori posibile și valori default	Constrângeri	Observații
abilitati_id	INT		PK	
scor_putere	INT	Valori de la 1 la 30	NOT NULL	Un personaj are 6 mari scoruri care ajută la determinarea succesului unei anumite acțiuni. (de exemplu, în cazul acestui scor, cu cât e mai mare cu atât îi este mai ușor unui personaj să facă lucruri ce necesită forță fizică)
scor_charisma	INT	Valori de la 1 la 30	NOT NULL	Acest scor reprezintă cât de ușor îi este unui personaj să fie charismatic (cât de ușor poate convinge lumea, poate minți, poate intimida etc.)
scor_constituție	INT	Valori de la 1 la 30	NOT NULL	Valoare ce determină cât de mult HP (health point = puncte de sănătate = viață) are un personaj.
scor_inteligenta	INT	Valori de la 1 la 30	NOT NULL	Atribut ce vede cât de mult și de ușor poate reține personajul informații. Dacă acesta își poate aduce aminte anumite informații citite. Dacă acesta poate observa și înțelege magia din jurul său.
scor_intelepciune	INT	Valori de la 1 la 30	NOT NULL	Scor ce determină inteligența nativă a unui personaj. Adaugă bonusuri asupra

				acțiunilor ce implică percepția personajului. De exemplu: dacă acesta poate observa detalii mărunte, dacă își dă seama când minte cineva etc.
scor_dexteritate	INT	Valori de la 1 la 30	NOT NULL	Scor ce reprezintă îndemânarea unui personaj.

BONUS ABILITĂȚI

Atribut	Tip	Valori posibile și valori default	Constrângeri	Observații
scor	INT	Valori de la 1 la 20	PK	Pentru lookup table-ul <i>bonus abilități</i> voi lega scorul cu scorurile specifice din <i>arbore abilități</i> pentru a determina ce bonus trebuie adăugat la zarurile date de utilizator
bonus	INT	Valori de la -4 la +4	NOT NULL	

ATRIBUT

Atribut	Tip	Valori posibile și valori default	Constrângeri	Observații
atribut_id	INT		PK	
nume	VARCHAR(255)		NOT NULL	
descriere	CLOB (TEXT)			
bonus	CLOB			

ECHIPAMENT (subentitate *atribut*, moștenește attributele sale pe lângă cele specificate mai jos)

Atribut	Tip	Valori posibile și valori default	Constrângeri	Observații
greutate	INT			Greutatea este păstrată în kilograme
putere	VARCHAR(6)			Câmp în care se rețin valori de genul: 1d6, 2d4. Acestea reprezintă numărul de zaruri și felul lor. (trebuie aruncate pentru a calcula puterea loviturii personajului)
tip_echipament	VARCHAR(6)			Tipul echipamentului este necesar deoarece anumite clase pot folosi tipuri diferite de echipament. (de exemplu un paladin poate folosi toate tipurile de armura, dar un vrajitor nu)
tip_lovitura	VARCHAR(10)			Apare necesitatea de a pastra tipul de lovitura al armei deoarece anumite personaje pot fi mai vulnerabile sau mai rezistente la tipuri diferite de lovituri.
distanța_lovitura	INT			Descrie de la ce distanță poate fi lovit un inamic cu această armă, de exemplu cu un arc vor fi în jur de 30-40 de metrii, dar cu un pumnal va fi în jur de un metru. Distanța este păstrată în metrii.

VRAJĂ (subentitate *atribut*, moștenește atributele sale pe lângă cele specificate mai jos)

Atribut	Tip	Valori posibile și valori default	Constrângeri	Observații
putere	VARCHAR(10)			Similară cu puterea unui tip de echipament însă se modifică în funcție de nivelul cu care vrăja este făcută
tip_lovitura	VARCHAR(50)			Similar cu atributul echipamentului, însă sunt valori posibile diferite
durată	VARCHAR(50)	un turn, un minut, o ora etc.		
timp_necesar	VARCHAR(50)	instantaneu, un minut, 10 minute, o ora, etc.		
școală	VARCHAR(50)			In engleză school, atributul descrie și caracterizează felul de magie folosit în vrăja respectivă. (iluzie, tranzmutație, divinatie etc.)

COMPONENTĂ

Atribut	Tip	Valori posibile și valori default	Constrângeri	Observații
componenta_id	INT		PK	
tip_componenta	VARCHAR(255)		NOT NULL	O vrajă are nevoie de diverse componente pentru a putea fi folosită. De exemplu unele au nevoie doar de o propozitie sau de anumite sunete, altele au nevoie de materiale specifice sau semne gesticulate etc.
descriere	CLOB			Atribut ce permite stocarea unei descrieri specifice componentei.

COMPONENTĂ VRAJĂ

Atribut	Tip	Valori posibile și valori default	Constrângeri	Observații
componenta_id	INT		PK	
atribut_id	INT		PK	

ARHETIP

Atribut	Tip	Valori posibile și valori default	Constrângeri	Observații
arhetip_id	INT		PK	
clasa_id	INT		FK	
nume	VARCHAR(255)		NOT NULL	
descriere	CLOB/TEXT			

SUBNEAM

Atribut	Tip	Valori posibile și valori default	Constrângeri	Observații
subneam_id	INT		PK	
neam_id	INT		FK	
nume	VARCHAR(255)		NOT NULL	
descriere	TEXT			

ITEM NEAM

Atribut	Tip	Valori posibile și valori default	Constrângeri	Observații
neam_id	INT		PK, FK	
atribut_id	INT		PK, FK	

ITEM SUBNEAM

Atribut	Tip	Valori posibile și valori default	Constrângerি	Observații
subneam_id	INT		PK, FK	
atribut_id	INT		PK, FK	

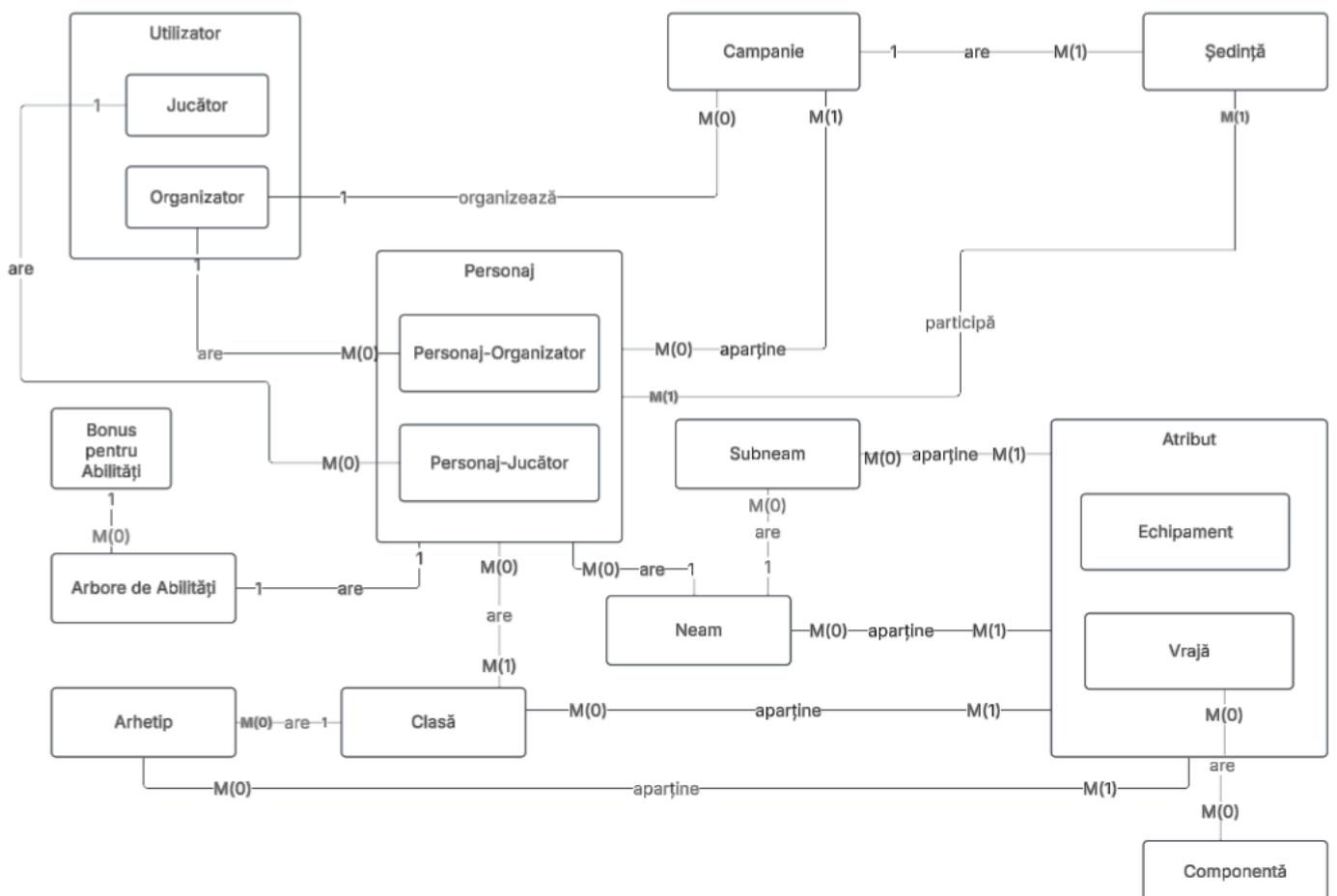
ITEM CLASĂ

Atribut	Tip	Valori posibile și valori default	Constrângerি	Observații
clasa_id	INT		PK, FK	
atribut_id	INT		PK, FK	

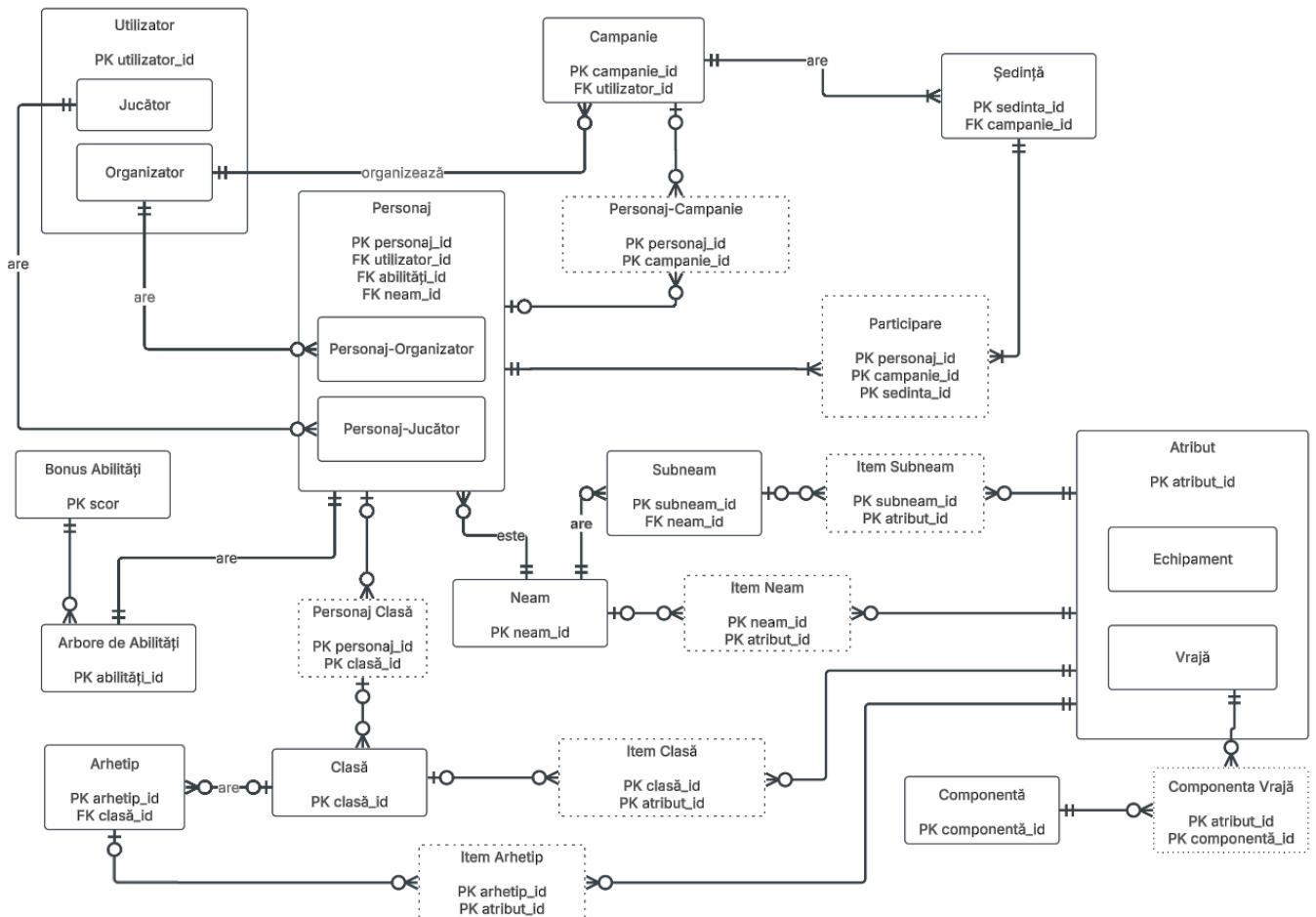
ITEM ARHETIP

Atribut	Tip	Valori posibile și valori default	Constrângerি	Observații
arhetip_id	INT		PK, FK	
atribut_id	INT		PK, FK	

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



7. Realizarea diagramei conceptuale corespunzătoare diagramei entitate-relație proiectate la punctul 6. Diagrama conceptuală obținută trebuie să conțină minimum 7 tabele (fără considerarea subentităților), dintre care cel puțin un tabel asociativ.



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

UTILIZATOR(#utilizator_id, tip_utilizator, nume, prenume, username, data_inregistrare, email, numar_telefon)
JUCĂTOR(#utilizator_id, tip_utilizator, nume, prenume, username, data_inregistrare, email, numar_telefon)
ORGANIZATOR(#utilizator_id, tip_utilizator, nume, prenume, username, data_inregistrare, email, numar_telefon)
CAMPANIE(#campanie_id,# utilizator_id FK, nume, data_inceput, data_final)
ȘEDINȚĂ(#sedinta_id, #campanie_id FK, data_inceput, data_final, ora_inceput, ora_final)
PERSONAJ(#personaj_id, #utilizator_id FK, #abilitati_id FK, #neam_id FK, nume, tip_personalitate)
PERSONAJ_CLASA(#personaj_id, #clasa_id, nivel)
PERSONAJ_CAMPANIE(#personaj_id, #campanie_id)
PARTICIPARE(#personaj_id, #sedinta_id, prezenta)
ARBORE_ABILITĂȚI(#abilitati_id, scor_putere, scor_charisma, scor_constitutie, scor_inteligenta, scor_intelepciune, scor_dexteritate)
BONUS_ABILITĂȚI(#scor, bonus)
CLASĂ(#clasa_id, nume, descriere, dificultate)
NEAM(#neam_id, nume, descriere)
ARHETIP(#arhetip_id, #clasa_id FK, nume, descriere)
SUBNEAM(#subneam_id, neam_id FK, nume, descriere)
ATRIBUT(#atribut_id, nume, descriere, bonus)
ECHIPAMENT(#atribut_id, nume, descriere, greutate, putere, tip_echipament, tip_lovitura)
VRAJĂ(#atribut_id, nume, descriere, putere, tip_lovitura, durata, timp_necesar, scoala)
COMPONENTĂ(#componenta_id, tip_componenta, descriere)
COMPONENTĂ_VRAJĂ(#componenta_id, #atribut_id)
ITEM_CLASĂ(#clasa_id PK, atribut_id PK)
ITEM_NEAM(#neam_id PK, atribut_id PK)
ITEM_ARHETIP(#arhetip_id, #atribut_id)
ITEM_SUBNEAM(#subneam_id, #atribut_id)

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

- *Exemplu non-FN1:*

VRAJA(atribut_id, nume, descriere, putere, tip_lovitura, durata, timp_necesar, componente)

Se observă că atributul *componente* al tabelei este multiplu, deci VRAJA nu este în FN1. Pentru a aduce relația în prima formă normală se creează o nouă tabelă numită COMPONENTA care va fi legată de tabela vrajă printr-o relație many-to-many (deoarece mai multe componente pot apartine mai multor vrăji și mai multe vrăji pot avea mai multe componente) și, deci, rezultă:

COMPONENTA(#componenta_id, tip_componenta)

VRAJA(#atribut_id, nume, descriere, putere, tip_lovitura, durata, timp_necesar)

În rest tabela vraja se află în FN2 și FN3 deoarece toate atributele depind de întreaga cheie și doar de aceasta. (nu există dependențe parțiale deoarece este o singură cheie primară și nu există nici dependențe tranzitive).

- *Exemplu non-FN2:*

PARTICIPARE(#personaj_id, #campanie_id, #sedinta_id, prezenta, nume_personaj, nume_campanie).

Tabela nu este în FN2 deoarece apar două dependențe parțiale:

personaj_id → nume_personaj

campanie_id → nume_campanie

(personaj_id, campanie_id, sedinta_id) → prezenta

Pentru a aduce tabela în FN2 se desparte în:

PARTICIPARE(personaj_id PK, campanie_id PK, sedinta_id PK, prezenta)

PERSONAJ(personaj_id PK, utilizator_id FK, abilitati_id FK, clasa_id FK, neam_id FK, nume, nivel, tip_personalitate)

CAMPANIE(campanie_id PK, utilizator_id FK, nume, data_inceput, data_fina)

Atributele nume_personaj și nume_campanie se vor găsi acum în tabelele corespunzătoare.

- *Exemplu non-FN3, dar în FN2:*

ARBORE_ABILITĂȚI(#abilitati_id, bonus_personaj, scor_putere, scor_charisma, scor_constitutie, scor_inteligenta, scor_intelepciune, scor_dexteritate, bonus_putere, bonus_charisma, bonus_constitutie, bonus_inteligenta, bonus_intelepciune, bonus_dexteritate)

Tabela ARBORE DE ABILITĂȚI se află în FN2 (având doar o cheie principală), însă se observă următoarele dependențe tranzitive:

abilitati_id → scor_putere → bonus_putere

abilitati_id → scor_charisma → bonus_charisma

abilitati_id → scor_constitutie → bonus_constitutie
abilitati_id → scor_inteligenta → bonus_inteligenta
abilitati_id → scor_intelepciune → bonus_intelepciune
abilitati_id → scor_dexteritate → bonus_dexteritate

O observație necesară legată de regulile jocului de D&D: fiecare scor (de la 1 la 20) determină un bonus specific acelei abilități, de exemplu scorul 10 are bonusul +0, scor 11 și 12 au bonusul +1, scorul 13 și 14 au bonusul +2, scorul 9 și 8 au bonusul -1 și aşa mai departe.

Pentru a aduce tabela în FN3 o despărțim în:

ARBORE DE ABILITĂȚI(#abilitati_id, bonus_personaj, scor_putere, scor_charisma, scor_constitutie, scor_inteligenta, scor_intelepciune, scor_dexteritate,
BONUS ABILITĂȚI(#scor, bonus)

Astfel tabela BONUS ABILITĂȚI reprezintă un lookup table (tabel de căutare) ce va reține bonusurile pentru fiecare scor posibil și care va fi folosită la calcularea bonusurilor specifice abilităților.

10. Crearea unei sevențe ce va fi utilizată în inserarea înregistrărilor în tabele.

Cod SQL:

```
CREATE SEQUENCE clase_si_arhetipuri
START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE neamuri_si_subneamuri
START WITH 1 INCREMENT BY 1;

CREATE SEQUENCE componente
START WITH 1 INCREMENT BY 1;
```

PrintScreen:

The screenshot shows the Oracle SQL Developer interface with two panes. The top pane displays the creation of three sequences with their respective start values and increments. The bottom pane shows the execution of these statements and the resulting output, including the current schema being set to SYSTEM and the completion times for each sequence creation.

```
301 ✓ CREATE SEQUENCE clase_si_arhetipuri
302     START WITH 1 INCREMENT BY 1;
303
304 ✓ CREATE SEQUENCE neamuri_si_subneamuri
305     START WITH 1 INCREMENT BY 1;
306

SYSTEM> alter session set current_schema = SYSTEM
[2025-05-18 18:14:11] completed in 2 ms
SYSTEM> CREATE SEQUENCE clase_si_arhetipuri
    START WITH 1 INCREMENT BY 1
[2025-05-18 18:14:11] completed in 6 ms
SYSTEM> CREATE SEQUENCE neamuri_si_subneamuri
    START WITH 1 INCREMENT BY 1
[2025-05-18 18:14:11] completed in 3 ms

RA 295 ✓ CREATE SEQUENCE componente
296     START WITH 1 INCREMENT BY 1;
297

[Output]
[2025-05-20 01:03:03] 10 rows retrieved starting at row 1
SYSTEM> CREATE SEQUENCE componente
    START WITH 1 INCREMENT BY 1
[2025-05-20 01:04:51] completed in 10 ms
```

11. 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; maxim 30 de înregistrări în fiecare tabel).

CREAREA TABELELOR:

1. CLASA

Cod Sql:

```
CREATE TABLE clasa (
    clasa_id INT PRIMARY KEY,
    nume VARCHAR(255) NOT NULL UNIQUE,
    descriere CLOB,
    dificultate VARCHAR(13) NOT NULL,
    CONSTRAINT c_dificultate CHECK (
        dificultate='incepator' OR
        dificultate='experimentat' OR
        dificultate='expert'
    )
);
```

PrintScreen:

The screenshot shows the Oracle SQL Developer interface. On the left, there's a tree view of database objects under 'Server Objects'. In the center, a code editor window displays the SQL script for creating the 'clasa' table. On the right, the 'Services' panel shows a transaction history with several entries, including the creation of the table.

```
44
45 ✓ CREATE TABLE clasa (
46     clasa_id INT PRIMARY KEY,
47     nume VARCHAR(255) NOT NULL UNIQUE,
48     descriere CLOB,
49     dificultate VARCHAR(13) NOT NULL,
50     CONSTRAINT c_dificultate CHECK (
51         dificultate='incepator' OR
52         dificultate='experimentat' OR
53         dificultate='expert'
54     )
55 );
```

Services

- Tx: + ⌂ ⌂ | ⌂ ⌂
- ✓ Database @127.0.0.1
- ☰ console_1 66 ms
- ☰ console 96 ms

```
[2025-05-18 20:03:58] completed in 66 ms
SYSTEM> CREATE TABLE clasa (
    clasa_id INT PRIMARY KEY,
    nume VARCHAR(255) NOT NULL UNIQUE,
    descriere CLOB,
    dificultate VARCHAR(13) NOT NULL,
    CONSTRAINT c_dificultate CHECK (
        dificultate='incepator' OR
        dificultate='experimentat' OR
        dificultate='expert'
    )
)
```

2. UTILIZATOR

Cod Sql:

```
CREATE TABLE utilizator (
    utilizator_id INT GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,
    tip_utilizator VARCHAR(20) NOT NULL,
    nume VARCHAR(255),
    prenume VARCHAR(255),
    username VARCHAR(255) NOT NULL UNIQUE,
    parola VARCHAR(255) NOT NULL,
    data_inregistrare DATE,
    email VARCHAR(255) NOT NULL,
    numar_telefon NUMBER(10),
    --CONSTRAINT u_username UNIQUE(username),
    CONSTRAINT u_tip_utilizator CHECK(tip_utilizator='jucator'
    OR tip_utilizator='organizator'),
    CONSTRAINT u_parola CHECK(length(parola) > 5)
);
```

PrintScreen:

The screenshot shows the Oracle SQL Developer interface. In the Database Explorer, a connection to '127.0.0.1' is selected. The 'SYSTEM' schema is expanded, showing various system tables like AQ\$INTERNET_AGENT_PRIVS, AQ\$INTERNET_AGENTS, AQ\$KEY_SHARD_MAP, AQ\$QUEUE_TABLES, AQ\$QUEUES, AQ\$SCHEDULES, and LOGMNR_* tables. In the central workspace, a code editor window titled 'console' is open, displaying the SQL script for creating the 'utilizator' table. The code includes constraints for unique usernames and specific tip_utilizator values ('jucator' or 'organizator'), and a length constraint for the password. A note at the bottom of the code indicates it uses single table inheritance for jucator and organizator. Below the code editor, the SQL command is executed in the 'Tx' tab, showing the execution progress and completion message '[2025-05-18 20:03:58] completed in 66 ms'. The Services panel on the left shows the database connection status.

```
CREATE TABLE utilizator (
    utilizator_id INT GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,
    tip_utilizator VARCHAR(20) NOT NULL,
    nume VARCHAR(255),
    prenume VARCHAR(255),
    username VARCHAR(255) NOT NULL UNIQUE,
    parola VARCHAR(255) NOT NULL,
    data_inregistrare DATE,
    email VARCHAR(255) NOT NULL,
    numar_telefon NUMBER(10),
    --CONSTRAINT u_username UNIQUE(username),
    CONSTRAINT u_tip_utilizator CHECK(tip_utilizator='jucator' OR tip_utilizator='organizator'),
    CONSTRAINT u_parola CHECK(length(parola) > 5)
);
-- folosește single table inheritance pentru jucator și organizator
CREATE TABLE campanie (
    campanie_id INT GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,
```

3. CAMPANIE:

Cod Sql:

```
CREATE TABLE campanie (
    campanie_id INT GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,
    utilizator_id INT,
    nume VARCHAR(255),
    data_inceput DATE NOT NULL,
    data_final DATE,
    FOREIGN KEY (utilizator_id)
        REFERENCES utilizator(utilizator_id)
) ;
```

PrintScreen:

The screenshot shows a database management interface with a sidebar on the left containing 'collection types' (6), 'synonyms' (8), and 'Server Objects'. The main area displays the SQL code for creating the 'campanie' table, with line numbers 25 through 32 visible. The code defines columns for 'campanie_id' (primary key, identity), 'utilizator_id' (foreign key), 'nume' (VARCHAR(255)), 'data_inceput' (DATE NOT NULL), and 'data_final' (DATE). A note indicates that the 'data_final' column is missing. The code concludes with a closing parenthesis and a semicolon. Below this, another table definition for 'sedinta' is partially visible. On the left, under 'Services', there is a 'Transactions' section showing a transaction named 'console' with a status of 'Completed' and a duration of 24 ms. The transaction tree shows 'Database' > '@127.0.0.1' > 'console_1' (66 ms) > 'console' (79 ms). The transaction history shows the execution of the 'CREATE TABLE campanie' command.

```
25
26
27
28
29
30
31
32
33
34
35
```

```
CREATE TABLE campanie (
    campanie_id INT GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,
    utilizator_id INT,
    nume VARCHAR(255),
    data_inceput DATE NOT NULL,
    data_final DATE, |  
FOREIGN KEY (utilizator_id)  
    REFERENCES utilizator(utilizator_id)  
);  
CREATE TABLE sedinta (
```

```
Tx: + ⌂ | ⌂ × [2025-05-18 20:17:22] completed in 24 ms  
SYSTEM> CREATE TABLE campanie (  
    campanie_id INT GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,  
    utilizator_id INT,  
    nume VARCHAR(255),  
    data_inceput DATE NOT NULL,  
    data_final DATE,  
    FOREIGN KEY (utilizator_id)  
        REFERENCES utilizator(utilizator_id)  
)  
[2025-05-18 20:20:01] completed in 23 ms
```

4. SEDINTA:

Cod Sql:

```
CREATE TABLE sedinta (
    sedinta_id INT GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,
    campanie_id INT NOT NULL,
    data_inceput DATE NOT NULL,
    data_final DATE,
    ora_inceput DATE,
    ora_final DATE,
    FOREIGN KEY (campanie_id)
        REFERENCES campanie(campanie_id)
);
```

PrintScreen:

The screenshot shows the Oracle SQL Developer interface. On the left, there's a sidebar with navigation links: object types (6), collection types (6), synonyms (8), and Server Objects. The main area is a code editor with the following content:

```
31 | REFERENCES Utilizator(Utilizator_id)
32 );
33
34 ✓ CREATE TABLE sedinta (
35     sedinta_id INT GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,
36     campanie_id INT NOT NULL,
37     data_inceput DATE NOT NULL,
38     data_final DATE,
39     ora_inceput DATE,
40     ora_final DATE,
41     FOREIGN KEY (campanie_id)
42         REFERENCES campanie(campanie_id)
43 );
44
```

Below the code editor, the SQL history pane shows the command executed:

```
SYSTEM> CREATE TABLE sedinta (
    sedinta_id INT GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,
    campanie_id INT NOT NULL,
    data_inceput DATE NOT NULL,
    data_final DATE,
    ora_inceput DATE,
    ora_final DATE,
    FOREIGN KEY (campanie_id)
        REFERENCES campanie(campanie_id)
)
[2025-05-18 20:25:17] completed in 23 ms
```

The bottom left of the interface shows a services tree with a node expanded to show 'console' and 'console 1'.

5. NEAM:

Cod Sql:

```
CREATE TABLE neam (
    neam_id INT PRIMARY KEY,
    nume VARCHAR(255) NOT NULL,
    descriere CLOB,
    tip VARCHAR(255) NOT NULL,
    CONSTRAINT n_tip CHECK (
        tip='uman' OR
        tip='gigant' OR
        tip='elemental' OR
        tip='planta' OR
        tip='aberatie' OR
        tip='diavol' OR
        tip='monstruoza' OR
        tip='dragon' OR
        tip='strigoi' OR
        tip='zana'
    )
);
```

PrintScreen:

The screenshot shows a SQL database management interface with a tree view of objects on the left and a code editor on the right.

Object Explorer (Left):

- System
- Tables (136)
- Views (10)
- Routines (6)
- Sequences (9)
- Object Types (6)
- Collection Types (6)
- Synonyms (8)
- Server Objects

Code Editor (Right):

```
CREATE TABLE neam (
    neam_id INT PRIMARY KEY,
    nume VARCHAR(255) NOT NULL,
    descriere CLOB,
    tip VARCHAR(255) NOT NULL,
    CONSTRAINT n_tip CHECK (
        tip='uman' OR
        tip='gigant' OR
        tip='elemental' OR
        tip='planta' OR
        tip='aberatie' OR
        tip='diavol' OR
        tip='monstruoza' OR
        tip='dragon' OR
        tip='strigoi' OR
        tip='zana'
    )
);
```

Services (Bottom):

- + ⊕ ⊖ ⊗ ⊖ ⊖
- Database @127.0.0.1
- Console
- Console
- Console

[2025-05-18 20:31:03] completed in 22 ms

6. BONUS_ABILITATI:

Cod Sql:

```
CREATE TABLE bonus_abilitati (
    scor INT PRIMARY KEY,
    bonus INT,
    CONSTRAINT ba_interval_scor CHECK ( scor>=1 AND scor<=30 ),
    CONSTRAINT ba_interval_bonus CHECK ( bonus>=-5 AND bonus<=10 )
);
```

PrintScreen:

The screenshot shows the Oracle SQL Developer interface. In the top-left corner, there's a 'Services' tree view with a green checkmark icon next to 'Database'. Below it, under 'Database', is a connection node for '127.0.0.1' with two child nodes: 'console_1' (66 ms) and 'console_1' (34 ms). The main area is a code editor with the following content:

```
107
108
109 ✓ CREATE TABLE bonus_abilitati (
110     scor INT PRIMARY KEY,
111     bonus INT,
112     CONSTRAINT ba_interval_scor CHECK ( scor>=1 AND scor<=30 ),
113     CONSTRAINT ba_interval_bonus CHECK ( bonus>=-5 AND bonus<=10 )
114 );
115
```

Below the code editor, the SQL command is executed:

```
[2025-05-18 20:40:46] Position: 1026
SYSTEM> CREATE TABLE bonus_abilitati (
    scor INT PRIMARY KEY,
    bonus INT,
    CONSTRAINT ba_interval_scor CHECK ( scor>=1 AND scor<=30 ),
    CONSTRAINT ba_interval_bonus CHECK ( bonus>=-5 AND bonus<=10 )
)
```

At the bottom of the interface, a status bar indicates: [2025-05-18 20:41:00] completed in 14 ms.

7. ARBORE DE ABILITATI:

Cod Sql:

```
CREATE TABLE arbore_de_abilitati (
    abilitati_id INT GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,
    scor_putere INT NOT NULL,
    scor_charisma INT NOT NULL,
    scor_constitutie INT NOT NULL,
    scor_inteligenta INT NOT NULL,
    scor_intelepciune INT NOT NULL,
    scor_dexteritate INT NOT NULL,

    CONSTRAINT aa_scor_str CHECK ( scor_putere>=1 AND
scor_putere<=30 ),
    CONSTRAINT aa_scor_char CHECK ( scor_charisma>=1 AND
scor_charisma<=30 ),
    CONSTRAINT aa_scor_const CHECK ( scor_constitutie>=1 AND
scor_constitutie<=30 ),
    CONSTRAINT aa_scor_int CHECK ( scor_inteligenta>=1 AND
scor_inteligenta<=30 ),
    CONSTRAINT aa_scor_wis CHECK ( scor_intelepciune>=1 AND
scor_intelepciune<=30 ),
    CONSTRAINT aa_scor_dex CHECK ( scor_dexteritate>=1 AND
scor_dexteritate<=30 ),

    FOREIGN KEY (scor_putere)
        REFERENCES bonus_abilitati(scor),
    FOREIGN KEY (scor_charisma)
        REFERENCES bonus_abilitati(scor),
    FOREIGN KEY (scor_constitutie)
        REFERENCES bonus_abilitati(scor),
    FOREIGN KEY (scor_inteligenta)
        REFERENCES bonus_abilitati(scor),
    FOREIGN KEY (scor_intelepciune)
        REFERENCES bonus_abilitati(scor),
    FOREIGN KEY (scor_dexteritate)
        REFERENCES bonus_abilitati(scor)
);
```

PrintScreen:

The screenshot shows the Database Explorer and a SQL query window in SSMS.

Database Explorer:

- Connected to **@127.0.0.1** (2 of 37)
- PUBLIC
 - synonyms: 7098
- SYSTEM
 - tables: 138
 - views: 10
 - routines: 6
 - sequences: 9
 - object types: 6
 - collection types: 6
 - synonyms: 8
- Server Objects

Services:

- Database
 - @127.0.0.1
 - console_1: 66 ms
 - console_1: 34 ms
 - console: 86 ms
 - console: 86 ms

SQL Query Window:

```

CREATE TABLE arbore_de_abilitati (
    scor_inteligenta INT NOT NULL,
    scor_intelepciuene INT NOT NULL,
    scor_dexteritate INT NOT NULL,
    ...
    CONSTRAINT aa_scor_str CHECK ( scor_putere>=1 AND scor_putere<=30 ),
    CONSTRAINT aa_scor_char CHECK ( scor_charisma>=1 AND scor_charisma<=30 ),
    CONSTRAINT aa_scor_const CHECK ( scor_constitutie>=1 AND scor_constitutie<=30 ),
    CONSTRAINT aa_scor_int CHECK ( scor_inteligenta>=1 AND scor_inteligenta<=30 ),
    CONSTRAINT aa_scor_wis CHECK ( scor_intelepciuene>=1 AND scor_intelepciuene<=30 ),
    CONSTRAINT aa_scor_dex CHECK ( scor_dexteritate>=1 AND scor_dexteritate<=30 ),
    ...
    -- iau cu referinta/ idk foreign key pt a putea sa le dau join in interogari cand calcu
    FOREIGN KEY (scor_putere)
        REFERENCES bonus_abilitati(scor),
    FOREIGN KEY (scor_charisma)
        REFERENCES bonus_abilitati(scor),
    FOREIGN KEY (scor_constitutie)
        REFERENCES bonus_abilitati(scor),
    FOREIGN KEY (scor_inteligenta)
        REFERENCES bonus_abilitati(scor),
    FOREIGN KEY (scor_intelepciuene)
        REFERENCES bonus_abilitati(scor),
    FOREIGN KEY (scor_dexteritate)
        REFERENCES bonus_abilitati(scor)
)
[2025-05-18 20:42:40] completed in 29 ms
  
```

8. PERSONAJ:

Cod Sql:

```
CREATE TABLE personaj (
    personaj_id INT GENERATED BY DEFAULT AS IDENTITY PRIMARY
KEY,
    utilizator_id INT,
    abilitati_id INT UNIQUE,
    neam_id INT,
    nume VARCHAR(255) NOT NULL,
    tip_personalitate VARCHAR(255),
    FOREIGN KEY (utilizator_id)
        REFERENCES utilizator(utilizator_id),
    FOREIGN KEY (abilitati_id)
        REFERENCES arbore_de_abilitati(abilitati_id),
    FOREIGN KEY (neam_id)
        REFERENCES neam(neam_id),
    CONSTRAINT p_tip_personalitate CHECK (
        tip_personalitate IS NULL OR
        tip_personalitate='legiuit-bun' OR
        tip_personalitate='legiuit-neutru' OR
        tip_personalitate='legiuit-malefic' OR
        tip_personalitate='neutru-bun' OR
        tip_personalitate='complet-neutru' OR
        tip_personalitate='neutru-malefic' OR
        tip_personalitate='haotic-bun' OR
        tip_personalitate='haotic-neutru' OR
        tip_personalitate='haotic-malefic'
    )
);
```

PrintScreen:

The screenshot shows the Oracle SQL Developer interface. On the left, the database schema browser displays the PUBLIC and SYSTEM schemas. The SYSTEM schema contains a folder named 'tables' which lists various system tables such as AQ\$_INTERNET_AGENT_PRIV, AQ\$_INTERNET_AGENTS, AQ\$_KEY_SHARD_MAP, AQ\$_QUEUE_TABLES, AQ\$_QUEUES, AQ\$_SCHEMES, ARBORE_DE_ABILITATI, BONUS_ABILITATI, CAMPANIE, CLASA, HELP, LOGMNR_ATTRIBUTE\$, LOGMNR_CCOLS, LOGMNR_CDEF\$, LOGMNR_COL\$, LOGMNR_COLTYPES\$, LOGMNR_CON\$, LOGMNR_CONTAINER\$, and LOGMNR_DICTIONARY\$. A context menu is open over the 'tables' folder.

The main pane shows the SQL code for creating the 'personaj' table. The code is as follows:

```
CREATE TABLE personaj (
    personaj_id INT GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,
    utilizator_id INT,
    abilitati_id INT,
    neam_id INT,
    nume VARCHAR(255) NOT NULL,
    tip_personalitate VARCHAR(255),
    FOREIGN KEY (utilizator_id)
        REFERENCES utilizator(utilizator_id),
    FOREIGN KEY (abilitati_id)
        REFERENCES arbore_de_abilitati(abilitati_id),
    FOREIGN KEY (neam_id)
        REFERENCES neam(neam_id),
    CONSTRAINT p_tip_personalitate CHECK (
        tip_personalitate IS NULL OR
        tip_personalitate='legivit-bun' OR
        tip_personalitate='legivit-neutrul' OR
        tip_personalitate='legivit-malefic' OR
        tip_personalitate='neutrul-bun' OR
        tip_personalitate='complet-neutrul' OR
        tip_personalitate='neutrul-malefic' OR
        tip_personalitate='haotic-bun' OR
        tip_personalitate='haotic-neutrul' OR
        tip_personalitate='haotic-malefic'
    )
);
```

Below the code, the services panel shows a connection to the database at 127.0.0.1 and a log entry for a query named 'console_1' completed in 66 ms.

9. PERSONAJ_CLASA:

Cod Sql:

```
CREATE TABLE personaj_clasa (
    personaj_id INT,
    clasa_id INT,
    nivel INT NOT NULL,
    PRIMARY KEY (personaj_id, clasa_id),
    FOREIGN KEY (personaj_id)
        REFERENCES personaj(personaj_id),
    FOREIGN KEY (clasa_id)
        REFERENCES clasa(clasa_id),
    CONSTRAINT pc_verifica_nivel_clasa CHECK ( nivel>=1 AND
    nivel <=20 )
);
```

PrintScreen:

The screenshot shows the Oracle SQL Developer interface. On the left, there's a tree view of database objects under 'Database' for the connection '@127.0.0.1'. The 'personaj_clasa' table is highlighted. The main pane displays the SQL code for creating the table, which includes primary key constraints, foreign key references to 'personaj' and 'clasa' tables, and a check constraint 'pc_verifica_nivel_clasa' ensuring 'nivel' is between 1 and 20. The code is numbered from 141 to 155. Below the code, the execution log shows the command 'CREATE TABLE personaj_clasa (' followed by the table definition, and the message '[2025-05-18 20:45:56] completed in 21 ms'. The bottom status bar also shows the completion time.

```
> ┌─ AQ$_INTERNET_AGENTS
> ┌─ AQ$_KEY_SHARD_MAP
> ┌─ AQ$_QUEUE_TABLES
> ┌─ AQ$_QUEUES
> ┌─ AQ$_SCHEDULES
> ┌─ ARBORE_DE_ABILITATI
> ┌─ BONUS_ABILITATI
> ┌─ CAMPANIE
> ┌─ CLASA
> ┌─ HELP
> ┌─ LOGMNR_AGE_SPILL$
> ┌─ LOGMNR_ATTRCOL$
> ┌─ LOGMNR_ATTRIBUTE$
```

```
141   ),
142   ),
143 ✓ CREATE TABLE personaj_clasa (
144     personaj_id INT,
145     clasa_id INT,
146     nivel INT NOT NULL,
147     PRIMARY KEY (personaj_id, clasa_id),
148     FOREIGN KEY (personaj_id)
149       REFERENCES personaj(personaj_id),
150     FOREIGN KEY (clasa_id)
151       REFERENCES clasa(clasa_id),
152     CONSTRAINT pc_verifica_nivel_clasa CHECK ( nivel>=1 AND nivel <=20 )
153   );
154
155
```

```
[2025-05-18 20:45:56] completed in 21 ms
SYSTEM> CREATE TABLE personaj_clasa (
    personaj_id INT,
    clasa_id INT,
    nivel INT NOT NULL,
    PRIMARY KEY (personaj_id, clasa_id),
    FOREIGN KEY (personaj_id)
        REFERENCES personaj(personaj_id),
    FOREIGN KEY (clasa_id)
        REFERENCES clasa(clasa_id),
    CONSTRAINT pc_verifica_nivel_clasa CHECK ( nivel>=1 AND nivel <=20 )
)
[2025-05-18 20:49:28] completed in 28 ms
```

10. PERSONAJ_CAMPANIE:

Cod Sql:

```
CREATE TABLE personaj_campanie (
    personaj_id INT,
    campanie_id INT,
    PRIMARY KEY (personaj_id, campanie_id),
    FOREIGN KEY (personaj_id)
        REFERENCES personaj(personaj_id),
    FOREIGN KEY (campanie_id)
        REFERENCES campanie(campanie_id)
);
```

PrintScreen:

The screenshot shows the Oracle SQL Developer interface. On the left, there is a tree view of database objects under 'Database' for a connection named '@127.0.0.1'. Under 'Database', there are nodes for 'Database', 'Schema', 'Tables', 'Views', 'Procedures', 'Functions', 'Triggers', and 'Synonyms'. Below these are 'console_1' and 'console'. The 'Tables' node is expanded, showing 'BONUS_ABILITATI', 'CAMPANIE', 'CLASA', 'HELP', 'LOGMNR_ATTRCOL\$', 'LOGMNR_ATTRIBUTE\$', 'LOGMNR_CCOL\$', 'LOGMNR_CDEF\$', 'LOGMNR_COL\$', and 'LOGMNR_COLTYPE\$'. On the right, the main pane displays the SQL code for creating the 'personaj_campanie' table. The code is numbered from 154 to 164. Lines 156 through 164 are highlighted with a green background, indicating the current selection or execution range. The code itself is as follows:

```
154
155
156 CREATE TABLE personaj_campanie (
157     personaj_id INT,
158     campanie_id INT,
159     PRIMARY KEY (personaj_id, campanie_id),
160     FOREIGN KEY (personaj_id)
161         REFERENCES personaj(personaj_id),
162     FOREIGN KEY (campanie_id)
163         REFERENCES campanie(campanie_id)
164 );
```

Below the code, the status bar indicates '[2025-05-18 20:47:20] completed in 20 ms'. The bottom pane shows the command history with two entries:

```
SYSTEM> CREATE TABLE personaj_campanie (
    personaj_id INT,
    campanie_id INT,
    PRIMARY KEY (personaj_id, campanie_id),
    FOREIGN KEY (personaj_id)
        REFERENCES personaj(personaj_id),
    FOREIGN KEY (campanie_id)
        REFERENCES campanie(campanie_id)
)
[2025-05-18 20:50:25] completed in 18 ms
```

11. PARTICIPARE:

Cod Sql:

```
CREATE TABLE participare (
    personaj_id INT,
    sedinta_id INT,
    prezenta VARCHAR(2) NOT NULL,
    PRIMARY KEY (personaj_id, sedinta_id),
    FOREIGN KEY (personaj_id)
        REFERENCES personaj(personaj_id),
    FOREIGN KEY (sedinta_id)
        REFERENCES sedinta(sedinta_id),
    CONSTRAINT participare_verif_prezenta CHECK (
        prezenta='da' OR prezenta='nu' )
);
```

PrintScreen:

The screenshot shows the Oracle SQL Developer interface. On the left, there's a tree view of database objects under 'Services'. Under 'Database', there's a connection to 'localhost:127.0.0.1' with three sessions listed: 'console_1' (66 ms), 'console_2' (34 ms), and 'console' (61 ms). The main area displays the SQL code for creating the 'participare' table. The code is numbered from 165 to 177. Lines 166 and 176 have green checkmarks next to them, indicating successful execution. The code defines the table structure with columns for personaj_id and sedinta_id, a primary key constraint, foreign key constraints, and a CHECK constraint for the prezenta column.

```
> ┌─ AQ$_QUEUES          165
> ┌─ AQ$_SCHEDULES       166 ✓ CREATE TABLE participare (
> ┌─ ARBORE_DE_ABILITATI 167   personaj_id INT,
> ┌─ BONUS_ABILITATI     168   sedinta_id INT,
> ┌─ CAMPANIE            169   prezenta VARCHAR(2) NOT NULL,
> ┌─ CLASA               170   PRIMARY KEY (personaj_id, sedinta_id),
> ┌─ HELP                171   FOREIGN KEY (personaj_id)
> ┌─ LOGMNR_AGE_SPILL$   172       REFERENCES personaj(personaj_id),
> ┌─ LOGMNR_ATTRCOL$     173   FOREIGN KEY (sedinta_id)
> ┌─ LOGMNR_ATTRIBUTE$   174       REFERENCES sedinta(sedinta_id),
> ┌─ LOGMNR_CCOL$        175   CONSTRAINT participare_verif_prezenta CHECK ( prezenta='da' OR prezenta='nu' )
> ┌─ LOGMNR_CDEF$        176   );
> ┌─ LOGMNR_CDEF$        177

services
+ - ⊞ □ + | ◇ ×
  ▾ Database
    ▾ @127.0.0.1
      ▾ console_1 66 ms
      ▾ console_2 34 ms
      ▾ console 61 ms
      ▾ console 61 ms
```

```
personaj_id INT,
sedinta_id INT,
prezenta VARCHAR(2) NOT NULL,
PRIMARY KEY (personaj_id, sedinta_id),
FOREIGN KEY (personaj_id)
    REFERENCES personaj(personaj_id),
FOREIGN KEY (sedinta_id)
    REFERENCES sedinta(sedinta_id),
CONSTRAINT participare_verif_prezenta CHECK ( prezenta='da' OR prezenta='nu' )

)
[2025-05-18 20:54:59] completed in 17 ms
```

12. ARHETIP:

Cod Sql:

```
CREATE TABLE arhetip (
    arhetip_id INT PRIMARY KEY,
    clasa_id INT,
    nume VARCHAR(255) NOT NULL,
    descriere CLOB,
    FOREIGN KEY (clasa_id)
        REFERENCES clasa(clasa_id)
);
```

PrintScreen:

The screenshot shows a database interface with a sidebar on the left containing various schema names like ARBORE_DE_ABILITATI, ARHETIP, BONUS_ABILITATI, etc. The main area displays a SQL code editor with the following content:

```
CREATE TABLE arhetip (
    arhetip_id INT PRIMARY KEY,
    clasa_id INT,
    nume VARCHAR(255) NOT NULL,
    descriere CLOB,
    FOREIGN KEY (clasa_id)
        REFERENCES clasa(clasa_id)
);
```

Below the code editor, the command was executed in the console:

```
SYSTEM> CREATE TABLE arhetip (
    arhetip_id INT PRIMARY KEY,
    clasa_id INT,
    nume VARCHAR(255) NOT NULL,
    descriere CLOB,
    FOREIGN KEY (clasa_id)
        REFERENCES clasa(clasa_id)
)
```

Execution details are shown at the bottom:

- [2025-05-18 20:54:59] completed in 17 ms
- [2025-05-18 20:55:50] completed in 12 ms

13. SUBNEAM:

Cod Sql:

```
CREATE TABLE subneam (
    subneam_id INT PRIMARY KEY,
    neam_id INT,
    nume VARCHAR(255) NOT NULL,
    descriere CLOB,
    FOREIGN KEY (neam_id)
        REFERENCES neam(neam_id)
);
```

PrintScreen:

The screenshot shows the Oracle SQL Developer interface. On the left, there's a tree view of database objects under 'Services'. Under 'Database', there's a connection to '127.0.0.1' with sessions named 'console_1' and 'console'. The main area shows the SQL editor with the following code:

```
CREATE TABLE subneam (
    subneam_id INT PRIMARY KEY,
    neam_id INT,
    nume VARCHAR(255) NOT NULL,
    descriere CLOB,
    FOREIGN KEY (neam_id)
        REFERENCES neam(neam_id)
);
```

The line 'CREATE TABLE subneam (' is highlighted with a green checkmark. The status bar at the bottom indicates the command was completed in 12 ms at [2025-05-18 20:55:50]. Below it, the full SQL statement is shown again with the message '[2025-05-18 20:56:35] completed in 20 ms'.

14. ATRIBUT:

Cod Sql:

```
CREATE TABLE atribut (
    atribut_id INT GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,
    nume VARCHAR(255) NOT NULL,
    descriere CLOB,
    bonus CLOB
);
```

PrintScreen:

The screenshot shows a database management interface with two main panes. The left pane, titled 'Services', lists several databases and their connection status. The right pane shows the SQL command for creating the 'atribut' table, with line numbers 183 through 189 visible. The command is highlighted with a green border. Below the command, the system outputs the execution details: '[2025-05-18 20:56:35] completed in 20 ms'. The command is repeated again with the same execution details: '[2025-05-18 20:57:58] completed in 29 ms'. The table structure defined in the SQL is:

```
CREATE TABLE atribut (
    atribut_id INT GENERATED BY DEFAULT AS IDENTITY PRIMARY KEY,
    nume VARCHAR(255) NOT NULL,
    descriere CLOB,
    bonus CLOB
)
```

15. ECHIPAMENT:

Cod Sql:

```
CREATE TABLE echipament (
    atribut_id INT PRIMARY KEY,
    greutate INT,
    putere VARCHAR(6),
    tip_echipament VARCHAR(6) NOT NULL,
    tip_lovitura VARCHAR(10),
    distanta INT,
    FOREIGN KEY (atribut_id)
        REFERENCES atribut(atribut_id),
    CONSTRAINT e_tip_echipament CHECK (
        tip_echipament='simplu' OR
        tip_echipament='militar' OR
        tip_echipament='scut' OR
        tip_echipament='armura'
    )
);
```

PrintScreen:

The screenshot shows the MySQL Workbench interface with the following details:

- Left Panel (Object Navigator):** Shows a tree view of database objects. The 'ECHIPAMENT' table is listed under the 'Tables' section.
- Right Panel (SQL Editor):** Displays the SQL code for creating the 'echipament' table. The code is identical to the one shown in the previous text block.
- Bottom Panel (Logs):** Shows the execution log for the query. It includes the query text, execution time (completed in 13 ms), and the timestamp [2025-05-18 20:58:36].

16. VRAJA:

Cod Sql:

```
CREATE TABLE vraja (
    atribut_id INT PRIMARY KEY,
    putere VARCHAR(10),
    tip_lovitura VARCHAR(50),
    durata VARCHAR(50),
    timp_necesar VARCHAR(50),
    scoala VARCHAR(50),
    FOREIGN KEY (atribut_id)
        REFERENCES atribut(atribut_id),
    CONSTRAINT v_scoli_posibile CHECK (
        scoala='evocare' OR
        scoala='divinatie' OR
        scoala='invocare' OR
        scoala='transmutatie' OR
        scoala='abjurare' OR -- renegare practic
        scoala='farmec' OR
        scoala='necromantie' OR
        scoala='iluzie'
    )
);
```

PrintScreen:

The screenshot shows the Oracle SQL Developer interface. On the left, there's a tree view of database objects under 'Services'. Under 'Database', there's a connection to '127.0.0.1' with two sessions: 'console_1' and 'console'. The main area displays the SQL code for creating the 'vraja' table. The code is numbered from 209 to 230. Lines 210 through 230 show the table definition and its constraints. The constraint part of the code is highlighted in green. The entire SQL statement ends with a semicolon on line 230.

```
@127.0.0.1
> ┌─ AQ$_SCHEDULES
> ┌─ ARBORE_DE_ABILITATI
> ┌─ ARHETIP
> ┌─ ATRIBUT
> ┌─ BONUS_ABILITATI
> ┌─ CAMPANIE
> ┌─ CLASA
> ┌─ ECHIPAMENT
> ┌─ HELP
> ┌─ LOGMNR_AGE_SPILL$
> ┌─ LOGMNR_ATTRCOL$
> ┌─ LOGMNR_ATTRIBUTE$
> ┌─ LOGMNR_CCOL$
> ┌─ LOGMNR_CDEF$
> ┌─ LOGMNR_COL$
> ┌─ LOGMNR_COLTYPE$
> ┌─ LOGMNR_CON$
> ┌─ LOGMNR_CONTAINER$
> ┌─ LOGMNR_DICTIONARY$ 209
  210 ✓ CREATE TABLE vraja (
  211     atribut_id INT PRIMARY KEY,
  212     putere VARCHAR(10),
  213     tip_lovitura VARCHAR(50),
  214     durata VARCHAR(50),
  215     timp_necesar VARCHAR(50),
  216     scoala VARCHAR(50),
  217     FOREIGN KEY (atribut_id)
  218         REFERENCES atribut(atribut_id),
  219     CONSTRAINT v_scoli_posibile CHECK (
  220         scoala='evocare' OR
  221         scoala='divinatie' OR
  222         scoala='invocare' OR
  223         scoala='transmutatie' OR
  224         scoala='abjurare' OR -- renegare practic
  225         scoala='farmec' OR
  226         scoala='necromantie' OR
  227         scoala='iluzie'
  228     )
  229 );
  230

Services
+ ⊕ □ ^ ×
Database
  - 127.0.0.1
    - console_1 66 ms
      - console 34 ms
    - console 52 ms
      - console 52 ms
```

17. ITEM_NEAM:

Cod Sql:

```
CREATE TABLE item_neam (
    neam_id INT,
    atribut_id INT,
    PRIMARY KEY (neam_id, atribut_id),
    FOREIGN KEY (neam_id)
        REFERENCES neam(neam_id),
    FOREIGN KEY (atribut_id)
        REFERENCES atribut(atribut_id)
);
```

PrintScreen:

The screenshot shows the Oracle SQL Developer interface. On the left, there is a tree view under 'Services' with nodes like 'Database' and 'Console'. The 'Console' node is expanded, showing a history of SQL statements. The statement for creating the 'item_neam' table is highlighted with a green border. The code is as follows:

```
CREATE TABLE item_neam (
    neam_id INT,
    atribut_id INT,
    PRIMARY KEY (neam_id, atribut_id),
    FOREIGN KEY (neam_id)
        REFERENCES neam(neam_id),
    FOREIGN KEY (atribut_id)
        REFERENCES atribut(atribut_id)
);
```

Below the code, the SQL command is executed in the console, and the output shows it was completed in 16 ms. The timestamp for the execution is [2025-05-18 21:03:27].

```
[2025-05-18 21:03:27] completed in 16 ms
SYSTEM> CREATE TABLE item_neam (
    neam_id INT,
    atribut_id INT,
    PRIMARY KEY (neam_id, atribut_id),
    FOREIGN KEY (neam_id)
        REFERENCES neam(neam_id),
    FOREIGN KEY (atribut_id)
        REFERENCES atribut(atribut_id)
)
[2025-05-18 21:04:44] completed in 16 ms
```

18. ITEM_SUBNEAM:

Cod Sql:

```
CREATE TABLE item_subneam (
    subneam_id INT,
    atribut_id INT,
    PRIMARY KEY (subneam_id, atribut_id),
    FOREIGN KEY (subneam_id)
        REFERENCES subneam(subneam_id),
    FOREIGN KEY (atribut_id)
        REFERENCES atribut(atribut_id)
);
```

PrintScreen:

The screenshot shows a database interface with a sidebar on the left containing a tree view of schemas and tables. The main area displays the SQL code for creating the 'item_subneam' table. The code is highlighted with syntax coloring. The right side of the interface shows the execution results, including the timestamp of completion and the generated SQL code.

```
CREATE TABLE item_subneam (
    subneam_id INT,
    atribut_id INT,
    PRIMARY KEY (subneam_id, atribut_id),
    FOREIGN KEY (subneam_id)
        REFERENCES subneam(subneam_id),
    FOREIGN KEY (atribut_id)
        REFERENCES atribut(atribut_id)
);
```

[2025-05-18 21:04:44] completed in 16 ms
SYSTEM> CREATE TABLE item_subneam (
 subneam_id INT,
 atribut_id INT,
 PRIMARY KEY (subneam_id, atribut_id),
 FOREIGN KEY (subneam_id)
 REFERENCES subneam(subneam_id),
 FOREIGN KEY (atribut_id)
 REFERENCES atribut(atribut_id)
)
[2025-05-18 21:06:20] completed in 13 ms

19. ITEM_CLASA:

Cod Sql:

```
CREATE TABLE item_clasa (
    clasa_id INT,
    atribut_id INT,
    PRIMARY KEY (clasa_id, atribut_id),
    FOREIGN KEY (clasa_id)
        REFERENCES clasa(clasa_id),
    FOREIGN KEY (atribut_id)
        REFERENCES atribut(atribut_id)
);
```

PrintScreen:

The screenshot shows the Oracle SQL Developer interface. On the left, there's a tree view of schema objects: CAMPANIE, CLASA, ECHIPAMENT, HELP, ITEM_CLASA (which is selected), ITEM_NEAM, ITEM_SUBNEAM, LOGMNR_AGE_SPILL\$, LOGMNR_ATTRCOL\$, and LOGMNR_ATTRRITF\$. Below the tree is a 'Services' section with a connection to 'Database @127.0.0.1' and a 'console_1' session.

In the main area, a code editor window displays the SQL code for creating the 'item_clasa' table. The code is highlighted with syntax coloring. A green checkmark is visible next to line 251, indicating the operation was successful.

```
CREATE TABLE item_clasa (
    clasa_id INT,
    atribut_id INT,
    PRIMARY KEY (clasa_id, atribut_id),
    FOREIGN KEY (clasa_id)
        REFERENCES clasa(clasa_id),
    FOREIGN KEY (atribut_id)
        REFERENCES atribut(atribut_id)
```

Below the code editor, the SQL command is run in the system console, and the output shows the table creation completed successfully in 12 ms.

```
SYSTEM> CREATE TABLE item_clasa (
    clasa_id INT,
    atribut_id INT,
    PRIMARY KEY (clasa_id, atribut_id),
    FOREIGN KEY (clasa_id)
        REFERENCES clasa(clasa_id),
    FOREIGN KEY (atribut_id)
        REFERENCES atribut(atribut_id)
)
[2025-05-18 21:07:04] completed in 12 ms
```

20. ITEM_ARHETIP:

Cod Sql:

```
CREATE TABLE item_arhetip (
    arhetip_id INT,
    atribut_id INT,
    PRIMARY KEY (arhetip_id, atribut_id),
    FOREIGN KEY (arhetip_id)
        REFERENCES arhetip(arhetip_id),
    FOREIGN KEY (atribut_id)
        REFERENCES atribut(atribut_id)
);
```

PrintScreen:

The screenshot shows the Oracle SQL Developer interface. On the left, there's a tree view of database objects under 'SCOTT'. In the center, a code editor window displays the SQL command for creating the 'item_arhetip' table. On the right, the results of the query execution are shown, including the SQL command itself and the execution details.

```
CREATE TABLE item_arhetip (
    arhetip_id INT,
    atribut_id INT,
    PRIMARY KEY (arhetip_id, atribut_id),
    FOREIGN KEY (arhetip_id)
        REFERENCES arhetip(arhetip_id),
    FOREIGN KEY (atribut_id)
        REFERENCES atribut(atribut_id)
);
```

Execution details from the bottom right:

```
SYSTEM> CREATE TABLE item_arhetip (
    arhetip_id INT,
    atribut_id INT,
    PRIMARY KEY (arhetip_id, atribut_id),
    FOREIGN KEY (arhetip_id)
        REFERENCES arhetip(arhetip_id),
    FOREIGN KEY (atribut_id)
        REFERENCES atribut(atribut_id)
)
[2025-05-18 21:07:56] completed in 13 ms
```

21. COMPONENTA:

Cod Sql:

```
CREATE TABLE componenta (
    componenta_id INT PRIMARY KEY,
    tip_componenta VARCHAR(255) NOT NULL,
    descriere CLOB
);
```

PrintScreen:

The screenshot shows the Oracle SQL Developer interface. On the left, there's a tree view of database objects: COMPONENTA, ECHIPAMENT, HELP, ITEM_ARHETIP, ITEM_CLASA, ITEM_NEAM, ITEM_SUBNEAM, and LOGMNR_AGE_SPII. The 'componenta' entry is selected. The main pane displays the SQL code for creating the table:

```
CREATE TABLE componenta (
    componenta_id INT PRIMARY KEY,
    tip_componenta VARCHAR(255) NOT NULL,
    descriere CLOB
);
```

Below the code, the history shows two entries:

- [2025-05-18 21:07:56] completed in 13 ms
SYSTEM> CREATE TABLE componenta (componenta_id INT PRIMARY KEY, tip_componenta VARCHAR(255) NOT NULL, descriere CLOB)
- [2025-05-18 21:10:32] completed in 20 ms
[Console] > @127.0.0.1 > [Console]

22. COMPONENTA_VRAJA:

Cod Sql:

```
CREATE TABLE componenta_vraja (
    componenta_id INT,
    atribut_id INT,
    PRIMARY KEY (componenta_id, atribut_id),
    FOREIGN KEY (componenta_id)
        REFERENCES componenta(componenta_id),
    FOREIGN KEY (atribut_id)
        REFERENCES atribut(atribut_id)
);
```

PrintScreen:

The screenshot shows a database interface with two main panes. The left pane, titled 'Services', displays a tree structure of database connections. Under 'Database', there is a connection to '127.0.0.1' with two sessions: 'console_1' (66 ms) and 'console' (68 ms). The right pane shows the SQL command for creating the 'componenta_vraja' table. The command is highlighted with a green border. The timestamp '[2025-05-18 21:11:28]' and the message 'completed in 16 ms' are also visible.

```
CREATE TABLE componenta_vraja (
    componenta_id INT,
    atribut_id INT,
    PRIMARY KEY (componenta_id, atribut_id),
    FOREIGN KEY (componenta_id)
        REFERENCES componenta(componenta_id),
    FOREIGN KEY (atribut_id)
        REFERENCES atribut(atribut_id)
);
```

INSERAREA ÎN TABELE:

1. CLASA:

Cod Sql:

```
INSERT INTO clasa (clasa_id, nume, descriere, dificultate)
VALUES (clase_si_arhetipuri.nextval, 'Paladin', 'Un razboinic
sfant', 'experimentat');

INSERT INTO clasa (clasa_id, nume, descriere, dificultate)
VALUES (clase_si_arhetipuri.nextval, 'Escroc', 'Un "anti-erou"
ce lucreaza din umbra...', 'incepator');

INSERT INTO clasa (clasa_id, nume, descriere, dificultate)
VALUES (clase_si_arhetipuri.nextval, 'Vrajitor', 'Un student ce
isi dedica viata catre intelegerarea magiei (magic nerd)',
'experimentat');

INSERT INTO clasa (clasa_id, nume, descriere, dificultate)
VALUES (clase_si_arhetipuri.nextval, 'Luptator', 'Un personaj
abil si invatat in aproape toate stilurile de lupta cot la
cot', 'incepator');

INSERT INTO clasa (clasa_id, nume, descriere, dificultate)
VALUES (clase_si_arhetipuri.nextval, 'Druid', 'Un iubitor al
naturii, binecuvantat de aceasta cu puteri supranaturale',
'expert');

INSERT INTO clasa (clasa_id, nume, descriere, dificultate)
VALUES (clase_si_arhetipuri.nextval, 'Cantaret', 'Ce spune
numele doar ca are magie de la muzica cantata',
'experimentat');
```

PrintScreen:

The screenshot shows a database interface with several panes:

- Playground**: A code editor showing a series of SQL INSERT statements for a "clasa" table. The statements are numbered 318 to 333. The descriptions for the classes include "Un razboinic sfant", "Un anti-erou", "Un student ce isi dedica viata catre intelegerea magiei (magic nerd)", "Un iubitor al naturii, binecuvantat de aceasta cu puteri supranaturale", and "Ce spune numele doar ca are magie de la muzica cantata".
- Services**: A tree view of database services, including PUBLIC, SYSTEM, and various tables like AQ\$INTERNET_AGENT_PRV, AQ\$INTERNET_AGENTS, AQ\$KEY_SHARD_MAP, AQ\$QUEUE_TABLES, AQ\$QUEUES, AQ\$SCHEDULES, ARBORE_DE_ABILITATI, ARHETIP, and ATRIBUT.
- Console**: A log of SQL queries and their execution times. The log shows the execution of the INSERT statements from the playground.
- Database**: A table viewer showing the "clasa" table with columns CLASA_ID, NUME, DESCRIERE, and DIFICULTATE. The data is as follows:

CLASA_ID	NUME	DESCRIERE	DIFICULTATE
1	Paladin	Un razboinic sfant	experimentat
2	Escroc	Un "anti-erou" ce lucreaza din umbra...	incepator
3	Vrajitor	Un student ce isi dedica viata catre intelegerea magiei (magic nerd)	experimentat
4	Luptator	Un personaj abil si invatat in aproape toate stilurile de lupta cot la cot	incepator
5	Druid	Un iubitor al naturii, binecuvantat de aceasta cu puteri supranaturale	expert
6	Cantaret	Ce spune numele doar ca are magie de la muzica cantata	experimentat

2. UTILIZATOR:

Cod Sql:

```
INSERT INTO utilizator (tip_utilizator, nume, prenume,
username, parola, data_inregistrare, email, numar_telefon)
VALUES ('organizator', 'Poscu', 'Teo',
'sweetKid','pisici??',TO_DATE('2025-05-15', 'YYYY-MM-DD'),
'teo.rosca@gmail.com', 0727236404);

INSERT INTO utilizator (tip_utilizator, nume, prenume,
username, parola, data_inregistrare, email, numar_telefon)
VALUES ('jucator', 'Sofr', 'Luca', 'yue',
'yayJocDnd',TO_DATE('2025-05-20', 'YYYY-MM-DD'),
'yue@yahoo.com', '0744456799');

INSERT INTO utilizator (tip_utilizator, nume, prenume,
username, parola, data_inregistrare, email, numar_telefon)
VALUES ('organizator', 'Blinton', 'Felix', 'paradoxal',
'D3D3M4N',TO_DATE('2025-05-04', 'YYYY-MM-DD'),
'paradolix@gmail.com', '0712345678');

INSERT INTO utilizator (tip_utilizator, nume, prenume,
username, parola, data_inregistrare, email, numar_telefon)
VALUES ('jucator', 'Eiosanu', 'Ray', 's3rs.exe', 'nuStiu?',
TO_DATE('2025-04-07', 'YYYY-MM-DD'),
'rayzor.sharp@gmail.com', '0799999999');

INSERT INTO utilizator (tip_utilizator, nume, prenume,
username, parola, data_inregistrare, email, numar_telefon)
VALUES ('jucator', 'Branza', 'Mura', 'miurel', 'amUitat',
TO_DATE('2025-03-26', 'YYYY-MM-DD'), 'mura@gmail.com',
'0727343434');
```

PrintScreen:

The screenshot shows the Oracle SQL Developer interface with three main panes: Object Navigator, SQL Worksheet, and Services.

Object Navigator: Shows the schema structure under @127.0.0.1 (2 of 37). It includes PUBLIC, SYSTEM, and several tables: AQ\$INTERNET_AGENT_PRIV, AQ\$INTERNET_AGENTS, AQ\$KEY_SHARD_MAP, AQ\$QUEUE_TABLES, AQ\$QUEUES, AQ\$SCHEDULES, ARBORE_DE_ABILITATI, ARHETIP, ATTRIBUT, BONUS_ABILITATI, and CAMPANIE.

SQL Worksheet: Displays the DDL for creating the UTILIZATOR table and inserting data into it. The code is as follows:

```

297 -- UTILIZATOR
298 ✓ INSERT INTO utilizator (tip_utilizator, nume, prenume, username, parola, data_inregistrare, email, numar_telefon)
299 VALUES ( tip_utilizator 'organizator', nume 'Poscu', prenume 'Teo', username 'sweetKid', parola 'pisici??', data_inregistrare TO_DATE('2025-05-15', 'YYYY-MM-DD')
300
301 ✓ INSERT INTO utilizator (tip_utilizator, nume, prenume, username, parola, data_inregistrare, email, numar_telefon)
302 VALUES ( tip_utilizator 'jucator', nume 'Sofr', prenume 'Luca', username 'yue', parola 'yayJocDnd', data_inregistrare TO_DATE('2025-05-20', 'YYYY-MM-DD')
303
304 ✓ INSERT INTO utilizator (tip_utilizator, nume, prenume, username, parola, data_inregistrare, email, numar_telefon)
305 VALUES ( tip_utilizator 'organizator', nume 'Blinton', prenume 'Felix', username 'paradoxal', parola 'D3D3M4N', data_inregistrare TO_DATE('2025-05-04', 'YYYY-MM-DD')
306
307 ✓ INSERT INTO utilizator (tip_utilizator, nume, prenume, username, parola, data_inregistrare, email, numar_telefon)
308 VALUES ( tip_utilizator 'jucator', nume 'Eiosanu', prenume 'Ray', username 's3rs.exe', parola 'nuStiu?', data_inregistrare TO_DATE('2025-04-07', 'YYYY-MM-DD')
309
310 ✓ INSERT INTO utilizator (tip_utilizator, nume, prenume, username, parola, data_inregistrare, email, numar_telefon)
311 VALUES ( tip_utilizator 'jucator', nume 'Branza', prenume 'Mura', username 'miurel', parola 'amUitat', data_inregistrare TO_DATE('2025-03-26', 'YYYY-MM-DD')
312
313 -- CLASA

```

Services: Shows the execution history in the Database section for the session @127.0.0.1. The log output is as follows:

```

[2025-05-19 13:58:43] completed in 16 ms
SYSTEM> INSERT INTO utilizator (tip_utilizator, nume, prenume, username, parola, data_inregistrare, email, numar_telefon)
VALUES ('organizator', 'Poscu', 'Teo', 'sweetKid', 'pisici??', TO_DATE('2025-05-15', 'YYYY-MM-DD'), 'teo.rosca@gmail.com', 0727236404)
[2025-05-19 13:58:43] 1 row affected in 34 ms
SYSTEM> INSERT INTO utilizator (tip_utilizator, nume, prenume, username, parola, data_inregistrare, email, numar_telefon)
VALUES ('jucator', 'Sofr', 'Luca', 'yue', 'yayJocDnd', TO_DATE('2025-05-20', 'YYYY-MM-DD'), 'yue@yahoo.com', 744456799)
[2025-05-19 13:58:43] 1 row affected in 52 ms
SYSTEM> INSERT INTO utilizator (tip_utilizator, nume, prenume, username, parola, data_inregistrare, email, numar_telefon)
VALUES ('organizator', 'Blinton', 'Felix', 'paradoxal', 'D3D3M4N', TO_DATE('2025-05-04', 'YYYY-MM-DD'), 'paradolix@gmail.com', 712345678)
[2025-05-19 13:58:43] 1 row affected in 7 ms
SYSTEM> INSERT INTO utilizator (tip_utilizator, nume, prenume, username, parola, data_inregistrare, email, numar_telefon)
VALUES ('jucator', 'Eiosanu', 'Ray', 's3rs.exe', 'nuStiu?', TO_DATE('2025-04-07', 'YYYY-MM-DD'), 'rayzor.sharp@gmail.com', 7999999999)
[2025-05-19 13:58:43] 1 row affected in 8 ms
SYSTEM> INSERT INTO utilizator (tip_utilizator, nume, prenume, username, parola, data_inregistrare, email, numar_telefon)
VALUES ('jucator', 'Branza', 'Mura', 'miurel', 'amUitat', TO_DATE('2025-03-26', 'YYYY-MM-DD'), 'mura@gmail.com', 0727343434)
[2025-05-19 13:58:43] 1 row affected in 6 ms

```

Object Navigator (Bottom): Shows the ARBORE_DE_ABILITATI table with a SELECT * query. The code is:

```

312 SELECT *
313 FROM UTILIZATOR;
314
315 -- CLASA
316
317 TRUNCATE TABLE UTILIZATOR;

```

Services (Bottom): Shows the UTILIZATOR table data in a grid view. The columns are UTILIZATOR_ID, TIP_UTILIZATOR, NUME, PRENUME, USERNAME, PAROLA, DATA_INREGISTRARE, EMAIL, and NUMAR_TELEFON. The data is as follows:

UTILIZATOR_ID	TIP_UTILIZATOR	NUME	PRENUME	USERNAME	PAROLA	DATA_INREGISTRARE	EMAIL	NUMAR_TELEFON
1	organizator	Poscu	Teo	sweetKid	pisici??	2025-05-15	teo.rosca@gmail.com	0727236404
2	jucator	Sofr	Luca	yue	yayJocDnd	2025-05-20	yue@yahoo.com	744456799
3	organizator	Blinton	Felix	paradoxal	D3D3M4N	2025-05-04	paradolix@gmail.com	712345678
4	jucator	Eiosanu	Ray	s3rs.exe	nuStiu?	2025-04-07	rayzor.sharp@gmail.com	7999999999
5	jucator	Branza	Mura	miurel	amUitat	2025-03-26	mura@gmail.com	0727343434

3. CAMPANIE:

Cod Sql:

```
INSERT INTO campanie (utilizator_id, nume, data_inceput)
VALUES (1, 'NethRodas gods', TO_DATE('2022-01-20',
'YYYY-MM-DD'));

INSERT INTO campanie (utilizator_id, nume, data_inceput)
VALUES (3, 'TralaleloTralala', TO_DATE('2025-03-28',
'YYYY-MM-DD'));

INSERT INTO campanie (utilizator_id, nume, data_inceput)
VALUES (3, 'Werewolves', TO_DATE('2024-09-24', 'YYYY-MM-DD'));

INSERT INTO campanie (utilizator_id, nume, data_inceput,
data_final)
VALUES (1, 'ballerinaCappucina', TO_DATE('2022-01-20',
'YYYY-MM-DD'), TO_DATE('2023-02-03', 'YYYY-MM-DD'));

INSERT INTO campanie (utilizator_id, nume, data_inceput,
data_final)
VALUES (1, 'Descend to Avernus', TO_DATE('2023-07-14',
'YYYY-MM-DD'), TO_DATE('2024-01-13', 'YYYY-MM-DD'));

SELECT * FROM campanie;
```

PrintScreen:

Services

```

> ROLLING$STATISTICS 339
> ROLLING$STATUS 340 ✓ INSERT INTO campanie (utilizator_id, nume, data_incep)
> SCHEDULER_JOB_ARGS_TBI 341 VALUES ( utilizator_id_1, nume 'NethRodas_gods', data_incep TO_DATE('2022-01-20', 'YYYY-MM-DD'));
> SCHEDULER_PROGRAM_ARC 342
> SEDINTA 343 ✓ INSERT INTO campanie (utilizator_id, nume, data_incep)
> SQLPLUS_PRODUCT_PROFILE 344 VALUES ( utilizator_id_3, nume 'TralaleloTralala', data_incep TO_DATE('2025-03-28', 'YYYY-MM-DD'));
> SUBNEAM 345
> UTILIZATOR 346 ✓ INSERT INTO campanie (utilizator_id, nume, data_incep)
>   columns 9 347 VALUES ( utilizator_id_3, nume 'Werewolves', data_incep TO_DATE('2024-09-24', 'YYYY-MM-DD'));
>   keys 2 348
>   indexes 2 349 ✓ INSERT INTO campanie (utilizator_id, nume, data_incep, data_final)
>   checks 2 350 VALUES ( utilizator_id_1, nume 'ballerinaCappucina', data_incep TO_DATE('2022-01-20', 'YYYY-MM-DD'), data_final TO_DATE('2023-02-03', 'YYYY-MM-DD'));
> VRAJA 351
> views 10 352 ✓ INSERT INTO campanie (utilizator_id, nume, data_incep, data_final)
> routines 6 353 VALUES ( utilizator_id_1, nume 'Descend to Avernus', data_incep TO_DATE('2023-07-14', 'YYYY-MM-DD'), data_final TO_DATE('2024-01-13', 'YYYY-MM-DD'));
354

```

Services

```

> Output SYSTEM.UTILIZATOR
[2025-05-19 14:35:52] Position: 70
Data
@ VALUES (1, 'NethRodas gods', TO_DATE('2022-01-20', 'YYYY-MM-DD'))
[2025-05-19 14:34:26] 1 row affected in 9 ms
SYSTEM> INSERT INTO campanie (utilizator_id, nume, data_incep)
VALUES (3, 'TralaleloTralala', TO_DATE('2025-03-28', 'YYYY-MM-DD'))
[2025-05-19 14:34:26] 1 row affected in 8 ms
SYSTEM> INSERT INTO campanie (utilizator_id, nume, data_incep)
VALUES (3, 'Werewolves', TO_DATE('2024-09-24', 'YYYY-MM-DD'))
[2025-05-19 14:34:26] 1 row affected in 7 ms
SYSTEM> INSERT INTO campanie (utilizator_id, nume, data_incep, data_final)
VALUES (1, 'ballerinaCappucina', TO_DATE('2022-01-20', 'YYYY-MM-DD'), TO_DATE('2023-02-03', 'YYYY-MM-DD'))
[2025-05-19 14:34:26] 1 row affected in 8 ms
SYSTEM> INSERT INTO campanie (utilizator_id, nume, data_incep, data_final)
VALUES (1, 'Descend to Avernus', TO_DATE('2023-07-14', 'YYYY-MM-DD'), TO_DATE('2024-01-13', 'YYYY-MM-DD'))
[2025-05-19 14:34:26] 1 row affected in 6 ms

```

Services

```

> Output SYSTEM.CAMPANIE
[2025-05-19 14:35:52] Position: 70
Data
@ CAMPAINE_ID UTILIZATOR_ID NUME DATA_INCEPUT DATA_FINAL
1 1 NethRodas gods 2022-01-20 <null>
2 2 TralaleloTralala 2025-03-28 <null>
3 3 Werewolves 2024-09-24 <null>
4 1 ballerinaCappuci... 2022-01-20 2023-02-03
5 1 Descend to Avern... 2023-07-14 2024-01-13

```

4. NEAM:

Cod Sql:

```
INSERT INTO neam (neam_id, nume, descriere, tip)
VALUES (neamuri_si_subneamuri.nextval, 'Om', 'Practic omul din
viata reala', 'uman');

INSERT INTO neam (neam_id, nume, descriere, tip)
VALUES (neamuri_si_subneamuri.nextval, 'Elf', 'Creaturi
similar cu oamenii, dar mereu tineri si mereu frumosi (Legolas
din stapanul inelelor ca idee)', 'uman');

INSERT INTO neam (neam_id, nume, descriere, tip)
VALUES (neamuri_si_subneamuri.nextval, 'Gnom', 'Pitici ce
traiesc in mine si au *aproape* mereu barba lunga (Gimpli din
stapanul inelelor)', 'uman');

INSERT INTO neam (neam_id, nume, descriere, tip)
VALUES (neamuri_si_subneamuri.nextval, 'Zână', 'Mici creaturi
posnase cu aripi transparente', 'zana');

INSERT INTO neam (neam_id, nume, descriere, tip)
VALUES (neamuri_si_subneamuri.nextval, 'Hobbit', 'Oameni care
zici ca au fost micsorati ca o imagine dintr-un word (atunci
cand tragi de un colt si se pastreaza dimensiunile, dar se
micsoreaza tot)', 'uman');

INSERT INTO neam (neam_id, nume, descriere, tip)
VALUES (neamuri_si_subneamuri.nextval, 'Vampir', 'Exact ce pare
din nume', 'strigoi');

INSERT INTO neam (neam_id, nume, descriere, tip)
VALUES (neamuri_si_subneamuri.nextval, 'Imp', 'Mici diavoli
rosii', 'diavol');
```

PrintScreen:

Services

```

@127.0.0.1 376 VALUES ( neam_id neamuri_si_subneamuri.nextval, nume 'Hobbit', descriere 'Oameni care zici ca au fost micsorati ca o imagine dintr-un word (atunci cand tragi %23
> [REDACTED] ROLLING$PLAN 371
> [REDACTED] ROLLING$STAT 372 ✓ INSERT INTO neam (neam_id, nume, descriere, tip)
> [REDACTED] ROLLING$STAT 373 VALUES ( neam_id neamuri_si_subneamuri.nextval, nume 'Vampir', descriere 'Exact ce pare din nume', tip 'strigoi');
> [REDACTED] SCHEDULER_JO 374
> [REDACTED] SCHEDULER_PR 375 ✓ INSERT INTO neam (neam_id, nume, descriere, tip)
> [REDACTED] SEDINTA 376 VALUES ( neam_id neamuri_si_subneamuri.nextval, nume 'Imp', descriere 'Mici diavoli rosii', tip 'diavol');
> [REDACTED] SQLPLUS_PROD 377
> [REDACTED] SUBNEAM 378
> [REDACTED] UTILIZATOR 379

```

Services

```

Output SYSTEM.CAMPANIE
Data
  SYSTEM> INSERT INTO neam (neam_id, nume, descriere, tip)
    VALUES (neamuri_si_subneamuri.nextval, 'Om', 'Practic omul din viata reala', 'uman')
  [2025-05-19 14:43:12] 1 row affected in 10 ms
  SYSTEM> INSERT INTO neam (neam_id, nume, descriere, tip)
    VALUES (neamuri_si_subneamuri.nextval, 'Elf', 'Creaturi similare cu oamenii, dar mereu tineri si mereu frumosi (Legolas din stapanul inelelor)', 'uman')
  [2025-05-19 14:43:12] 1 row affected in 8 ms
  SYSTEM> INSERT INTO neam (neam_id, nume, descriere, tip)
    VALUES (neamuri_si_subneamuri.nextval, 'Gnom', 'Pitici ce traiesc in mine si au *aproape* mereu barba lunga (Gimli din stapanul inelelor)', 'uman')
  [2025-05-19 14:43:12] 1 row affected in 11 ms
  SYSTEM> INSERT INTO neam (neam_id, nume, descriere, tip)
    VALUES (neamuri_si_subneamuri.nextval, 'Zâna', 'Mici creaturi posnase cu aripi transparente', 'zana')
  [2025-05-19 14:43:13] 1 row affected in 9 ms
  SYSTEM> INSERT INTO neam (neam_id, nume, descriere, tip)
    VALUES (neamuri_si_subneamuri.nextval, 'Hobbit', 'Oameni care zici ca au fost micsorati ca o imagine dintr-un word (atunci cand tragi de un colt si se pastreaza dimensiunile, dar s
  [2025-05-19 14:43:13] 1 row affected in 8 ms
  SYSTEM> INSERT INTO neam (neam_id, nume, descriere, tip)
    VALUES (neamuri_si_subneamuri.nextval, 'Vampir', 'Exact ce pare din nume', 'strigoi')
  [2025-05-19 14:43:13] 1 row affected in 8 ms
  SYSTEM> INSERT INTO neam (neam_id, nume, descriere, tip)
    VALUES (neamuri_si_subneamuri.nextval, 'Imp', 'Mici diavoli rosii', 'diavol')
  [2025-05-19 14:43:13] 1 row affected in 9 ms

```

```

Output SYSTEM.NEAM
Data
  NEAM_ID NUME DESCRIERE TIP
  1 Om Practic omul din viata reala uman
  2 Elf Creaturi similare cu oamenii, dar mereu tineri si mereu frumosi (Legolas... uman
  3 Gnom Pitici ce traiesc in mine si au *aproape* mereu barba lunga (Gimli din ... uman
  4 Zâna Mici creaturi posnase cu aripi transparente zana
  5 Hobbit Oameni care zici ca au fost micsorati ca o imagine dintr-un word (atunci... uman
  6 Vampir Exact ce pare din nume strigoi
  7 Imp Mici diavoli rosii diavol

```

5. BONUS_ABILITATI: (exact 30 de linii; regulile astea chiar sunt din D&D totuși)

Cod Sql:

```
INSERT INTO bonus_abilitati (scor, bonus)
VALUES (1, -4);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (2, -4);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (3, -4);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (4, -3);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (5, -3);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (6, -2);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (7, -2);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (8, -1);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (9, -1);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (10, 0);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (11, 0);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (12, 1);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (13, 1);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (14, 2);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (15, 2);

INSERT INTO bonus_abilitati (scor, bonus)
```

```
VALUES (16, 3);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (17, 3);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (18, 4);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (19, 4);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (20, 5);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (21, 5);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (22, 6);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (23, 6);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (24, 7);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (25, 7);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (26, 8);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (27, 8);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (28, 9);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (29, 9);

INSERT INTO bonus_abilitati (scor, bonus)
VALUES (30, 10);
```

PrintScreen:

```

SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (1, -4)
[2025-05-19 14:57:35] 1 row affected in 8 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (2, -4)
[2025-05-19 14:57:35] 1 row affected in 6 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (3, -4)
[2025-05-19 14:57:36] 1 row affected in 6 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (4, -3)
[2025-05-19 14:57:36] 1 row affected in 6 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (5, -3)
[2025-05-19 14:57:36] 1 row affected in 36 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (6, -2)
[2025-05-19 14:57:36] 1 row affected in 6 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (7, -2)
[2025-05-19 14:57:36] 1 row affected in 6 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (8, -1)
[2025-05-19 14:57:36] 1 row affected in 6 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (9, -1)
[2025-05-19 14:57:36] 1 row affected in 5 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (10, 0)
[2025-05-19 14:57:36] 1 row affected in 5 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (11, 0)
[2025-05-19 14:57:36] 1 row affected in 5 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (12, 0)
[2025-05-19 14:57:36] 1 row affected in 6 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (13, 1)
[2025-05-19 14:57:36] 1 row affected in 6 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (14, 1)
[2025-05-19 14:57:36] 1 row affected in 7 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (15, 2)
[2025-05-19 14:57:36] 1 row affected in 6 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (16, 2)
[2025-05-19 14:57:36] 1 row affected in 6 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (17, 3)
[2025-05-19 14:57:36] 1 row affected in 6 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (18, 4)
[2025-05-19 14:57:36] 1 row affected in 5 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (19, 4)
[2025-05-19 14:57:36] 1 row affected in 6 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (20, 5)
[2025-05-19 14:57:36] 1 row affected in 5 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (21, 5)
[2025-05-19 14:57:36] 1 row affected in 6 ms
SYSTEM> INSERT INTO bonus_abilitati (scor, bonus)
      VALUES (22, 6)
[2025-05-19 14:57:36] 1 row affected in 6 ms

```

[2025-05-19 14:57:36] 1 row affected in 6 ms

SCOR	BONUS	ABILITATI
1	1	-4
2	2	-4
3	3	-4
4	4	-3
5	5	-3
6	6	-2
7	7	-2
8	8	-1
9	9	-1
10	10	0
11	11	0
12	12	1
13	13	1
14	14	2
15	15	2
16	16	3
17	17	3
18	18	4
19	19	4
20	20	5
21	21	5
22	22	6
23	23	6

6. ARBORE DE ABILITATI:

Cod Sql:

```
INSERT INTO arbore_de_abilitati (scor_putere, scor_charisma,
scor_constitutie, scor_inteligenta, scor_intelepciune,
scor_dexteritate)
VALUES (18, 16, 14, 10, 12, 8);

INSERT INTO arbore_de_abilitati (scor_putere, scor_charisma,
scor_constitutie, scor_inteligenta, scor_intelepciune,
scor_dexteritate)
VALUES (8, 14, 15, 13, 14, 19);

INSERT INTO arbore_de_abilitati (scor_putere, scor_charisma,
scor_constitutie, scor_inteligenta, scor_intelepciune,
scor_dexteritate)
VALUES (11, 13, 14, 16, 19, 13);

INSERT INTO arbore_de_abilitati (scor_putere, scor_charisma,
scor_constitutie, scor_inteligenta, scor_intelepciune,
scor_dexteritate)
VALUES (19, 12, 16, 8, 14, 12);

INSERT INTO arbore_de_abilitati (scor_putere, scor_charisma,
scor_constitutie, scor_inteligenta, scor_intelepciune,
scor_dexteritate)
VALUES (8, 13, 14, 19, 16, 12);
```

PrintScreen:

@127.0.0.1 [2 of 37]

```

55 PUBLIC
>   □ synonyms 7098
56 SYSTEM
<   □ tables 153
>     □ AQ$_INTERNET_AGENT_PRIVS
>     □ AQ$_INTERNET_AGENTS
>     □ AQ$_KEY_SHARD_MAP
>     □ AQ$_QUEUE_TABLES
>     □ AQ$_QUEUES
>     □ AQ$_SCHEDULES
>     □ ARBORE_DE_ABILITATI
>     □ ARHETIP
>     □ ATRIBUT

```

476 ✓ INSERT INTO arbore_de_abilitati (scor_putere, scor_charisma, scor_constitutie, scor_inteligenta, scor_intelepciuene, scor_dexteritate) VALUES (scor_putere 18, scor_charisma 16, scor_constitutie 14, scor_inteligenta 10, scor_intelepciuene 12, scor_dexteritate 8);
477
478
479 ✓ INSERT INTO arbore_de_abilitati (scor_putere, scor_charisma, scor_constitutie, scor_inteligenta, scor_intelepciuene, scor_dexteritate) VALUES (scor_putere 8, scor_charisma 14, scor_constitutie 15, scor_inteligenta 13, scor_intelepciuene 14, scor_dexteritate 19);
480
481
482 ✓ INSERT INTO arbore_de_abilitati (scor_putere, scor_charisma, scor_constitutie, scor_inteligenta, scor_intelepciuene, scor_dexteritate) VALUES (scor_putere 11, scor_charisma 13, scor_constitutie 14, scor_inteligenta 16, scor_intelepciuene 19, scor_dexteritate 13);
483
484
485 ✓ INSERT INTO arbore_de_abilitati (scor_putere, scor_charisma, scor_constitutie, scor_inteligenta, scor_intelepciuene, scor_dexteritate) VALUES (scor_putere 19, scor_charisma 12, scor_constitutie 16, scor_inteligenta 8, scor_intelepciuene 14, scor_dexteritate 12);
486
487
488 ✓ INSERT INTO arbore_de_abilitati (scor_putere, scor_charisma, scor_constitutie, scor_inteligenta, scor_intelepciuene, scor_dexteritate) VALUES (scor_putere 8, scor_charisma 13, scor_constitutie 14, scor_inteligenta 19, scor_intelepciuene 16, scor_dexteritate 12);
489
490

es

+ ☀ | □ | ☈ | ×

Database @127.0.0.1

Output SYSTEM.CLASA

```

SYSTEM> SELECT * FROM CLASA
[2025-05-19 16:21:12] 6 rows retrieved starting from 1 in 170 ms (execution: 18 ms, fetching: 152 ms)
SYSTEM> INSERT INTO arbore_de_abilitati (scor_putere, scor_charisma, scor_constitutie, scor_inteligenta, scor_intelepciuene, scor_dexteritate)
VALUES (18, 16, 14, 10, 12, 8)
[2025-05-19 16:21:23] 1 row affected in 6 ms
SYSTEM> INSERT INTO arbore_de_abilitati (scor_putere, scor_charisma, scor_constitutie, scor_inteligenta, scor_intelepciuene, scor_dexteritate)
VALUES (8, 14, 15, 13, 14, 19)
[2025-05-19 16:21:23] 1 row affected in 7 ms
SYSTEM> INSERT INTO arbore_de_abilitati (scor_putere, scor_charisma, scor_constitutie, scor_inteligenta, scor_intelepciuene, scor_dexteritate)
VALUES (11, 13, 14, 16, 19, 13)
[2025-05-19 16:21:23] 1 row affected in 8 ms
SYSTEM> INSERT INTO arbore_de_abilitati (scor_putere, scor_charisma, scor_constitutie, scor_inteligenta, scor_intelepciuene, scor_dexteritate)
VALUES (19, 12, 16, 8, 14, 12)
[2025-05-19 16:21:23] 1 row affected in 9 ms
SYSTEM> INSERT INTO arbore_de_abilitati (scor_putere, scor_charisma, scor_constitutie, scor_inteligenta, scor_intelepciuene, scor_dexteritate)
VALUES (8, 13, 14, 19, 16, 12)
[2025-05-19 16:21:23] 1 row affected in 7 ms

```

AQ\$_QUEUES AQ\$_SCHEDULES ARBORE_DE_ABILITATI ARHETIP ATRIBUT

497 VALUES (scor_putere 8, scor_charisma 13, scor_constitutie 14, scor_inteligenta 19, scor_intelepciuene 16, scor_dexteritate 12);
498
499 ✓ SELECT * FROM ARBORE_DE_ABILITATI;

Services

+ ☀ | □ | Output SYSTEM.ARBORE_DE_ABILITATI ×

Database @127

ABILITATI_ID	SCOR_PUTERE	SCOR_CHARISMA	SCOR_CONSTITUTIE	SCOR_INTELLIGENTA	SCOR_INTELEPCIUNE	SCOR_DEXTERITATE
1	2	18	16	14	10	12
2	3	8	14	15	13	14
3	4	11	13	14	16	19
4	5	19	12	16	8	14
5	6	8	13	14	19	16

7. PERSONAJ:

Cod Sql:

```
INSERT INTO personaj (utilizator_id, abilitati_id, neam_id,
nume, tip_personalitate)
VALUES (1, 2, 2, 'Cyrus', 'legiuit-bun');

INSERT INTO personaj (utilizator_id, abilitati_id, neam_id,
nume, tip_personalitate)
VALUES (1, 3, 5, 'Bilbo', 'complet-neutru');

INSERT INTO personaj (utilizator_id, abilitati_id, neam_id,
nume, tip_personalitate)
VALUES (5, 4, 1, 'Reiko', 'haotic-neutru');

INSERT INTO personaj (utilizator_id, abilitati_id, neam_id,
nume, tip_personalitate)
VALUES (2, 5, 3, 'Esteria', 'haotic-neutru');

INSERT INTO personaj (utilizator_id, abilitati_id, neam_id,
nume, tip_personalitate)
VALUES (3, 6, 1, 'Nyx', 'haotic-malefic');
```

PrintScreen:

0.0.1 [2 of 37]

```

496 ✓ INSERT INTO personaj (utilizator_id, abilitati_id, neam_id, nume, tip_personalitate)
VALUES (utilizator_id 1, abilitati_id 2, neam_id 2, nume 'Cyrus', tip_personalitate 'legiuit-bun');

498 ✓
499 ✓ INSERT INTO personaj (utilizator_id, abilitati_id, neam_id, nume, tip_personalitate)
VALUES (utilizator_id 1, abilitati_id 3, neam_id 5, nume 'Bilbo', tip_personalitate 'complet-neutrui');

500 ✓
501 ✓
502 ✓ INSERT INTO personaj (utilizator_id, abilitati_id, neam_id, nume, tip_personalitate)
VALUES (utilizator_id 5, abilitati_id 4, neam_id 1, nume 'Reiko', tip_personalitate 'haotic-neutrui');

503 ✓
504 ✓
505 ✓ INSERT INTO personaj (utilizator_id, abilitati_id, neam_id, nume, tip_personalitate)
VALUES (utilizator_id 2, abilitati_id 5, neam_id 3, nume 'Esteria', tip_personalitate 'haotic-neutrui');

506 ✓
507 ✓
508 ✓ INSERT INTO personaj (utilizator_id, abilitati_id, neam_id, nume, tip_personalitate)
VALUES (utilizator_id 3, abilitati_id 6, neam_id 1, nume 'Nyx', tip_personalitate 'haotic-malefic');

509 ✓
510

```

Database @127.0.0.1

console 154 ms

Output SYSTEM.NEAM SYSTEMUTILIZATOR SYSTEM.ARBORE_DE_ABILITATI

```

SYSTEM> SELECT * FROM ARBORE_DE_ABILITATI
[2025-05-19 16:36:31] 5 rows retrieved starting from 1 in 849 ms (execution: 8 ms, fetching: 841 ms)
SYSTEM> INSERT INTO personaj (utilizator_id, abilitati_id, neam_id, nume, tip_personalitate)
VALUES (1, 2, 2, 'Cyrus', 'legiuit-bun')
[2025-05-19 16:38:15] 1 row affected in 7 ms
SYSTEM> INSERT INTO personaj (utilizator_id, abilitati_id, neam_id, nume, tip_personalitate)
VALUES (1, 3, 5, 'Bilbo', 'complet-neutrui')
[2025-05-19 16:38:15] 1 row affected in 12 ms
SYSTEM> INSERT INTO personaj (utilizator_id, abilitati_id, neam_id, nume, tip_personalitate)
VALUES (5, 4, 1, 'Reiko', 'haotic-neutrui')
[2025-05-19 16:38:15] 1 row affected in 9 ms
SYSTEM> INSERT INTO personaj (utilizator_id, abilitati_id, neam_id, nume, tip_personalitate)
VALUES (2, 5, 3, 'Esteria', 'haotic-neutrui')
[2025-05-19 16:38:15] 1 row affected in 7 ms
SYSTEM> INSERT INTO personaj (utilizator_id, abilitati_id, neam_id, nume, tip_personalitate)
VALUES (3, 6, 1, 'Nyx', 'haotic-malefic')
[2025-05-19 16:38:15] 1 row affected in 6 ms

```

TATI

511 ✓ SELECT * FROM PERSONAJ;

Output SYSTEM.PERSONAJ SYSTEMUTILIZATOR SYSTEM.ARBORE_DE_ABILITATI

PERSONAJ_ID	UTILIZATOR_ID	ABILITATI_ID	NEAM_ID	NUME	TIP_PERSONALITATE
1	3	1	2	Cyrus	legiuit-bun
2	4	1	3	Bilbo	complet-neutrui
3	5	5	4	Reiko	haotic-neutrui
4	6	2	5	Esteria	haotic-neutrui
5	7	3	6	Nyx	haotic-malefic

8. PERSONAJ_CLASA:

Cod Sql:

```
INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel)
VALUES (3, 1, 5);

INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel)
VALUES (3, 6, 2);

INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel)
VALUES (4, 2, 3);

INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel)
VALUES (4, 3, 1);

INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel)
VALUES (5, 5, 10);

INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel)
VALUES (5, 4, 1);

INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel)
VALUES (5, 2, 2);

INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel)
VALUES (6, 2, 3);

INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel)
VALUES (7, 2, 4);

INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel)
VALUES (7, 3, 2);
```

PrintScreen:

1 2 of 37 533 534 ✓ INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel)

	Output	SYSTEM.PERSONAJ	SYSTEM.CLASA	SYSTEM.ARBORE_DE_ABILITATI
base	SYSTEM> INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel) VALUES (3, 1, 5)			
@127.0.0.1	[2025-05-19 16:51:43] 1 row affected in 25 ms			
console 495 ms	SYSTEM> INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel) VALUES (3, 6, 2)			
console 35 ms	[2025-05-19 16:51:43] 1 row affected in 10 ms			
	SYSTEM> INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel) VALUES (4, 2, 3)			
	[2025-05-19 16:51:43] 1 row affected in 6 ms			
	SYSTEM> INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel) VALUES (4, 3, 1)			
	[2025-05-19 16:51:43] 1 row affected in 6 ms			
	SYSTEM> INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel) VALUES (5, 5, 10)			
	[2025-05-19 16:51:43] 1 row affected in 8 ms			
	SYSTEM> INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel) VALUES (5, 4, 1)			
	[2025-05-19 16:51:43] 1 row affected in 6 ms			
	SYSTEM> INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel) VALUES (5, 2, 2)			
	[2025-05-19 16:51:43] 1 row affected in 8 ms			
	SYSTEM> INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel) VALUES (6, 2, 3)			
	[2025-05-19 16:51:43] 1 row affected in 9 ms			
	SYSTEM> INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel) VALUES (7, 2, 4)			
	[2025-05-19 16:51:43] 1 row affected in 5 ms			
	SYSTEM> INSERT INTO personaj_clasa (personaj_id, clasa_id, nivel) VALUES (7, 3, 2)			
	[2025-05-19 16:51:43] 1 row affected in 7 ms			

542 543 ✓ SELECT * FROM PERSONAJ_CLASA; 544

	Output	SYSTEM.PERSONAJ	SYSTEM.CLASA	SYSTEM.PERSONAJ_CLASA
ABILITIES				
ABILITYTIES				
52 ms				
262 ms				
		PERSONAJ_ID	CLASA_ID	NIVEL
		1	3	1
		2	3	6
		3	4	2
		4	4	3
		5	5	5
		6	5	4
		7	5	2
		8	6	2
		9	7	2
		10	7	3

9. PERSONAJ_CAMPANIE:

Cod Sql:

```
INSERT INTO personaj_campanie (personaj_id, campanie_id)
VALUES (3, 1);

INSERT INTO personaj_campanie (personaj_id, campanie_id)
VALUES (3, 2);

INSERT INTO personaj_campanie (personaj_id, campanie_id)
VALUES (4, 1);

INSERT INTO personaj_campanie (personaj_id, campanie_id)
VALUES (4, 5);

INSERT INTO personaj_campanie (personaj_id, campanie_id)
VALUES (5, 5);

INSERT INTO personaj_campanie (personaj_id, campanie_id)
VALUES (6, 1);

INSERT INTO personaj_campanie (personaj_id, campanie_id)
VALUES (6, 3);

INSERT INTO personaj_campanie (personaj_id, campanie_id)
VALUES (7, 3);

INSERT INTO personaj_campanie (personaj_id, campanie_id)
VALUES (7, 4);

INSERT INTO personaj_campanie (personaj_id, campanie_id)
VALUES (7, 2);
```

PrintScreen:

```
Console 38 ms
SYSTEM> INSERT INTO personaj_camparie (personaj_id, camparie_id)
VALUES (3, 1)
[2025-05-19 16:59:23] 1 row affected in 7 ms
SYSTEM> INSERT INTO personaj_camparie (personaj_id, camparie_id)
VALUES (3, 2)
[2025-05-19 16:59:23] 1 row affected in 8 ms
SYSTEM> INSERT INTO personaj_camparie (personaj_id, camparie_id)
VALUES (4, 1)
[2025-05-19 16:59:23] 1 row affected in 6 ms
SYSTEM> INSERT INTO personaj_camparie (personaj_id, camparie_id)
VALUES (4, 5)
[2025-05-19 16:59:24] 1 row affected in 14 ms
SYSTEM> INSERT INTO personaj_camparie (personaj_id, camparie_id)
VALUES (5, 5)
[2025-05-19 16:59:24] 1 row affected in 5 ms
SYSTEM> INSERT INTO personaj_camparie (personaj_id, camparie_id)
VALUES (6, 1)
[2025-05-19 16:59:24] 1 row affected in 5 ms
SYSTEM> INSERT INTO personaj_camparie (personaj_id, camparie_id)
VALUES (6, 3)
[2025-05-19 16:59:24] 1 row affected in 5 ms
SYSTEM> INSERT INTO personaj_camparie (personaj_id, camparie_id)
VALUES (7, 3)
[2025-05-19 16:59:24] 1 row affected in 4 ms
SYSTEM> INSERT INTO personaj_camparie (personaj_id, camparie_id)
VALUES (7, 4)
[2025-05-19 16:59:24] 1 row affected in 6 ms
SYSTEM> INSERT INTO personaj_camparie (personaj_id, camparie_id)
VALUES (7, 2)
[2025-05-19 16:59:24] 1 row affected in 7 ms
```

575 ✓ SELECT * FROM personaj_camparie;

	PERSONAJ_ID	CAMPANIE_ID
1	3	1
2	3	2
3	4	1
4	4	5
5	5	5
6	6	1
7	6	3
8	7	3
9	7	4
10	7	2

10 rows ▾ :

10. ARHETIP:

Cod Sql:

```
INSERT INTO arhetip (arhetip_id, clasa_id, nume)
VALUES (clase_si_arhetipuri.nextval, 2, 'Hoț');

INSERT INTO arhetip (arhetip_id, clasa_id, nume)
VALUES (clase_si_arhetipuri.nextval, 2, 'Asasin');

INSERT INTO arhetip (arhetip_id, clasa_id, nume, descriere)
VALUES (clase_si_arhetipuri.nextval, 2, 'Şmecher arcanic', 'Un
escroc ce își alege arhetipul astă va avea acces la vraji
specifice arhetipului (+ în engleză s.n. arcane trickster
lol)');

INSERT INTO arhetip (arhetip_id, clasa_id, nume, descriere)
VALUES (clase_si_arhetipuri.nextval, 1, 'Juramant al
devotamentului', 'Paladinul își jura devotamentul către o
anumită cauză nobila pentru a putea primi puteri speciale');

INSERT INTO arhetip (arhetip_id, clasa_id, nume, descriere)
VALUES (clase_si_arhetipuri.nextval, 1, 'Juramantul Coroanei',
'Paladin ce slujește un nobil și primește abilități
specifice');
```

PrintScreen:

7.0.0.1 2 of 37

```

576 ✓ INSERT INTO arhetip (arhetip_id, clasa_id, nume)
577 VALUES (clase_si_arhetipuri.nextval, 2, 'Hot');
578
579
580 ✓ INSERT INTO arhetip (arhetip_id, clasa_id, nume)
581 VALUES (clase_si_arhetipuri.nextval, 2, 'Asasin');
582
583 ✓ INSERT INTO arhetip (arhetip_id, clasa_id, nume, descriere)
584 VALUES (clase_si_arhetipuri.nextval, 2, 'Smecher arcanic', 'Un escroc ce isi alege arhetipul asta va avea acces la vreji specifice arhetipului');
585
586 ✓ INSERT INTO arhetip (arhetip_id, clasa_id, nume, descriere)
587 VALUES (clase_si_arhetipuri.nextval, 1, 'Juramant al devotamentului', 'Paladinul isi jura devotamentul catre o anumita cauza nobila pentru care este obligat sa se implice');
588
589 ✓ INSERT INTO arhetip (arhetip_id, clasa_id, nume, descriere)
590 VALUES (clase_si_arhetipuri.nextval, 1, 'Juramantul Coroanei', 'Paladin ce slujeste un nobil si primeste abilitati specifice');

```

Database @127.0.0.1

console 199 ms

Output ARHETIP PERSONAJ_CAMPANIE SYSTEM.PERSONAJ_CAMPANIE

```

[2025-05-19 17:01:43] 6 rows retrieved starting from 1 in 206 ms (execution: 6 ms, fetching: 200 ms)
SYSTEM> INSERT INTO arhetip (arhetip_id, clasa_id, nume)
VALUES (clase_si_arhetipuri.nextval, 2, 'Hot')
[2025-05-19 17:09:00] 1 row affected in 12 ms
SYSTEM> INSERT INTO arhetip (arhetip_id, clasa_id, nume)
VALUES (clase_si_arhetipuri.nextval, 2, 'Asasin')
[2025-05-19 17:09:00] 1 row affected in 8 ms
SYSTEM> INSERT INTO arhetip (arhetip_id, clasa_id, nume, descriere)
VALUES (clase_si_arhetipuri.nextval, 2, 'Smecher arcanic', 'Un escroc ce isi alege arhetipul asta va avea acces la vreji specifice arhetipului');
[2025-05-19 17:09:00] 1 row affected in 9 ms
SYSTEM> INSERT INTO arhetip (arhetip_id, clasa_id, nume, descriere)
VALUES (clase_si_arhetipuri.nextval, 1, 'Juramant al devotamentului', 'Paladinul isi jura devotamentul catre o anumita cauza nobila pentru care este obligat sa se implice');
[2025-05-19 17:09:00] 1 row affected in 6 ms
SYSTEM> INSERT INTO arhetip (arhetip_id, clasa_id, nume, descriere)
VALUES (clase_si_arhetipuri.nextval, 1, 'Juramantul Coroanei', 'Paladin ce slujeste un nobil si primeste abilitati specifice');
[2025-05-19 17:09:00] 1 row affected in 6 ms

```

591

592 ✓ SELECT * FROM ARHETIP;

Output ARHETIP SYSTEM.ARHETIP SYSTEM.PERSONAJ_CAMPANIE

ARHETIP_ID	CLASA_ID	NUME	DESCRIERE
1	7	2 Hot	<null>
2	8	2 Asasin	<null>
3	9	2 Smecher arcanic	Un escroc ce isi alege arhetipul asta va avea acces la vreji specifice arhetipului
4	10	1 Juramant al devotamentului	Paladinul isi jura devotamentul catre o anumita cauza nobila pentru care este obligat sa se implice
5	11	1 Juramantul Coroanei	Paladin ce slujeste un nobil si primeste abilitati specifice

11. SUBNEAM:

Cod Sql:

```
INSERT INTO subneam (subneam_id, neam_id, nume)
VALUES (neamuri_si_subneamuri.nextval, 2, 'Elf al padurii');

INSERT INTO subneam (subneam_id, neam_id, nume)
VALUES (neamuri_si_subneamuri.nextval, 2, 'Elf nobil');

INSERT INTO subneam (subneam_id, neam_id, nume, descriere)
VALUES (neamuri_si_subneamuri.nextval, 3, 'Gnom de piatra',
'Crescut in mine si adanc in interiorul muntilor, acest tip de
gnom este extrem de icsusit si abil cand vine vorba de creearea
armelor, armurilor s.a.m.d.');

INSERT INTO subneam (subneam_id, neam_id, nume, descriere)
VALUES (neamuri_si_subneamuri.nextval, 3, 'Gnom al padurii',
'Padurea a invatat acesti gnomi sa se camunfleze si sa isi
creeze cat mai multe iluzii pentru a evada atat restul
civilizatiei cat si animalele periculoase');

INSERT INTO subneam (subneam_id, neam_id, nume, descriere)
VALUES (neamuri_si_subneamuri.nextval, 1, 'Standard', 'Omul de
zi cu zi');
```

PrintScreen:

1 es 153
 Q\$._INTERNET_AGENT_PRIVS
 Q\$._INTERNET_AGENTS
 Q\$._KEY_SHARD_MAP
 Q\$._QUEUE_TABLES
 Q\$._QUEUES
 Q\$._SCHEDULES
 RBORE_DE_ABILITATI
 RHETIP
 TRIBUT
 ONUS_ABILITATI
 1 columns 2

```

594 ✓ INSERT INTO subneam (subneam_id, neam_id, nume)
VALUES ( subneam_id neamuri_si_subneamuri.nextval, neam_id 2, nume 'Elf al padurii');
596
597 ✓ INSERT INTO subneam (subneam_id, neam_id, nume)
VALUES ( subneam_id neamuri_si_subneamuri.nextval, neam_id 2, nume 'Elf nobil');
598
599
600 ✓ INSERT INTO subneam (subneam_id, neam_id, nume, descriere)
VALUES ( subneam_id neamuri_si_subneamuri.nextval, neam_id 3, nume 'Gnom de piatra', descriere 'Crescut in mine si adanc in interiorul muntilor, acest tip de gnom este foarte rar');
601
602
603 ✓ INSERT INTO subneam (subneam_id, neam_id, nume, descriere)
VALUES ( subneam_id neamuri_si_subneamuri.nextval, neam_id 3, nume 'Gnom al padurii', descriere 'Padurea a invatat aceste gnomi sa se camunfleze si sa isi creeze cat');
604
605
  
```

base 127.0.0.1
 console 144 ms
 console 26 ms

Output ARHETIP SYSTEM.ARHETIP SYSTEM.NEAM

```

SYSTEM> SELECT * FROM NEAM
[2025-05-19 17:11:51] 7 rows retrieved starting from 1 in 308 ms (execution: 8 ms, fetching: 300 ms)
SYSTEM> INSERT INTO subneam (subneam_id, neam_id, nume)
VALUES (neamuri_si_subneamuri.nextval, 2, 'Elf al padurii')
[2025-05-19 17:18:26] 1 row affected in 7 ms
SYSTEM> INSERT INTO subneam (subneam_id, neam_id, nume)
VALUES (neamuri_si_subneamuri.nextval, 2, 'Elf nobil')
[2025-05-19 17:18:26] 1 row affected in 7 ms
SYSTEM> INSERT INTO subneam (subneam_id, neam_id, nume, descriere)
VALUES (neamuri_si_subneamuri.nextval, 3, 'Gnom de piatra', 'Crescut in mine si adanc in interiorul muntilor, acest tip de gnom este foarte rar');
[2025-05-19 17:18:26] 1 row affected in 5 ms
SYSTEM> INSERT INTO subneam (subneam_id, neam_id, nume, descriere)
VALUES (neamuri_si_subneamuri.nextval, 3, 'Gnom al padurii', 'Padurea a invatat aceste gnomi sa se camunfleze si sa isi creeze cat');
[2025-05-19 17:18:26] 1 row affected in 5 ms
  
```

606 ✓ INSERT INTO subneam (subneam_id, neam_id, nume, descriere)
607 VALUES (subneam_id neamuri_si_subneamuri.nextval, neam_id 1, nume 'Standard', descriere 'Omul de zi cu zi');

608

Output ARHETIP SYSTEM.ARHETIP SYSTEM.NEAM

```

VALUES (neamuri_si_subneamuri.nextval, 3, 'Gnom al padurii', 'Padurea a invatat aceste gnomi sa se camunfleze si sa isi creeze cat');
[2025-05-19 17:18:26] 1 row affected in 5 ms
SYSTEM> INSERT INTO subneam (subneam_id, neam_id, nume, descriere)
VALUES (neamuri_si_subneamuri.nextval, 1, 'Standard', 'Omul de zi cu zi')
[2025-05-19 17:20:14] 1 row affected in 8 ms
  
```

609 ✓ SELECT * FROM subneam;

610

Output SYSTEM.SUBNEAM × SYSTEM.ARHETIP SYSTEM.NEAM

SUBNEAM_ID	NEAM_ID	NUME	DESCRIERE
1	8	2 Elf al padurii	<null>
2	9	2 Elf nobil	<null>
3	10	3 Gnom de piatra	Crescut in mine si adanc in interiorul muntilor, acest tip de gnom este foarte rar
4	11	3 Gnom al padurii	Padurea a invatat aceste gnomi sa se camunfleze si sa isi creeze cat
5	12	1 Standard	Omul de zi cu zi

5 rows ↴ :

12. ATRIBUT:

Cod Sql:

```
INSERT INTO atribut (nume, descriere, bonus)
VALUES ('Spellcasting', 'Calitate a unui personaj de a face
vraji', '+vraji specific');

INSERT INTO atribut (nume, descriere, bonus)
VALUES ('Imbunatatire arbore de abilitati', 'Posibilitatea de a
modifica arborele de abilitati', '+2 la o abilitate/ +1 la doua
abilitati');

INSERT INTO atribut (nume, descriere, bonus)
VALUES ('Atac Extra', 'Personajul, atunci cand loveste fizic un
inamic (fie cu o arma de lunga distanta, una impovizata sau
una de distanta mica), poate sa mai efectueze un atac
consecutiv', '+1 attack');

INSERT INTO atribut (nume, bonus)
VALUES ('Imunitate Divina', '+inmunitate catre boli si
otrava');

INSERT INTO atribut (nume, descriere, bonus)
VALUES ('Val de actiune', 'Abilitate ce, odata folosita, poate
fi refolosita doar dupa ce personajul se odihneste; Ofera
posibilitatea personajului sa atace inca o data intr-o singura
tura (orice actiune nu doar atac fizic)', '+1 actiune');
```

PrintScreen:

```

611 ✓ INSERT INTO atribut (nume, descriere, bonus)
612   VALUES ( nume 'Spellcasting', descriere 'Calitate a unui personaj de a face vraji', bonus '+vraji specifice');
613
614 ✓ INSERT INTO atribut (nume, descriere, bonus)
615   VALUES ( nume 'Imbunatatire arbore de abilitati', descriere 'Posibilitatea de a modifica arborele de abilitati', bonus '+2 la o abilitate/ +1 la doua abilitati');
616
617 ✓ INSERT INTO atribut (nume, descriere, bonus)
618   VALUES ( nume 'Atac Extra', descriere 'Personajul, atunci cand loveste fizic un inamic (fie cu o arma de lunga distanta, una impovizata sau una de distanta) are o probabilitate de 10% sa infligă dano');
619
620 ✓ INSERT INTO atribut (nume, bonus)
621   VALUES ( nume 'Imunitate Divina', bonus '+inmunitate catre boli si otrava');
622
623 ✓ INSERT INTO atribut (nume, descriere, bonus)
624   VALUES ( nume 'Val de actiune', descriere 'Abilitate ce, odata folosita, poate fi refolosita doar dupa ce personajul se odihneste; Ofera posibilitatea de a recupera 10% din san');
625
626

```

```

Output SYSTEM.SUBNEAM SYSTEM.ARHETIP SYSTEM.NEAM
[2025-05-19 17:37:56] 5 rows retrieved starting from 1 in 409 ms (execution: 76 ms, retching: 365 ms)
SYSTEM> INSERT INTO atribut (nume, descriere, bonus)
VALUES ('Spellcasting', 'Calitate a unui personaj de a face vraji', '+vraji specifice')
[2025-05-19 17:37:56] 1 row affected in 10 ms
SYSTEM> INSERT INTO atribut (nume, descriere, bonus)
VALUES ('Imbunatatire arbore de abilitati', 'Posibilitatea de a modifica arborele de abilitati', '+2 la o abilitate/ +1 la doua abilitati');
[2025-05-19 17:37:56] 1 row affected in 6 ms
SYSTEM> INSERT INTO atribut (nume, descriere, bonus)
VALUES ('Atac Extra', 'Personajul, atunci cand loveste fizic un inamic (fie cu o arma de lunga distanta, una impovizata sau una de distanta) are o probabilitate de 10% sa infligă dano');
[2025-05-19 17:37:56] 1 row affected in 6 ms
SYSTEM> INSERT INTO atribut (nume, descriere, bonus)
VALUES ('Imunitate Divina', '+inmunitate catre boli si otrava')
[2025-05-19 17:37:56] 1 row affected in 6 ms
SYSTEM> INSERT INTO atribut (nume, descriere, bonus)
VALUES ('Val de actiune', 'Abilitate ce, odata folosita, poate fi refolosita doar dupa ce personajul se odihneste; Ofera posibilitatea de a recupera 10% din san');
[2025-05-19 17:37:56] 1 row affected in 7 ms

```

```

UE_TABLES      626 ✓ SELECT * FROM ATRIBUT;
UES
EDULES
DE_ABILITATI

```

	ATRIBUT_ID	NUME	DESCRIERE	BONUS
1	1	Spellcasting	Calitate a unui personaj de a face vraji	+vraji specifice
2	2	Imbunatatire arbore de abilitati	Posibilitatea de a modifica arborele de abilitati	+2 la o abilitate/ +1 la doua abilitati
3	3	Atac Extra	Personajul, atunci cand loveste fizic un inamic (fie cu o... +1 attack	
4	4	Imunitate Divina	<null>	+inmunitate catre boli si otrava
5	5	Val de actiune	Abilitate ce, odata folosita, poate fi refolosita doar du... +1 actiune	

13. ECHIPAMENT:

Cod Sql:

```
INSERT INTO atribut (nume, descriere)
VALUES ('Pumnal', 'Arma mica, usor de ascuns');
INSERT INTO echipament (atribut_id, greutate, putere,
tip_echipament, tip_lovitura, distanta)
VALUES (26, 1, '1d4', 'simplu', 'strapungere', 1);

INSERT INTO atribut (nume)
VALUES ('Arbaleta usoara');
INSERT INTO echipament (atribut_id, greutate, putere,
tip_echipament, tip_lovitura, distanta)
VALUES (27, 2, '1d8', 'simplu', 'strapungere', 80);

INSERT INTO atribut (nume, descriere, bonus)
VALUES ('Arc scurt vrajit', 'Arma ce poate lovi la distanta,
tinuta cu doua maini', '+1 catre loviturii');
INSERT INTO echipament (atribut_id, greutate, putere,
tip_echipament, tip_lovitura, distanta)
VALUES (28, 1, '1d6', 'simplu', 'strapungere', 80);

INSERT INTO atribut (nume)
VALUES ('Spada');
INSERT INTO echipament (atribut_id, greutate, putere,
tip_echipament, tip_lovitura, distanta)
VALUES(29, 1, '1d8', 'militar', 'strapungere', 1);

INSERT INTO atribut (nume)
VALUES ('Ciocan de razboi');
INSERT INTO echipament (atribut_id, greutate, putere,
tip_echipament, tip_lovitura, distanta)
VALUES (30, 3, '1d8', 'militar', 'maciuca', 1);
```

PrintScreen:

```
SYSTEM> INSERT INTO atribut (nume, descriere, bonus)
      VALUES ('Arc scurt vrajit', 'Arma ce poate lovi la distanta, tinuta cu doua maini', '+1 catre lovitur')
[2025-05-19 21:52:28] 1 row affected in 4 ms
SYSTEM> INSERT INTO echipament (atribut_id, greutate, putere, tip_echipament, tip_lovitura, distanta)
      VALUES (27, 1, '1d6', 'simplu', 'strapungere', 80)
```

```
SYSTEM> INSERT INTO echipament (atribut_id, greutate, putere, tip_echipament, tip_lovitura, distanta)
      VALUES (28, 1, '1d6', 'simplu', 'strapungere', 80)
[2025-05-19 21:53:05] 1 row affected in 4 ms
SYSTEM> INSERT INTO atribut (nume)
      VALUES ('Spada')
[2025-05-19 21:53:05] 1 row affected in 3 ms
SYSTEM> INSERT INTO echipament (atribut_id, greutate, putere, tip_echipament, tip_lovitura, distanta)
      VALUES(29, 1, '1d8', 'martial', 'strapungere', 1)
```

```
[2025-05-19 21:55:35] Position: 0
SYSTEM> INSERT INTO echipament (atribut_id, greutate, putere, tip_echipament, tip_lovitura, distanta)
      VALUES(29, 1, '1d8', 'militar', 'strapungere', 1)
[2025-05-19 21:56:03] 1 row affected in 6 ms
```

```
SYSTEM> INSERT INTO atribut (nume)
      VALUES ('Ciocan de razboi')
[2025-05-19 21:56:09] 1 row affected in 5 ms
SYSTEM> INSERT INTO echipament (atribut_id, greutate, putere, tip_echipament, tip_lovitura, distanta)
      VALUES (30, 3, '1d8', 'militar', 'maciuca', 1)
[2025-05-19 21:56:09] 1 row affected in 7 ms
```

	CLASA	COMPONENTA	COMPONENTA_VRA	ECHIPAMENT	HELP	ITEM.ARHETIP	SELECT * FROM ATIBUT LEFT JOIN ECHIPAMENT 1<->0..1 ON ECHIPAMENT.atribut_id = ATIBUT.atribut_id;	SELECT CLASE_SI_ARHETIPURI.curval FROM DUAL;
1								
2								
3								
4								
5								
6								
7								
8								
9								
10								

	BONUS	ECHIPAMENT.ATRIBUT_ID	DESCRIERE	GREUTATE	PUTERE	TIPECHIPAMENT	TIPOLOVITURA	DISTANTA
1	+vraji specific	1	Spellcasting	Calitate a unui personaj de a face vraji	<null>	<null>	<null>	<null>
2	+2 la o abilitate/ +1 la doua abilitati	2	Imbutatire arbore de abilitati	Possibilitatea de a modifica arborele de abilitati	<null>	<null>	<null>	<null>
3	+1 attack	3	Atac Extra	Personajul, atunci cand loveste fizic un inamic (fie cu oarma de lunga ...	<null>	<null>	<null>	<null>
4	+inmunitate catre boli si otrava	4	Imunitate Divina	Personajul, atunci cand loveste fizic un inamic (fie cu oarma de lunga ...	<null>	<null>	<null>	<null>
5	+1 actiune	5	Val de actiune	Abilitate ce, odata folosita, poate fi refolosita doar dupa ce personajul	<null>	<null>	<null>	<null>
6	+1 attack	26	Pumnal	Arma mica, usor de ascuns	<null>	<null>	<null>	<null>
7	+inmunitate catre boli si otrava	27	Arbalesta usoara	<null>	<null>	<null>	<null>	<null>
8	+1 catre lovitururi	28	Arc scurt vrajit	Arma ce poate lovi la distanta, tinuta cu doua maini	<null>	<null>	<null>	<null>
9	+1 catre lovitururi	29	Spada	<null>	<null>	<null>	<null>	<null>
10	+1 catre lovitururi	30	Ciocan de razboi	<null>	<null>	<null>	<null>	<null>

14. VRAJA:

Cod Sql:

```
INSERT INTO atribut (nume, descriere)
VALUES ('Minge de foc', 'O explozie masiva de foc ce arde totul
intr-o anumita zona');
INSERT INTO vraja (atribut_id, putere, tip_lovitura, durata,
scoala)
VALUES (31, '8d6', 'foc', 'instantaneu', 'evocare');

INSERT INTO atribut (nume, descriere, bonus)
VALUES ('Vindeca Rani', 'O vraja ce poate vindeca orice rana',
'In loc de lovitura se va adauga acelasi numar catre viata unui
aliat');
INSERT INTO vraja (atribut_id, putere, tip_lovitura, durata,
scoala)
VALUES (32, '1d8+Bonus', 'vindecare', 'instantaneu',
'evocare');

INSERT INTO atribut (nume, descriere, bonus)
VALUES ('Invizibilitate', 'Face o creatura invizibila pentru o
perioada limitata de timp', 'atata timp cat un personaj nu face
zgomot are avantaj la zarurile de d20');
INSERT INTO vraja (atribut_id, durata, timp_necesar, scoala)
VALUES (33, 'o ora', 'o actiune', 'iluzie');

INSERT INTO atribut (nume, descriere)
VALUES ('Farmece', 'Mici iluzii care nu modifica cu mult lumea
din jurul lor, personajul ce are aceasta vraja poate aprinde
sau stinge o lumanare, pot face cateva lumini sa apara, pot
curata o pata si asa mai departe');
INSERT INTO vraja (atribut_id, durata, timp_necesar, scoala)
VALUES (34, 'pana la o ora', 'instantaneu', 'transmutatie');

INSERT INTO atribut (nume, descriere)
VALUES ('Scut magic', 'Creeaza un scut temporar ce face
personajul mai usor de lovit');
INSERT INTO vraja (atribut_id, durata, timp_necesar, scoala)
VALUES ('35', 'pana ce personajul se odihneste', 'o runda',
'abjurare');
```

PrintScreen:

```
SYSTEM> INSERT INTO atribut (nume, descriere)
VALUES ('Minge de foc', 'O explozie masiva de foc ce arde totul intr-o anumita zona')
[2025-05-19 22:27:56] 1 row affected in 4 ms
SYSTEM> INSERT INTO vraja (atribut_id, putere, tip_lovitura, durata, scoala)
SYSTEM> INSERT INTO vraja (atribut_id, putere, tip_lovitura, durata, scoala)
VALUES (31, '8d6', 'foc', 'instantaneu', 'evocare')
[2025-05-19 22:29:28] 1 row affected in 7 ms
SYSTEM> INSERT INTO atribut (nume, descriere, bonus)
VALUES ('Vindeca Rani', 'O vraja ce poate vindeca orice rana', 'In loc de lovitura se va adauga acelasi numar catre viata unui aliat')
[2025-05-19 22:29:33] 1 row affected in 5 ms

SYSTEM> INSERT INTO vraja (atribut_id, putere, tip_lovitura, durata, scoala)
VALUES (32, '1d8+Bonus', 'vindecare', 'instantaneu', 'evocare')
[2025-05-19 22:29:53] 1 row affected in 5 ms
SYSTEM> INSERT INTO atribut (nume, descriere, bonus)
VALUES ('Invizibilitate', 'Face o creatura invizibila pentru o perioada limitata de timp', 'atata timp cat un personaj nu face zgromot are avantaj la zarurile de d20')
[2025-05-19 22:30:00] 1 row affected in 8 ms
SYSTEM> INSERT INTO vraja (atribut_id, durata, timp_necesar, scoala)
VALUES (33, 'o ora', 'o actiune', 'iluzie')
[2025-05-19 22:30:00] 1 row affected in 5 ms
SYSTEM> SELECT *
SYSTEM> INSERT INTO atribut (nume, descriere)
VALUES ('Farmece', 'Mici iluzii care nu modifica cu mult lumea din jurul lor, personajul ce are aceasta vraja poate aprinde sau')
[2025-05-19 22:47:09] 1 row affected in 6 ms
SYSTEM> INSERT INTO vraja (atribut_id, durata, timp_necesar, scoala)
VALUES (34, 'pana la o ora', 'instantaneu', 'transmutatie')
[2025-05-19 22:47:09] 1 row affected in 3 ms
SYSTEM> INSERT INTO atribut (nume, descriere)
VALUES ('Scut magic', 'Creeaza un scut temporar ce face personajul mai usor de lovitur')
[2025-05-19 22:50:55] 1 row affected in 5 ms
SYSTEM> INSERT INTO vraja (atribut_id, durata, timp_necesar, scoala)
VALUES ('35', 'pana ce personajul se odihneste', 'o runda', 'abjurare')
[2025-05-19 22:50:55] 1 row affected in 3 ms
SYSTEM> SELECT *
COMPONENTA_VRAJA 692
ECHIPAMENT 693 ✓ SELECT *
HELP 694 FROM ATRIBUT
ITEM_ARHETIP 695 --LEFT JOIN ECHIPAMENT ON ECHIPAMENT.atribut_id = ATRIBUT.atribut_id
ITEM_CLASA 696 LEFT JOIN VRAJA 1<->0..1 ON VRAJA.atribut_id = ATRIBUT.atribut_id
ITEM_NEAM 697 WHERE vraja.atribut_id IN (SELECT ATRIBUT_ID FROM VRAJA);|
ITEM_NEAM 698
Output Result 124 x Result 120 Result 123
ATRIBUT.ATRIBUT_ID NUME DESCRIERE BONUS VRAJA
31 Minge de foc O explozie masiva de foc ce arde totul intr-o anumita zona <null>
32 Vindeca Rani O vraja ce poate vindeca orice rana In loc de lovitura se va adauga acelasi numar catre viata unui aliat
33 Invizibilitate Face o creatura invizibila pentru o perioada limitata de timp atata timp cat un personaj nu face zgromot are avantaj la zarurile de d20
34 Farmece Mici iluzii care nu modifica cu mult lumea din jurul lor, personajul ce ... <null>
35 Scut magic Creeaza un scut temporar ce face personajul mai usor de lovitur <null>
VRAJA 692
693 ✓ SELECT *
FROM ATRIBUT
694
695 --LEFT JOIN ECHIPAMENT ON ECHIPAMENT.atribut_id = ATRIBUT.atribut_id
696 LEFT JOIN VRAJA 1<->0..1 ON VRAJA.atribut_id = ATRIBUT.atribut_id
697 WHERE vraja.atribut_id IN (SELECT ATRIBUT_ID FROM VRAJA);
698
result 124 x Result 120 Result 123
BONUS VRAJA.ATTRIBUT_ID PUTERE TIP_LOVITURA DURATA TEMP_NECESAR SCODALA
<null> 31 8d6 foc instantaneu <null> evocare
In loc de lovitura s... 32 1d8+Bonus vindecare instantaneu <null> evocare
atata timp cat un pe... 33 <null> <null> o ora o actiune iluzie
sonajul ce ... <null> 34 <null> <null> pana la o ora instantaneu transmutatie
mai usor de lovitur <null> 35 <null> <null> pana ce personajul se odihneste o runda abjurare
5 rows :
```

15. ITEM_NEAM:

Cod Sql:

```
INSERT INTO item_neam (neam_id, atribut_id)
VALUES (2, 1);

INSERT INTO item_neam (neam_id, atribut_id)
VALUES (4, 1);

INSERT INTO item_neam (neam_id, atribut_id)
VALUES (7, 1);

INSERT INTO item_neam (neam_id, atribut_id)
VALUES (6, 3);

INSERT INTO item_neam (neam_id, atribut_id)
VALUES (1, 2);

INSERT INTO item_neam (neam_id, atribut_id)
VALUES (2, 2);

INSERT INTO item_neam (neam_id, atribut_id)
VALUES (3, 2);

INSERT INTO item_neam (neam_id, atribut_id)
VALUES (5, 2);

INSERT INTO item_neam (neam_id, atribut_id)
VALUES (7, 36);

INSERT INTO item_neam (neam_id, atribut_id)
VALUES (4, 39);
```

PrintScreen:

```
[2025-05-19 23:21:42] 15 rows retrieved starting from 1 in 292
SYSTEM> INSERT INTO item_neam (neam_id, atribut_id)
      VALUES (2, 1)
[2025-05-19 23:22:35] 1 row affected in 21 ms
SYSTEM> INSERT INTO item_neam (neam_id, atribut_id)
      VALUES (4, 1)
[2025-05-19 23:22:35] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_neam (neam_id, atribut_id)
      VALUES (7, 1)
[2025-05-19 23:22:35] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_neam (neam_id, atribut_id)
      VALUES (6, 3)
[2025-05-19 23:22:35] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_neam (neam_id, atribut_id)
      VALUES (1, 2)
[2025-05-19 23:22:35] 1 row affected in 3 ms
SYSTEM> INSERT INTO item_neam (neam_id, atribut_id)
      VALUES (2, 2)
[2025-05-19 23:22:35] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_neam (neam_id, atribut_id)
      VALUES (3, 2)
[2025-05-19 23:22:36] 1 row affected in 3 ms
SYSTEM> INSERT INTO item_neam (neam_id, atribut_id)
      VALUES (5, 2)
[2025-05-19 23:22:36] 1 row affected in 3 ms
SYSTEM> INSERT INTO item_neam (neam_id, atribut_id)
      VALUES (7, 36)
[2025-05-19 23:22:36] 1 row affected in 3 ms
SYSTEM> INSERT INTO item_neam (neam_id, atribut_id)
      VALUES (4, 39)
[2025-05-19 23:22:36] 1 row affected in 4 ms
```

NEAM_ID	ATRIBUT_ID
1	2
2	1
3	2
4	2
5	1
6	39
7	2
8	3
9	1
10	36

16. ITEM_SUBNEAM:

Cod Sql:

```
INSERT INTO item_subneam (subneam_id, atribut_id)
VALUES (9, 2);

INSERT INTO item_subneam (subneam_id, atribut_id)
VALUES (10, 2);

INSERT INTO item_subneam (subneam_id, atribut_id)
VALUES (11, 2);

INSERT INTO item_subneam (subneam_id, atribut_id)
VALUES (12, 2);

INSERT INTO item_subneam (subneam_id, atribut_id)
VALUES (13, 2);

INSERT INTO item_subneam (subneam_id, atribut_id)
VALUES (10, 4);

INSERT INTO item_subneam (subneam_id, atribut_id)
VALUES (11, 3);

INSERT INTO item_subneam (subneam_id, atribut_id)
VALUES (12, 39);

INSERT INTO item_subneam (subneam_id, atribut_id)
VALUES (9, 39);

INSERT INTO item_subneam (subneam_id, atribut_id)
VALUES (13, 26);
```

PrintScreen:

```

[2025-05-19 23:26:20] 5 rows retrieved starting from 1 in 289 ms (execution)
SYSTEM> INSERT INTO item_subneam (subneam_id, atribut_id)
      VALUES (9, 2)
[2025-05-19 23:29:01] 1 row affected in 6 ms
SYSTEM> INSERT INTO item_subneam (subneam_id, atribut_id)
      VALUES (10, 2)
[2025-05-19 23:29:01] 1 row affected in 6 ms
SYSTEM> INSERT INTO item_subneam (subneam_id, atribut_id)
      VALUES (11, 2)
[2025-05-19 23:29:01] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_subneam (subneam_id, atribut_id)
      VALUES (12, 2)
[2025-05-19 23:29:01] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_subneam (subneam_id, atribut_id)
      VALUES (13, 2)
[2025-05-19 23:29:01] 1 row affected in 3 ms
SYSTEM> INSERT INTO item_subneam (subneam_id, atribut_id)
      VALUES (10, 4)
[2025-05-19 23:29:01] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_subneam (subneam_id, atribut_id)
      VALUES (11, 3)
[2025-05-19 23:29:01] 1 row affected in 5 ms
SYSTEM> INSERT INTO item_subneam (subneam_id, atribut_id)
      VALUES (12, 39)
[2025-05-19 23:29:01] 1 row affected in 3 ms
SYSTEM> INSERT INTO item_subneam (subneam_id, atribut_id)
      VALUES (9, 39)
[2025-05-19 23:29:01] 1 row affected in 3 ms
SYSTEM> INSERT INTO item_subneam (subneam_id, atribut_id)
      VALUES (13, 26)
[2025-05-19 23:29:01] 1 row affected in 4 ms

```

754 SELECT * FROM NEAM,

755 ✓ SELECT * FROM ITEM_SUBNEAM;

756

es

Output SYSTEM.ITEM_SUBNEAM × SYSTEM.ITEM_NEAM SYSTEM.SUBNEAM

ITEM_ARHETIP ITEM_CLASA

	SUBNEAM_ID	ATRIBUT_ID
1	9	2
2	9	39
3	10	2
4	10	4
5	11	2
6	11	3
7	12	2
8	12	39
9	13	2
10	13	26

10 rows

17. ITEM_CLASA:

Cod Sql:

```
INSERT INTO item_clasa (clasa_id, atribut_id)
VALUES (1, 1);

INSERT INTO item_clasa (clasa_id, atribut_id)
VALUES (1, 3);

INSERT INTO item_clasa (clasa_id, atribut_id)
VALUES (1, 4);

INSERT INTO item_clasa (clasa_id, atribut_id)
VALUES (1, 29);

INSERT INTO item_clasa (clasa_id, atribut_id)
VALUES (2, 26);

INSERT INTO item_clasa (clasa_id, atribut_id)
VALUES (2, 27);

INSERT INTO item_clasa (clasa_id, atribut_id)
VALUES (2, 3);

INSERT INTO item_clasa (clasa_id, atribut_id)
VALUES (3, 38);

INSERT INTO item_clasa (clasa_id, atribut_id)
VALUES (3, 39);

INSERT INTO item_clasa (clasa_id, atribut_id)
VALUES (3, 40);

INSERT INTO item_clasa (clasa_id, atribut_id)
VALUES (3, 36);

INSERT INTO item_clasa (clasa_id, atribut_id)
VALUES (4, 3);

INSERT INTO item_clasa (clasa_id, atribut_id)
VALUES (4, 2);

INSERT INTO item_clasa (clasa_id, atribut_id)
VALUES (5, 40);

INSERT INTO item_clasa (clasa_id, atribut_id)
VALUES (6, 39);
```

PrintScreen:

```
SYSTEM> INSERT INTO item_clasa (clasa_id, atribut_id)
      VALUES (1, 1)
[2025-05-19 23:44:31] 1 row affected in 5 ms
SYSTEM> INSERT INTO item_clasa (clasa_id, atribut_id)
      VALUES (1, 3)
[2025-05-19 23:44:31] 1 row affected in 5 ms
SYSTEM> INSERT INTO item_clasa (clasa_id, atribut_id)
      VALUES (1, 4)
[2025-05-19 23:44:31] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_clasa (clasa_id, atribut_id)
      VALUES (1, 29)
[2025-05-19 23:44:31] 1 row affected in 3 ms
SYSTEM> INSERT INTO item_clasa (clasa_id, atribut_id)
      VALUES (2, 26)
[2025-05-19 23:44:31] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_clasa (clasa_id, atribut_id)
      VALUES (2, 27)
[2025-05-19 23:44:31] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_clasa (clasa_id, atribut_id)
      VALUES (2, 3)
[2025-05-19 23:44:31] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_clasa (clasa_id, atribut_id)
      VALUES (3, 38)
[2025-05-19 23:44:31] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_clasa (clasa_id, atribut_id)
      VALUES (3, 39)
[2025-05-19 23:44:31] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_clasa (clasa_id, atribut_id)
      VALUES (3, 40)
[2025-05-19 23:44:31] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_clasa (clasa_id, atribut_id)
      VALUES (3, 36)

[2025-05-19 23:44:31] 1 row affected in 3 ms
SYSTEM> INSERT INTO item_clasa (clasa_id, atribut_id)
      VALUES (4, 5)
[2025-05-19 23:44:31] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_clasa (clasa_id, atribut_id)
      VALUES (4, 2)
[2025-05-19 23:44:32] 1 row affected in 3 ms
SYSTEM> INSERT INTO item_clasa (clasa_id, atribut_id)
      VALUES (5, 40)
[2025-05-19 23:44:32] 1 row affected in 3 ms
SYSTEM> INSERT INTO item_clasa (clase_id, atribut_id)
      VALUES (6, 39)
[2025-05-19 23:44:32] 1 row affected in 4 ms
SYSTEM> SELECT * FROM ITEM_CLASA
[2025-05-19 23:44:46] 15 rows retrieved starting from 1 in 211 ms (execution: 8 ms, fetching: 203 ms)
```

The screenshot shows a database interface with a table named 'ITEM_CLASA'. The table has two columns: 'CLASA_ID' and 'ATIBUT_ID'. The data is as follows:

CLASA_ID	ATIBUT_ID
1	1
2	1
3	4
4	1
5	29
6	3
7	26
8	2
9	27
10	3
11	36
12	3
13	38
14	39
15	40

18. ITEM_ARHETIP:

Cod Sql:

```
INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (7, 3);

INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (7, 27);

INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (7, 28);

INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (8, 29);

INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (9, 38);

INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (9, 39);

INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (10, 40);

INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (11, 39);

INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (11, 5);

INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (9, 1);
```

PrintScreen:

```
[2025-05-20 00:37:17] 3 rows retrieved starting from 1 in 347 ms (execution: 17 ms, fetching: 330 ms)
SYSTEM> INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (7, 3)
[2025-05-20 01:02:10] 1 row affected in 28 ms
SYSTEM> INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (7, 27)
[2025-05-20 01:02:10] 1 row affected in 5 ms
SYSTEM> INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (7, 28)
[2025-05-20 01:02:10] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (8, 29)
[2025-05-20 01:02:10] 1 row affected in 5 ms
SYSTEM> INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (9, 38)
[2025-05-20 01:02:10] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (9, 39)
[2025-05-20 01:02:10] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (10, 40)
[2025-05-20 01:02:10] 1 row affected in 2 ms
SYSTEM> INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (11, 39)
[2025-05-20 01:02:10] 1 row affected in 3 ms
SYSTEM> INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (11, 5)
[2025-05-20 01:02:10] 1 row affected in 4 ms
SYSTEM> INSERT INTO item_arhetip (arhetip_id, atribut_id)
VALUES (9, 1)
[2025-05-20 01:02:10] 1 row affected in 2 ms
```

836 ✓ SELECT * FROM ITEM_ARHETIP;

837

838 SELECT *

839 FROM ATRIBUT

840 --LEFT JOIN ECHIPAMENT ON ECHIPAMENT.atribut_id = ATRIBUT.atribut_id

	ARHETIP_ID	ATRIBUT_ID
1	7	3
2	7	27
3	7	28
4	8	29
5	9	1
6	9	38
7	9	39
8	10	40
9	11	5
10	11	39

10 rows

19. COMPONENTA:

Cod Sql:

```
INSERT INTO componenta (componenta_id, tip_componenta,
descriere)
VALUES (componente.nextval, 'vocala (V)', 'un cantec la
alegere');

INSERT INTO componenta (componenta_id, tip_componenta,
descriere)
VALUES (componente.nextval, 'vocala (V)', 'cuvintele "Sesam
deschide-te');

INSERT INTO componenta (componenta_id, tip_componenta,
descriere)
VALUES (componente.nextval, 'somatic (S)', 'un gest la
alegere');

INSERT INTO componenta (componenta_id, tip_componenta,
descriere)
VALUES (componente.nextval, 'material (M)', 'un talisman ce
poarta noroc');

INSERT INTO componenta (componenta_id, tip_componenta,
descriere)
VALUES (componente.nextval, 'material (M)', 'busuioc');
```

PrintScreen:

```
832 ✓ INSERT INTO componenta (componenta_id, tip_componenta, descriere)
833   VALUES ( componente.nextval, 'vocala (V)', 'un cantec la alegere');
834
835 ✓ INSERT INTO componenta (componenta_id, tip_componenta, descriere)
836   VALUES ( componente.nextval, 'vocala (V)', 'cuvintele "Sesam deschide-te");
837
838 ✓ INSERT INTO componenta (componenta_id, tip_componenta, descriere)
839   VALUES ( componente.nextval, 'somatic (S)', 'un gest la alegere');
840
841 ✓ INSERT INTO componenta (componenta_id, tip_componenta, descriere)
842   VALUES ( componente.nextval, 'material (M)', 'un talisman ce poarta noroc');
843
844 ✓ INSERT INTO componenta (componenta_id, tip_componenta, descriere)
845   VALUES ( componente.nextval, 'material (M)', 'busuioc');

[2025-05-20 01:04:51] completed in 10 ms
SYSTEM> INSERT INTO componenta (componenta_id, tip_componenta, descriere)
      VALUES (componente.nextval, 'vocala (V)', 'un cantec la alegere')
[2025-05-20 01:12:46] 1 row affected in 13 ms
SYSTEM> INSERT INTO componenta (componenta_id, tip_componenta, descriere)
      VALUES (componente.nextval, 'vocala (V)', 'cuvintele "Sesam deschide-te')
[2025-05-20 01:12:46] 1 row affected in 4 ms
SYSTEM> INSERT INTO componenta (componenta_id, tip_componenta, descriere)
      VALUES (componente.nextval, 'somatic (S)', 'un gest la alegere')
[2025-05-20 01:12:46] 1 row affected in 3 ms
SYSTEM> INSERT INTO componenta (componenta_id, tip_componenta, descriere)
      VALUES (componente.nextval, 'material (M)', 'un talisman ce poarta noroc')
[2025-05-20 01:12:46] 1 row affected in 6 ms
SYSTEM> INSERT INTO componenta (componenta_id, tip_componenta, descriere)
      VALUES (componente.nextval, 'material (M)', 'busuioc')
[2025-05-20 01:12:46] 1 row affected in 5 ms
```

```
857 ✓ SELECT * FROM COMPONENTA;
858
```

Output SYSTEM.COMPONENTA × SYSTEM.ARHETIP SYSTEM.ITEM_ARHETIP

COMPONENTA_ID	TIP_COMPONENTA	DESCRIERE
1	1 vocala (V)	un cantec la alegere
2	2 vocala (V)	cuvintele "Sesam deschide-te
3	3 somatic (S)	un gest la alegere
4	4 material (M)	un talisman ce poarta noroc
5	5 material (M)	busuioc

20. COMPONENTA_VRAJA:

Cod Sql:

```
INSERT INTO componenta_vraja (componenta_id, atribut_id)
VALUES (1, 36);

INSERT INTO componenta_vraja (componenta_id, atribut_id)
VALUES (1, 38);

INSERT INTO componenta_vraja (componenta_id, atribut_id)
VALUES (1, 39);

INSERT INTO componenta_vraja (componenta_id, atribut_id)
VALUES (3, 37);

INSERT INTO componenta_vraja (componenta_id, atribut_id)
VALUES (3, 40);

INSERT INTO componenta_vraja (componenta_id, atribut_id)
VALUES (4, 40);

INSERT INTO componenta_vraja (componenta_id, atribut_id)
VALUES (1, 37);

INSERT INTO componenta_vraja (componenta_id, atribut_id)
VALUES (2, 36);

INSERT INTO componenta_vraja (componenta_id, atribut_id)
VALUES (5, 37);

INSERT INTO componenta_vraja (componenta_id, atribut_id)
VALUES (5, 40);
```

PrintScreen:

```
[2025-05-20 01:18:48] 15 rows retrieved starting from 1 in 117 ms (execution: 7 ms, fetching: 110 ms)
SYSTEM> INSERT INTO componenta_vraja (componenta_id, atribut_id)
      VALUES (1, 36)
[2025-05-20 01:18:33] 1 row affected in 9 ms
SYSTEM> INSERT INTO componenta_vraja (componenta_id, atribut_id)
      VALUES (1, 38)
[2025-05-20 01:18:33] 1 row affected in 5 ms
SYSTEM> INSERT INTO componenta_vraja (componenta_id, atribut_id)
      VALUES (1, 39)
[2025-05-20 01:18:33] 1 row affected in 4 ms
SYSTEM> INSERT INTO componenta_vraja (componenta_id, atribut_id)
      VALUES (3, 37)
[2025-05-20 01:18:33] 1 row affected in 3 ms
SYSTEM> INSERT INTO componenta_vraja (componenta_id, atribut_id)
      VALUES (3, 40)
[2025-05-20 01:18:33] 1 row affected in 3 ms
SYSTEM> INSERT INTO componenta_vraja (componenta_id, atribut_id)
      VALUES (4, 40)
[2025-05-20 01:18:33] 1 row affected in 5 ms
SYSTEM> INSERT INTO componenta_vraja (componenta_id, atribut_id)
      VALUES (1, 37)
[2025-05-20 01:18:33] 1 row affected in 3 ms
SYSTEM> INSERT INTO componenta_vraja (componenta_id, atribut_id)
      VALUES (2, 36)
[2025-05-20 01:18:33] 1 row affected in 4 ms
SYSTEM> INSERT INTO componenta_vraja (componenta_id, atribut_id)
      VALUES (5, 37)
[2025-05-20 01:18:33] 1 row affected in 4 ms
SYSTEM> INSERT INTO componenta_vraja (componenta_id, atribut_id)
      VALUES (5, 40)
[2025-05-20 01:18:33] 1 row affected in 5 ms
SYSTEM> SELECT * FROM COMPONENTA_VRAJA
[2025-05-20 01:19:06] 10 rows retrieved starting from 1 in 185 ms (execution: 8 ms, fetching: 177 ms)
```

The screenshot shows a database management system interface. At the top, there is a toolbar with icons for Output, Refresh, Undo, Redo, Insert, Delete, Select, DDL, and Search. Below the toolbar, the title bar displays 'Output' and the table names 'SYSTEM.COMPONENTA_VRAJA' and 'SYSTEM.ATRIBUT'. The main area contains a table with two columns: 'COMPONENTA_ID' and 'ATRIBUT_ID'. The data is as follows:

COMPONENTA_ID	ATRIBUT_ID
1	36
2	37
3	38
4	39
5	36
6	37
7	40
8	40
9	37
10	40

21. SEDINTĂ:

Cod Sql:

```
INSERT INTO sedinta (campanie_id, data_inceput, data_final,
ora_inceput, ora_final)
VALUES (1, TO_DATE('2025-05-10', 'YYYY-MM-DD'),
TO_DATE('2025-05-11', 'YYYY-MM-DD'), '20:34','01:40');

INSERT INTO sedinta (campanie_id, data_inceput, ora_inceput,
ora_final)
VALUES (2, TO_DATE('2025-04-20', 'YYYY-MM-DD'), '13:02',
'16:40');

INSERT INTO sedinta (campanie_id, data_inceput, ora_inceput)
VALUES (3, TO_DATE('2025-04-24', 'YYYY-MM-DD'), '15:20');

INSERT INTO sedinta (campanie_id, data_inceput, ora_inceput,
ora_final)
VALUES (3, TO_DATE('2025-05-01', 'YYYY-MM-DD'), '16:30',
'19:16');

INSERT INTO sedinta (campanie_id, data_inceput, ora_inceput)
VALUES (4, TO_DATE('2025-03-01', 'YYYY-MM-DD'), '12:30');
```

PrintScreen:

```
SYSTEM> INSERT INTO sedinta (campanie_id, data_inceput, data_final, ora_inceput, ora_final)
      VALUES (1, TO_DATE('2025-05-10', 'YYYY-MM-DD'), TO_DATE('2025-05-11', 'YYYY-MM-DD'), '20:34','01:40')
[2025-05-20 01:57:50] 1 row affected in 7 ms
SYSTEM> INSERT INTO sedinta (campanie_id, data_inceput, ora_inceput, ora_final)
      VALUES (2, TO_DATE('2025-04-20', 'YYYY-MM-DD'), '13:02', '16:40')
[2025-05-20 01:57:50] 1 row affected in 6 ms
SYSTEM> INSERT INTO sedinta (campanie_id, data_inceput, ora_inceput)
      VALUES (3, TO_DATE('2025-04-24', 'YYYY-MM-DD'), '15:20')
[2025-05-20 01:57:50] 1 row affected in 7 ms
SYSTEM> INSERT INTO sedinta (campanie_id, data_inceput, ora_inceput, ora_final)
      VALUES (3, TO_DATE('2025-05-01', 'YYYY-MM-DD'), '16:30', '19:16')
[2025-05-20 01:57:50] 1 row affected in 3 ms
SYSTEM> INSERT INTO sedinta (campanie_id, data_inceput, ora_inceput)
      VALUES (4, TO_DATE('2025-03-01', 'YYYY-MM-DD'), '12:30')
[2025-05-20 01:57:50] 1 row affected in 3 ms
SYSTEM> select * from sedinta
[2025-05-20 01:57:57] 10 rows retrieved starting from 1 in 115 ms (execution: 7 ms, fetching: 108 ms)
```

SEDINTA_ID	CAMPAINE_ID	DATA_INCEPUT	DATA_FINAL	ORA_INCEPUT	ORA_FINAL
1	12	1 2025-05-10	2025-05-11	20:34	01:40
2	13	2 2025-04-20	<null>	13:02	16:40
3	14	3 2025-04-24	<null>	15:20	<null>
4	15	3 2025-05-01	<null>	16:30	19:16
5	16	4 2025-03-01	<null>	12:30	<null>

22. PARTICIPARE:

Cod Sql:

```
INSERT INTO participare (personaj_id, sedinta_id, prezenta)
VALUES (3,12, 'da');

INSERT INTO participare (personaj_id, sedinta_id, prezenta)
VALUES (3,13, 'da');

INSERT INTO participare (personaj_id, sedinta_id, prezenta)
VALUES (4,12, 'da');

INSERT INTO participare (personaj_id, sedinta_id, prezenta)
VALUES (5,12, 'nu');

INSERT INTO participare (personaj_id, sedinta_id, prezenta)
VALUES (6,12, 'da');

INSERT INTO participare (personaj_id, sedinta_id, prezenta)
VALUES (7,13, 'da');

INSERT INTO participare (personaj_id, sedinta_id, prezenta)
VALUES (7,14, 'da');

INSERT INTO participare (personaj_id, sedinta_id, prezenta)
VALUES (7,12, 'nu');

INSERT INTO participare (personaj_id, sedinta_id, prezenta)
VALUES (3,14, 'da');

INSERT INTO participare (personaj_id, sedinta_id, prezenta)
VALUES (5,14, 'da');
```

PrintScreen:

```
SYSTEM> INSERT INTO participare (personaj_id, sedinta_id, prezenta)
      VALUES (3,12, 'da')
[2025-05-20 02:01:09] 1 row affected in 5 ms
SYSTEM> INSERT INTO participare (personaj_id, sedinta_id, prezenta)
      VALUES (3,13, 'da')
[2025-05-20 02:01:09] 1 row affected in 11 ms
SYSTEM> INSERT INTO participare (personaj_id, sedinta_id, prezenta)
      VALUES (4,12, 'da')
[2025-05-20 02:01:09] 1 row affected in 5 ms
SYSTEM> INSERT INTO participare (personaj_id, sedinta_id, prezenta)
      VALUES (5,12, 'nu')
[2025-05-20 02:01:09] 1 row affected in 3 ms
SYSTEM> INSERT INTO participare (personaj_id, sedinta_id, prezenta)
      VALUES (6,12, 'da')
[2025-05-20 02:01:09] 1 row affected in 5 ms
SYSTEM> INSERT INTO participare (personaj_id, sedinta_id, prezenta)
      VALUES (7,13, 'da')
[2025-05-20 02:01:09] 1 row affected in 5 ms
SYSTEM> INSERT INTO participare (personaj_id, sedinta_id, prezenta)
      VALUES (7,14, 'da')
[2025-05-20 02:01:09] 1 row affected in 4 ms
SYSTEM> INSERT INTO participare (personaj_id, sedinta_id, prezenta)
      VALUES (7,12, 'nu')
[2025-05-20 02:01:09] 1 row affected in 5 ms
SYSTEM> INSERT INTO participare (personaj_id, sedinta_id, prezenta)
      VALUES (3,14, 'da')
[2025-05-20 02:01:09] 1 row affected in 4 ms
SYSTEM> INSERT INTO participare (personaj_id, sedinta_id, prezenta)
      VALUES (5,14, 'da')
[2025-05-20 02:01:09] 1 row affected in 4 ms
```

The screenshot shows the Oracle SQL Developer interface. At the top, there is a session history with three entries:

- 941: select * from sedinta;
- 942: SELECT * FROM PARTICIPARE;
- 943:

The main area shows the results of the second query in a grid format:

PERSONAJ_ID	SEDINTA_ID	PREZENTA
1	3	12 da
2	3	13 da
3	4	12 da
4	5	12 nu
5	6	12 da
6	7	13 da
7	7	14 da
8	7	12 nu
9	3	14 da
10	5	14 da

At the bottom left, it says "Console > @127.0.0.1 > console".

12. Formulați în limbaj natural și implementați 5 cereri SQL complexe ce vor utiliza, în ansamblul lor, următoarele elemente:

- a) subcereri sincronizate în care intervin cel puțin 3 tabele
 - b) subcereri nesincronizate în clauza FROM
 - c) grupări de date, funcții grup, filtrare la nivel de grupuri cu subcereri nesincronizate (în clauza de HAVING)
 - d) ordonări și utilizarea funcțiilor NVL și DECODE (în cadrul aceleiași cereri)
 - e) utilizarea a cel puțin 2 funcții pe siruri de caractere, 2 funcții pe date calendaristice, a cel puțin unei expresii CASE
 - f) utilizarea a cel puțin 1 bloc de cerere (clauza WITH)
- Observație: Într-o cerere se vor regăsi mai multe elemente dintre cele enumerate mai sus, astfel încât cele 5 cereri să le cuprindă pe toate.

I. Prima Cerere

(exercițiu din feedback, îndeplinește subpunctele **a** și **b**)

Enunt:

Pentru fiecare personaj din clasa ‘vrajitor’ care a participat la ședinte din campania ‘Werewolves’ să se afișeze lista atributelor de tip vrajă pe care le are.

Rezolvare în cod sql:

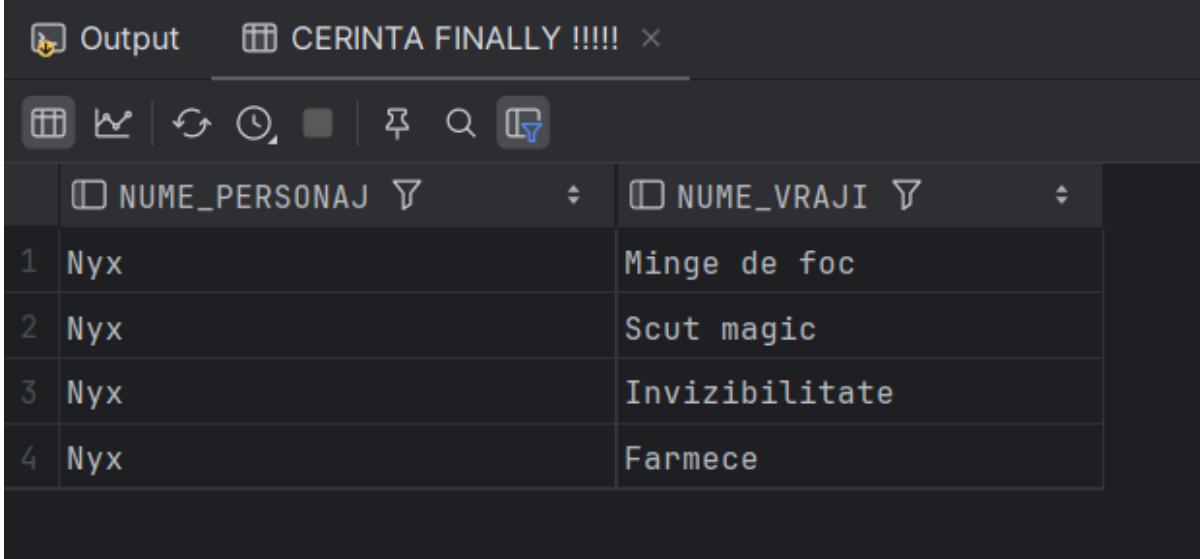
```
SELECT
    p2.nume_personaj,
    vraji_clasa.nume_vraji
FROM (
    SELECT
        p.personaj_id,
        p.nume AS nume_personaj,
        c.nume AS nume_clasa,
        c.clasa_id
    FROM personaj p
    LEFT JOIN personaj_clasa pc ON p.personaj_id =
pc.personaj_id
    LEFT JOIN clasa c ON pc.clasa_id = c.clasa_id
    WHERE c.nume='Vrajitor'
    GROUP BY p.personaj_id, p.nume, c.nume, c.clasa_id
) p2
LEFT JOIN personaj_campaine pc ON p2.personaj_id =
pc.personaj_id
LEFT JOIN campaine c ON pc.campaine_id = c.campaine_id
LEFT JOIN (
    SELECT
        c.clasa_id,
        c.nume AS nume_clasa_vrajitor,
        v.nume AS nume_vraji
```

```

FROM clasa c
LEFT JOIN item_clasa ic on c.clasa_id = ic.clasa_id
LEFT JOIN (
    SELECT
        a.atribut_id,
        a.nume,
        a.descriere,
        a.bonus,
        v.durata,
        v.putere,
        v.scoala,
        v.timp_necesar
    FROM atribut a
    LEFT JOIN vraja v on v.atribut_id = a.atribut_id
    WHERE a.atribut_id IN (SELECT v2.atribut_id FROM
vraja v2)
        ) v on ic.atribut_id = v.atribut_id
    WHERE c.nume='Vrajitor'
    GROUP BY c.nume, v.nume, c.clasa_id
) vraji_clasa ON vraji_clasa.clasa_id = p2.clasa_id
WHERE c.nume = 'Werewolves'
GROUP BY p2.nume_personaj, vraji_clasa.nume_vraji;

```

Output:



	NUME_PERSONAJ	NUME_VRAJI
1	Nyx	Minge de foc
2	Nyx	Scut magic
3	Nyx	Invizibilitate
4	Nyx	Farmece

II. A Doua Cerere (pentru subpunctul c)

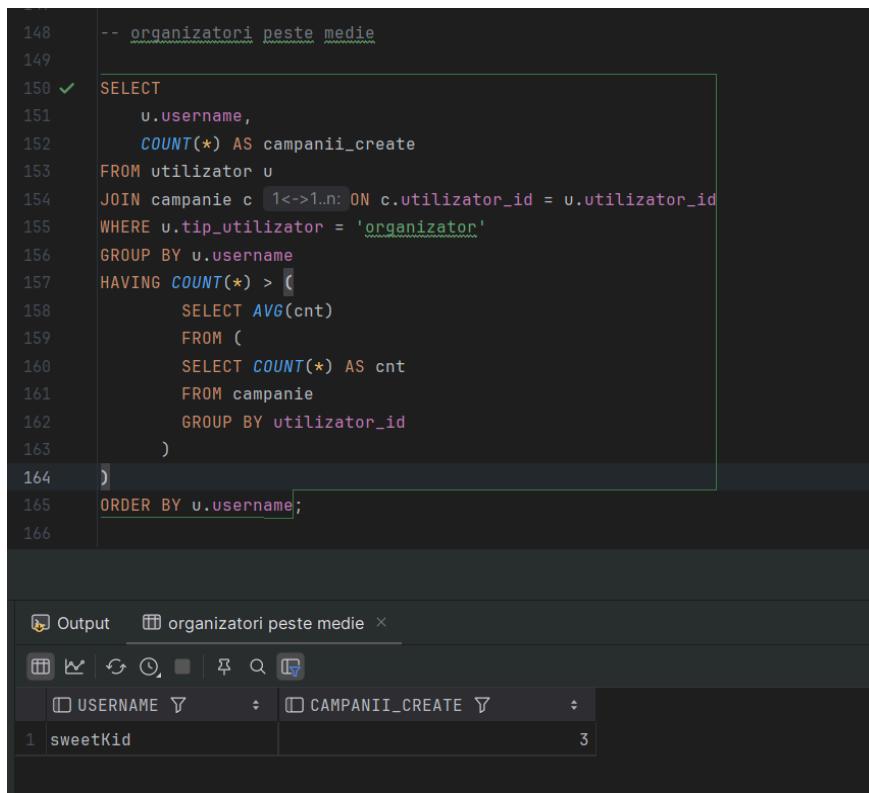
Enunt:

Să se afișeze numele și prenumele organizatorilor ce au creat mai multe campanii decât media generală a campaniilor pe utilizator.

Rezolvare în cod sql:

```
SELECT
    u.username,
    COUNT(*) AS campanii_create
FROM utilizator u
JOIN campanie c ON c.utilizator_id = u.utilizator_id
WHERE u.tip_utilizator = 'organizator'
GROUP BY u.username
HAVING COUNT(*) > (
    SELECT AVG(cnt)
    FROM (
        SELECT COUNT(*) AS cnt
        FROM campanie
        GROUP BY utilizator_id
    )
)
ORDER BY u.username;
```

Output:



The screenshot shows a code editor with a dark theme and a terminal window below it. The code editor displays a SQL query with line numbers from 148 to 166. The terminal window has a title bar 'Output organizeri peste medie'. It shows a table with two columns: 'USERNAME' and 'CAMPANII_CREATE'. There is one row with the value 'sweetKid' in the 'USERNAME' column and '3' in the 'CAMPANII_CREATE' column.

USERNAME	CAMPANII_CREATE
sweetKid	3

III. A Treia Cerere

(pentru subpunctele *d* și *e*)

Enunt:

Să se afișeze primii 4 utilizatori în felul următor: numele complet să apară concatenat; să se afișeze doar partea din email de dinainte caracterului @; numărul de telefon, în cazul în care utilizatorul ar avea un câmp NULL, să fie înlocuit cu *necunoscut*; tipul utilizatorului în mod normal (adică jucător sau organizator); data înregistrării să fie formatată; să se calculeze lunile de la înregistrare (câte luni au trecut de la înregistrarea utilizatorului) și să se afișeze o etichetă în funcție de rezultatul calculului (Nou/Vechi pentru mai puțin de 30 de zile, respectiv mai mult de 30 de zile).

Rezolvare în cod sql:

```
SELECT
    UPPER(u.nume || ', ' || u.prenume) AS nume_afisat,
    SUBSTR(u.email,1,INSTR(u.email,'@')-1) AS local_email,
    NVL(TO_CHAR(u.numar_telefon), 'necunoscut') AS telefon,
    DECODE(u.tip_utilizator,
        'jucator', 'Jucător',
        'organizator', 'Organizator',
        'necunoscut') AS tip_textual,
    TO_CHAR(u.data_inregistrare,'DD.MM.YYYY') AS data_fmt,
    ROUND(MONTHS_BETWEEN(SYSDATE, u.data_inregistrare),1) AS
luni_de_la,
    CASE
        WHEN u.data_inregistrare > ADD_MONTHS(SYSDATE,-1)
THEN 'Nou'
        ELSE 'Vechi'
    END AS statut_cont
FROM utilizator u
ORDER BY u.nume, u.prenume
FETCH FIRST 4 ROWS ONLY;
```

Output:

	NUME_AFISAT	LOCAL_EMAIL	TELEFON	TIP_TEXTUAL	DATA_FMT	LUNI_DE_LA	STATUT_CONT
1	BLINTON, FELIX	paradolix	712345678	Organizator	04.05.2025		0.5 Nou
2	BRANZA, MURA	mura	727343434	Jucător	26.03.2025		1.8 Vechi
3	EIOSANU, RAY	rayzor.sharp	799999999	Jucător	07.04.2025		1.4 Vechi
4	POSCU, TEO	teo.rosca	727236404	Organizator	15.05.2025		0.2 Nou

IV. A Patra Cerere

(pentru subpunctul *f*)

Enunt:

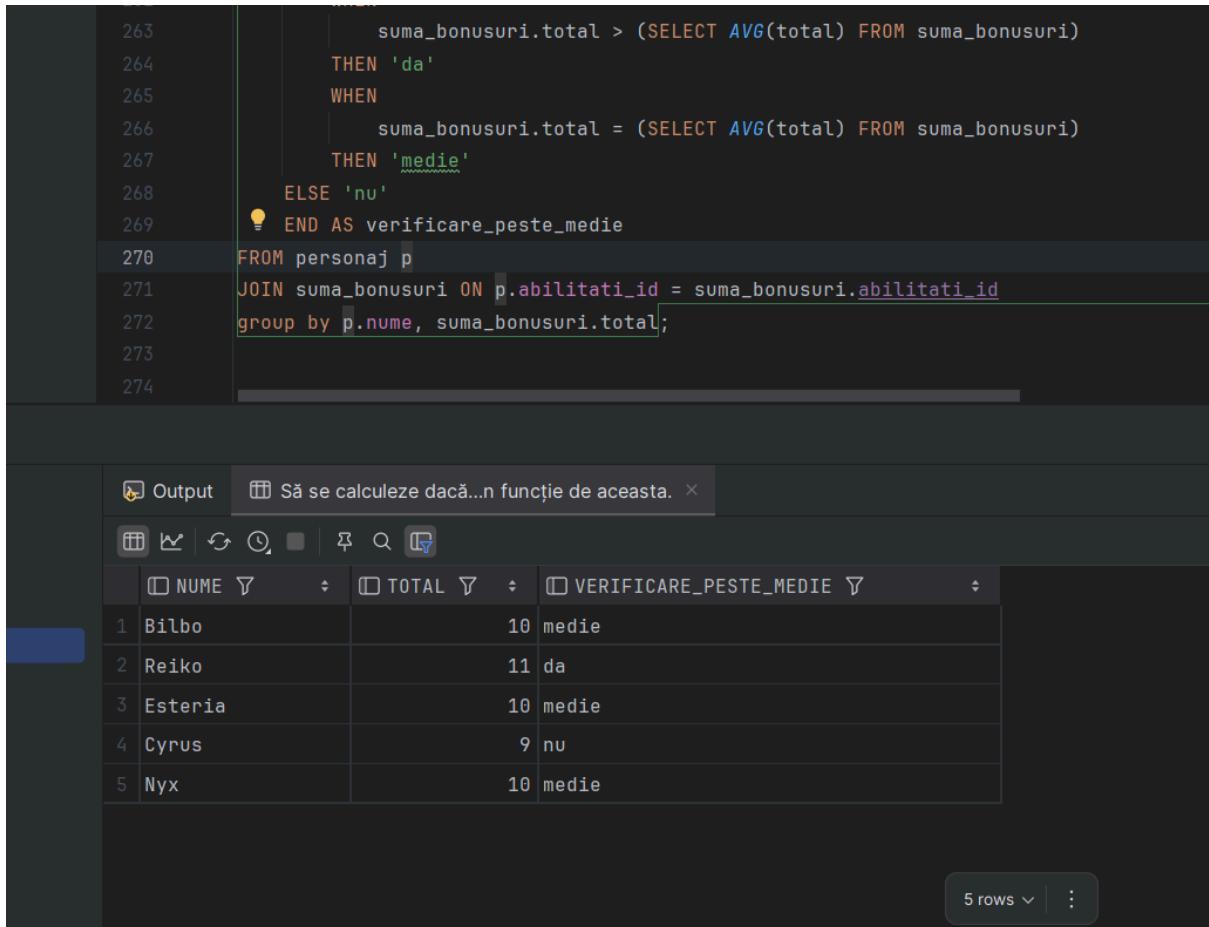
Să se verifice dacă puterea unui personaj este mai mare decât media puterii personajelor presupunând că ‘puterea’ se calculează ca suma bonusurilor aferente fiecărui scor din arbore_de_abilitati și bonus_abilitati. Să se afișeze numele personajului și eticheta ‘da’ sau ‘nu’ în funcție de rezultat.

Rezolvare în cod sql:

```
WITH traducere_lookup AS (
    SELECT
        ab.abilitati_id,
        b.bonus AS bonusChar,
        b2.bonus AS bonusCon,
        b3.bonus AS bonusStr,
        b4.bonus AS bonusDex,
        b5.bonus AS bonusInt,
        b6.bonus AS bonusWis
    FROM arbore_de_abilitati ab
    JOIN bonus_abilitati b ON ab.scor_charisma = b.scor
    JOIN bonus_abilitati b2 ON ab.scor_constitutie = b2.scor
    JOIN bonus_abilitati b3 ON ab.scor_putere = b3.scor
    JOIN bonus_abilitati b4 ON ab.scor_dexteritate = b4.scor
    JOIN bonus_abilitati b5 ON ab.scor_inteligenta = b5.scor
    JOIN bonus_abilitati b6 ON ab.scor_intelepciune = b6.scor
),
suma_bonusuri AS (
    SELECT
        abilitati_id,
        bonusWis + bonusStr + bonusInt + bonusCon + bonusChar +
        bonusDex AS total
    FROM traducere_lookup
)
SELECT
    p.nume,
    suma_bonusuri.total,
    CASE
        WHEN
            suma_bonusuri.total > (SELECT AVG(total) FROM
            suma_bonusuri)
            THEN 'da'
        WHEN
            suma_bonusuri.total = (SELECT AVG(total) FROM
            suma_bonusuri)
            THEN 'medie'
        ELSE 'nu'
    END AS verificare_peste_medie
```

```
FROM personaj p
JOIN suma_bonusuri ON p.abilitati_id =
suma_bonusuri.abilitati_id
group by p.nume, suma_bonusuri.total;
```

Output:



The screenshot shows a database interface with two main sections: a code editor at the top and a results table below it.

Code Editor (Lines 263-274):

```
263     |         suma_bonusuri.total > (SELECT AVG(total) FROM suma_bonusuri)
264     |         THEN 'da'
265     |         WHEN
266     |             suma_bonusuri.total = (SELECT AVG(total) FROM suma_bonusuri)
267     |             THEN 'medie'
268     |             ELSE 'nu'
269     |         END AS verificare_peste_medie
270 FROM personaj p
271 JOIN suma_bonusuri ON p.abilitati_id = suma_bonusuri.abilitati_id
272 group by p.nume, suma_bonusuri.total;
273
274
```

Results Table:

	NUME	TOTAL	VERIFICARE_PESTE_MEDIE
1	Bilbo	10	medie
2	Reiko	11	da
3	Esteria	10	medie
4	Cyrus	9	nu
5	Nyx	10	medie

V. A Cincea Cerere

(exercițiu din feedback, îndeplinește parțial cerința c)

Enunt:

Pentru fiecare utilizator să se afișeze numărul de campanii pe care le-a organizat și numărul de personaje pe care le deține.

Rezolvare în cod sql:

```
SELECT
    (u.nume || ' ' || u.prenume) AS nume_complet,
    u.tip_utilizator,
    COUNT(DISTINCT c.campanie_id) AS nr_campanii_organizate,
    COUNT(DISTINCT p.personaj_id) AS nr_personaje_detinute
FROM utilizator u
LEFT JOIN campanie c ON u.utilizator_id = c.utilizator_id
LEFT JOIN personaj p ON u.utilizator_id = p.utilizator_id
GROUP BY u.nume, u.prenume, u.tip_utilizator;
```

Output:

Output WHERE u.tip_utilizator = 'organizator';				
	NUME_COMPLET	TIP_UTILIZATOR	NR_CAMPANII_ORGANIZATE	NR_PERSONAJE_DETINUTE
1	Sofr Luca	jucator	0	1
2	Poscu Teo	organizator	3	2
3	Branza Mura	jucator	0	1
4	Blinton Felix	organizator	2	1
5	Eiosanu Ray	jucator	0	0

13. Implementarea a 3 operații de actualizare și de suprimare a datelor utilizând subcereri.

I. Prima Operatie:

Operatie scrisă în limbaj normal:

Actualizarea nivelului personajelor cu clasa Paladin ce participă în campania ‘NethRodas gods’:

Cod Sql:

```
UPDATE personaj_clasa pc
SET nivel = nivel + 1
WHERE pc.personaj_id IN (
    SELECT p.personaj_id
    FROM personaj p
    JOIN personaj_companie pca ON p.personaj_id =
        pca.personaj_id
    WHERE pca.companie_id = (
        SELECT companie_id
        FROM companie
        WHERE nume = 'NethRodas gods'
    )
)
AND pc.clasa_id = (
    SELECT clasa_id
    FROM clasa
    WHERE nume = 'Paladin'
);
```

PrintScreen:

Înainte de a modifica:

```

147
148 ✓ SELECT
149     PERSONAJ.NUME AS nume_personaj,
150     CAMPANIE.NUME AS nume_campanie,
151     CLASA.NUME AS nume_clasa,
152     PERSONAJ_CLASA.NIVEL AS nivel_clasa
153 FROM PERSONAJ
154 JOIN PERSONAJ_CAMPANIE 1<->1..n: ON PERSONAJ.PERSONAJ_ID = PERSONAJ_CAMPANIE.PERSONAJ_ID
155 JOIN CAMPANIE 1..n<->1: ON PERSONAJ_CAMPANIE.CAMPANIE_ID = CAMPANIE.CAMPANIE_ID
156 JOIN PERSONAJ_CLASA 1<->1..n: ON PERSONAJ.PERSONAJ_ID = PERSONAJ_CLASA.PERSONAJ_ID
157 JOIN CLASA 1..n<->1: ON PERSONAJ_CLASA.CLASA_ID = CLASA.CLASA_ID
158 WHERE UPPER(CAMPANIE.NUME)= 'NETHRODAS GODS'
159 GROUP BY PERSONAJ.NUME, CAMPANIE.NUME, CLASA.NUME, PERSONAJ_CLASA.NIVEL
160 ORDER BY nume_campanie;
161

```

Output EXERCITIUL 13 ×

	NUME_PERSONAJ	NUME_CAMPANIE	NUME_CLASA	NIVEL_CLASA
1	Bilbo	NethRodas gods	Escroc	3
2	Bilbo	NethRodas gods	Vrajitor	1
3	Cyrus	NethRodas gods	Cantaret	2
4	Cyrus	NethRodas gods	Paladin	5
5	Esteria	NethRodas gods	Escroc	3

După modificare:

Output EXERCITIUL 13 ×

	NUME_PERSONAJ	NUME_CAMPANIE	NUME_CLASA	NIVEL_CLASA
1	Bilbo	NethRodas gods	Escroc	3
2	Bilbo	NethRodas gods	Vrajitor	1
3	Cyrus	NethRodas gods	Cantaret	2
4	Cyrus	NethRodas gods	Paladin	6
5	Esteria	NethRodas gods	Escroc	3

II. A doua Operatie:

Operatie scrisă în limbaj normal:

Actualizarea parolei tuturor utilizatorilor ce participă la o campanie organizată de ‘Poscu Teo’.

Cod Sql:

```
UPDATE utilizator u
SET parola = 'nouaParola123'
WHERE u.utilizator_id IN (
    SELECT DISTINCT p.utilizator_id
    FROM personaj p
    JOIN personaj_campagne pc ON p.personaj_id = pc.personaj_id
    JOIN campagne c ON pc.campagne_id = c.campagne_id
    WHERE c.utilizator_id = (
        SELECT utilizator_id
        FROM utilizator
        WHERE nume = 'Poscu' AND prenume = 'Teo'
    )
);
```

PrintScreen:

Înainte de a modifica:

The screenshot shows a database interface with a query window and an output grid. The query window contains the SQL command:

```
184 ✓ SELECT nume, prenume, parola FROM UTILIZATOR;
```

The output grid is titled "SYSTEM.UTILIZATOR" and displays the following data:

	NUME	PRENUME	PAROLA
1	Poscu	Teo	pisici??
2	Sofr	Luca	yayJocDnd
3	Blinton	Felix	D3D3M4N
4	Eiosanu	Ray	nuStiu?
5	Branza	Mura	amUitat

După modificare:

The screenshot shows a database interface with a query window and an output grid. The query window contains the same SQL command as before:

```
184 ✓ SELECT nume, prenume, parola FROM UTILIZATOR;
```

The output grid is titled "SYSTEM.UTILIZATOR" and displays the following data, showing that the password column has been updated:

	NUME	PRENUME	PAROLA
1	Poscu	Teo	nouaParola123
2	Sofr	Luca	nouaParola123
3	Blinton	Felix	nouaParola123
4	Eiosanu	Ray	nuStiu?
5	Branza	Mura	nouaParola123

III. A treia Operatie:

Operatie scrisă în limbaj normal:

Ștergerea obiectelor din tabela Componenta-Vraja a căror vrajă aparține școlii ‘iluzie’.

Cod Sql:

```
DELETE FROM componenta_vraja
WHERE atribut_id IN (
    SELECT atribut_id
    FROM vraja
    WHERE scoala = 'iluzie'
);
```

PrintScreen:

Înainte de a modifica:

	WHERE	ORDER BY
	COMPONENTA_ID	ATRIBUT_ID
1		1 36
2		1 37
3		1 38
4		1 39
5		2 36
6		3 37
7		3 40
8		4 40
9		5 37
10		5 40

După modificare:

10 rows ▾ | :

The screenshot shows a database interface with a toolbar at the top containing various icons for operations like refresh, search, and DDL. Below the toolbar is a table with two columns: 'COMPONENTA_ID' and 'ATRIBUT_ID'. The data consists of nine rows, each with a row number (1-9) and corresponding values for the two columns.

	COMPONENTA_ID	ATRIBUT_ID
1	1	36
2	1	37
3	1	39
4	2	36
5	3	37
6	3	40
7	4	40
8	5	37
9	5	40

9 rows | :

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

Vizualizarea creată are ca scop afișarea personajelor și a utilizatorilor ce le-au creat. Aceasta este complexă deoarece conține operația de JOIN (este unită tabela de utilizatori pentru a putea afișa nimele utilizatorului aferent). O operație LMD ce este permisă asupra acestei vizualizări, pe lângă operația SELECT (care de fapt este și scopul principal al vizualizării- sunt interogate datele într-un mod mai simplu de scris și înțeles), este cea de DELETE. Aceasta este posibilă deoarece nu sunt folosite funcții grup, nu apare clauza de group by și nu este folosit nici cuvântul cheie distinct. De asemenea, este de remarcat faptul că vizualizarea conține o tabelă key preserved, anume tabela personaj. O operație LMD ce nu este permisă asupra vizualizării este cea de INSERT din cauza coloanei de tip expresie “nume complet”.

Cod Sql:

```
CREATE VIEW personaje_si_utilizator AS
SELECT
    p.personaj_id,
    p.abilitati_id,
    p.tip_personalitate,
    p.neam_id,
    p.nume,
    p.utilizator_id,
    (u.nume || ' ' || u.prenume) "nume complet"
FROM personaj p
JOIN utilizator u ON u.utilizator_id = p.utilizator_id;
```

PrintScreen:

```

7098 36 CREATE VIEW personaje_si_utilizator AS
37     SELECT
38         p.personaj_id,
39         p.abilitati_id,
40         p.tip_personalitate,
41         p.neam_id,
42         p.nume,
43         p.utilizator_id,
44         (u.nume || ' ' || u.prenume) "nume complet"
45     FROM personaj p
46     JOIN utilizator u 1..n<->1: ON u.utilizator_id = p.utilizator_id;
47

```

Output SYSTEM.PERSONAJE_SI_UTILIZATOR

```

SYSTEM> --
CREATE OR REPLACE VIEW toate_atributele_personaj AS
SELECT
    P.NUME AS nume_personaj,
    C.NUME AS clasa,
    ATR.ATRIBUT_ID,
    ATR.NUME AS nume_atribut,
    ATR.BONUS AS bonus
FROM PERSONAJ P
JOIN PERSONAJ_CLASA PC ON P.PERSONAJ_ID = PC.PERSONAJ_ID
JOIN CLASA C ON PC.CLASA_ID = C.CLASA_ID
JOIN ITEM_CLASA IC ON C.CLASA_ID = IC.CLASA_ID
JOIN ATRIBUT ATR ON "IC".ATRIBUT_ID = ATR.ATRIBUT_ID
[2025-06-05 21:43:18] completed in 86 ms
SYSTEM> SELECT * FROM TOATE_ATRIBUTELE_PERSONAJ

```

Output SYSTEM.PERSONAJE_SI_UTILIZATOR

PERSONAJ_ID	ABILITATI_ID	TIPI_PERSONALITATE	NEAM_ID	NUME	UTILIZATOR_ID	"nume complet"
1	3	2 Legiușit-bun	2	Cyrus	1	Poscu Teo
2	4	3 complet-neutră	5	Bilbo	1	Poscu Teo
3	6	5 haotic-neutră	3	Esteria	2	Sofr Luca
4	7	6 haotic-malefic	1	Nyx	3	Blinton Felix
5	5	4 haotic-neutră	1	Reiko	5	Branza Mura

15. Formulați în limbaj natural și implementați în SQL: o cerere ce utilizează operația outer-join pe minimum 4 tabele, o cerere ce utilizează operația division și o cerere care implementează analiza top-n.

I. Prima Cerere (utilizează operația outer join)

Pentru a exemplifica într-un mod complet operația de outer join voi realiza următoarele insert-uri în baza de date:

```
INSERT INTO UTILIZATOR (TIP_UTILIZATOR, NUME, PRENUME,  
USERNAME, PAROLA, DATA_INREGISTRARE, EMAIL, NUMAR_TELEFON)  
VALUES ('organizator', 'Stefan', 'Groza', 'grza',  
'aiaieplssss', TO_DATE('12.06.2025', 'DD.MM.YYYY'),  
'groza@gmail.com', 07272344444);  
  
INSERT INTO ARBORE_DE_ABILITATI (SCOR_PUTERE, SCOR_CHARISMA,  
SCOR_CONSTITUTIE, SCOR_INTELIGENTA, SCOR_INTELEPCIUNE,  
SCOR_DEXTERITATE)  
VALUES (10, 14, 20, 10, 8, 15);  
  
INSERT INTO PERSONAJ (UTILIZATOR_ID, ABILITATI_ID, NEAM_ID,  
NUME, TIP_PERSONALITATE)  
VALUES ('1', '41', '3', 'Lacusta', 'legiuit-bun');
```

Enunt:

Să se afișeze toate personajele tuturor utilizatorilor precizându-se numele utilizatorului, numele personajului și numărul de clase ale fiecărui personaj în parte și numărul de campanii la care acesta (personajul) participă.

Cod Sql:

```
SELECT  
    u.nume AS utilizator,  
    p.nume AS personaj,  
    count(DISTINCT persc.clasa_id) AS clase,  
    count(DISTINCT pc.campanie_id) AS campanii  
FROM utilizator u  
LEFT JOIN personaj p ON u.utilizator_id = p.utilizator_id  
LEFT JOIN personaj_campagne pc ON p.personaj_id =  
pc.personaj_id  
LEFT JOIN personaj_clasa persc ON p.personaj_id =  
persc.personaj_id  
GROUP BY u.nume, p.nume ;
```

PrintScreen:

The screenshot shows a database query editor with two main sections: a code editor at the top and a results table below it.

Code Editor (Top):

```
13 ✓    SELECT
14      u.NUME AS utilizator,
15      p.NUME AS personaj,
16      COUNT (DISTINCT persc.CLASA_ID) AS clase,
17      COUNT (DISTINCT pc.CAMPANIE_ID) AS campanii
18  FROM utilizator u
19  LEFT JOIN personaj p  1<->0..n: ON uUTILIZATOR_ID = pUTILIZATOR_ID
20  LEFT JOIN personaj_campanie pc  1<->0..n: ON pPERSONAJ_ID = pcPERSONAJ_ID
21  LEFT JOIN personaj_clasa persc  1<->0..n: ON pPERSONAJ_ID = perscPERSONAJ_ID
22  GROUP BY U.NUME, P.NUME ;|
```

Results Table (Bottom):

	UTILIZATOR	PERSONAJ	CLASE	CAMPANII
1	Sofr	Esteria	1	2
2	Poscu	Bilbo	2	2
3	Poscu	Cyrus	2	2
4	Poscu	Lacusta	0	0
5	Branza	Reiko	3	1
6	Stefan	<null>	0	0
7	Blinton	Nyx	2	3
8	Eiosanu	<null>	0	0

II. A doua Cerere (utilizează division)

Enunt:

Să se afle toate personajele (cod, nume) atașate tuturor campaniilor ce au început mai devreme de anul 2024.

Cod Sql:

```
SELECT personaj_id, nume
FROM personaj
WHERE NOT EXISTS (
    SELECT 1
    FROM campanie c
    WHERE c.data_inceput <= TO_DATE('2023.12.31', 'YYYY.MM.DD')
    AND NOT EXISTS (
        SELECT 1
        FROM personaj p
        JOIN personaj_campainie pc ON p.personaj_id =
pc.personaj_id
        JOIN campanie c2 ON c2.campanie_id = pc.campanie_id
    )
)
);
```

PrintScreen:

The screenshot shows a SQL editor window with the following content:

```
26 ✓  SELECT personaj_id, nume
27   FROM personaj
28   WHERE NOT EXISTS(
29       SELECT 1
30       FROM campanie c
31       WHERE c.data_inceput <= TO_DATE('2023.12.31', 'YYYY.MM.DD')
32       AND NOT EXISTS(
33           SELECT 1
34           FROM personaj p
35           JOIN personaj_campainie pc 1<->1..n: ON p.personaj_id = pc.personaj_id
36           JOIN campanie c2 1..n->>1: ON c2.campanie_id = pc.campanie_id
37       )
38   );
```

Below the code, there is an output pane with the following text:

Output numele personajului ...ersonajul) participă. Să se afle toate per...devreme de anul 2024.

And a results table:

PERSONAJ_ID	NUME
1	3 Cyrus
2	4 Bilbo
3	5 Reiko
4	6 Esteria
5	7 Nyx

III. A treia Cerere (utilizând analiza top n)

Enunt:

Să se afișeze primele 2 personaje cu cel mai mare număr de atribute (date de clasă) și numărul acesta.

Cod Sql:

```
SELECT
    p.personaj_id,
    p.nume,
    atribute
FROM (
    SELECT
        p.personaj_id,
        p.nume,
        COUNT(DISTINCT ic.atribut_id) AS atribute
    FROM personaj p
    JOIN personaj_clasa pc ON p.personaj_id = pc.personaj_id
    JOIN clasa c ON c.clasa_id = pc.clasa_id
    JOIN item_clasa ic ON c.clasa_id = ic.clasa_id
    GROUP BY p.personaj_id, p.nume, p.utilizator_id
    ORDER BY atribute DESC
) p
WHERE ROWNUM <= 2;
```

PrintScreen:

The screenshot shows a code editor and a results viewer. The code editor displays a multi-step SQL query. The results viewer shows the output of the query, which consists of three rows of data.

```
68
69 ✓ SELECT
70     p.personaj_id,
71     p.nume,
72     atribut
73 FROM (
74     SELECT
75         p.personaj_id,
76         p.nume,
77         COUNT(DISTINCT ic.atribut_id) AS atribut
78     FROM personaj p
79     JOIN personaj_clasa pc 1<->1..n: ON p.personaj_id = pc.personaj_id
80     JOIN clasa c 1..n<->1: ON c.clasa_id = pc.clasa_id
81     JOIN item_clasa ic 1<->1..n: ON c.clasa_id = ic.clasa_id
82     GROUP BY p.personaj_id, p.nume, p.utilizator_id
83     ORDER BY atribut DESC
84 ) p
85 WHERE ROWNUM <= 2;
```

Output numele personajului ...ersonajul) participă. Result 24 ×

	PERSONAJ_ID	NUME	ATRIBUTE
1		7 Nyx	7
2		4 Bilbo	7

16. 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), atât anterior cât și ulterior optimizării.

Enunț:

Să se afișeze numele și tipul de personalitate al personajelor create de utilizatorul cu numele de familie „Poscu” și care au neam_id-ul 5.

Cod Sql:

```
SELECT
    p.nume,
    p.tip_personalitate
FROM personaj p
JOIN utilizator u ON p.utilizator_id = u.utilizator_id
WHERE UPPER(u.nume) = 'POSCU' AND p.neam_id = 5;
```

Expresii algebrice:

R1 = SELECT(UTILIZATOR, nume = ‘Poscu’) - se elimină atributile redundante și impune primul criteriu din enunț

R2 = PROJECT(R1, utilizator_id) - eliminare elemente nefolositoare din tabelul utilizator restricționat mai sus

R3 = SELECT(PERSONAJ, neam_id = 5)

R4 = PROJECT(R3, personaj_id, nume, tip_personalitate)

R5 = SEMIJOIN(R4, R2, utilizator_id) - se folosește SEMIJOIN deoarece avem nevoie doar de date dintr-o singură relație

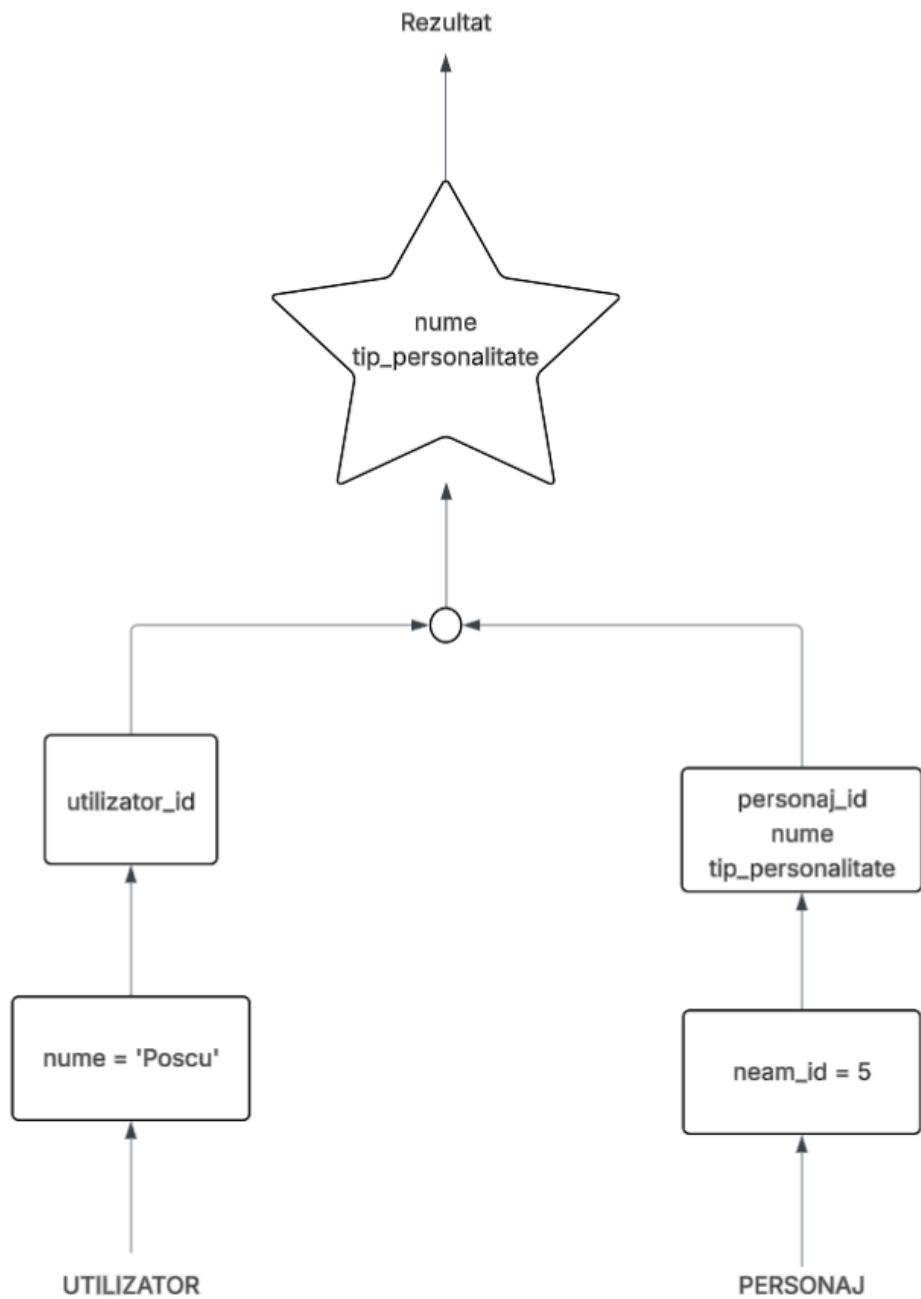
Rezultatul va fi: R6 = PROJECT(R5, nume, tip_personalitate)

Optimizare:

Condițiile de optimizare au fost respectate, anume:

- Se execută selecțiile primele și se elimină elementele redundante
- La fel ca și selecțiile, proiecțiile se execută cât mai repede cu același scop
- Produsele carteziene sunt înlocuite cu join-uri (chiar cu semijoin-uri fiind că erau necesare date dintr-o singură relație, nu din ambele)

Arbore Algebric:



17.

a. Realizarea normalizării BCNF, FN4, FN5.

Normalizarea „Boyce-Codd Normal Form”.

Fie următoarea relație care specifică faptul că un utilizator poate folosi un personaj ce nu îi aparține sau pe care el nu l-a creat pe baza unui abonament/ unei subsecvenții.

REZERVARE(#utilizator_id, #personaj_id, abonament_id, data_imprumut, sedinta_id);

(#utilizator_id, #personaj_id) -> (abonament_id, data_imprumut, sedinta_id)

Se observă că atributul abonament_id este o cheie candidat deoarece un abonament aparține unui singur utilizator:

abonament_id -> utilizator_id.

Deci baza de date nu se află în BCNF și trebuie aplicată regula Casey-Delobel pentru a obține următoarele relații:

REZ1(#utilizator_id, #personaj_id, data_imprumut, sedinta_id);

REZ2(#utilizator_id, #abonament_id);

Astfel, dependențele fiind asociate cheii principale și doar cheii principale, baza de date se află în Boyce-Codd Normal Form.

Forma Normală 4 - FN4:

Considerăm o tabelă nouă numită SPECIALIZARE ce va fi legată de tabela PERSONAJ printr-o relație de 1-to-many. Fiecare specializare va avea anumite clase sau neamuri accesibile. Atunci când specializarea nu conține o anumită clasă sau un anumit neam, personajul legat nu va putea avea acele clase/ neamuri.

Initial tabela va fi structurată în felul următor:

SPECIALIZARE(#nume, clasa, neam);

Problema apare deoarece fiecare specializare are multiple clase și multiple neamuri disponibile. Adică o specializare are o mulțime de clase și o mulțime diferită de neamuri din care poate alege (deci apar două dependențe cu mai multe valori).

Pentru a duce tabela în FN4 va trebui să creăm două tabele separate:

SPECIALIZARE_CLASA(#nume, clasa);

SPECIALIZARE_NEAM(#nume, neam);

Astfel, tabela nu conține multiple dependențe multivalorice și este în FN4.

Forma Normală 5 - FN5:

Se construiește tabela PREFERINȚE ce are o relație de many-to-one cu tabela UTILIZATORI și ce conține pentru fiecare utilizator ce clasă preferă și ce abilități. Totuși, se știe că fiecare clasă are anumite abilități disponibile, deci, pentru tabela PREFERINȚE ar trebui să introducem pentru fiecare abilitate și clasa căruia acesta provine (în exemplul dat se păstrează doar preferințele utilizatorilor fără ca acestea să fie legate de tabelele existente ale claselor sau abilităților din baza de date, având totuși cunoștința căror abilități sunt conținute de ce clase).

Tabela va arăta așa:

PREFERINȚE(#utilizator_id, clasa, abilitate)

Atunci când un utilizator este întrebat ce clase și ce abilități preferă, în baza de date trebuie marcate clasele pentru fiecare abilitate, acestea putându-se repeta. Iar, atunci când se adaugă o clasă preferată pentru un utilizator, aceasta va trebui adăugată în tabel pentru fiecare abilitate preferată a utilizatorului. (dintre cele pe care le are clasa). Deci tabela inițială nu este în FN5

Pentru ca baza de date să fie în FN5 se împarte tabela PREFERINȚE în trei tabele diferite:

```
ABILITATI_DUPA_CLASA(#clasa, abilitate);  
CLASE_PERFERATE_PERSOANA(#utilizator_id, clasa);  
ABILITATI_PREFERATE_PERSOANA(#utilizator_id, abilitate);
```

b. Aplicarea denormalizării, justificând necesitatea acesteia.

Denormalizarea este efectuată cu scopul de a reduce numărul necesar de join-uri pentru o interogare. Aceasta se aplică, de obicei, atunci când relația ce urmează scoasă are o rată de reactualizare minimă, dar o rată a interogărilor ridicată.

Luăm ca și exemplu o tabelă simplificată a atributelor ce păstrează doar numele acestuia și un vaoarea necesară de energie pentru a activa abilitatea. Această coloană (cea de energie) devine repetitivă, aceeași energie definind mai multe abilități. Dacă în baza de date există vreun tabel în care se află energia și numele abilității căreia îi corespunde atunci procesul de denormalizare se poate aplica și este chiar recomandat.