

Proiect Baze de Date

Atelier de tamplarie

PL/SQL

Orice afacere are nevoie să țină pasul cu tehnologia, implicit și cu sisteme de baze de date, iar atelierele de tamplarie nu fac excepție.

Fata de proiectul de semestrul trecut s-a mai adăugat un tabel: *tamplarie_salary_changes* care ține cont de schimbările salariale făcute. Acesta include marca angajatului, salariul nou și cel vechi și data efectuării schimbării.

2. Se va crea un package, care să conțină:

- 4 proceduri stocate, din care 2 publice și 2 private. Cele private vor fi apelate în cadrul package-ului dintr-o altă procedură (publică sau privată).
- 3 funcții, două publice și una privată. Cea privată va fi apelată în cadrul package-ului dintr-o altă funcție sau procedură (publice).
- În cadrul a 2 funcții (sau proceduri) se vor utiliza 4 cursoare (unul implicit, unul explicit cu parametri, iar altul fără, respectiv un cursor multiplu);
- Să aveți o variabilă publică, pe care să o și utilizați.

Realizați și utilizați:

- 2 declanșatori (triggers), pentru diverse operațiuni de tip DML;
- 1 secvență pe care **SĂ O ȘI FOLOSITI** în cadrul unui bloc cu nume.

Am creat package-ul tamplarie care are 2 funcții publice și 2 proceduri publice. Prima procedură se numește *schimbare_salariu* și ia 2 parametri: marca angajatului a căruia îi vom modifica salariul și noul salariu. În interiorul acestei proceduri publice avem și o procedură privată care preia marca angajatului, noul și vechiul salariu și le introduce în tabelul *tamplarie_salary_changes*. De asemenea, în acest tabel, folosim o secvență pentru a auto-incrementa id-ul schimbărilor.

```
BEGIN
    tamplarie.schimbare_salariu(1, 4500);
END;
```

CHANGE_ID	MARCA	OLD_SALARY	NEW_SALARY	CHANGE_DATE
2	1	4600	4500	02-Jan-2025
1	1	4400	4600	02-Jan-2025
7	1	4500	4000	02-Jan-2025
8	1	4000	4000	02-Jan-2025

De asemenea, am adăugat 2 triggeri care transmit câte un mesaj. Primul trigger transmite un mesaj când se modifică date din tabela angajați iar cel de-al doilea când se introduce o înregistrare în tabela cu schimbări salariale

```
Operatiune efectuata cu succes!  
S-a produs o modificare de salariu!  
  
Statement processed.  
  
0.00 seconds
```

Functia *adresa_este* ia ca parapetru codul unic al unui client si returneaza adresa acestuia, alcatuita din mai multe campuri ale tabelului. Aceasta functie publica se foloseste si de o functie privata *display_adresa* pentru a simplifica codul.

```
BEGIN  
    DBMS_OUTPUT.PUT_LINE(tamplarie.adresa_este(111111));  
END;
```

```
Clientul Eleganta Interioarelor are adresa la Strada Armoniei , Nr. 27 , Ap. 2  
  
Statement processed.  
  
0.01 seconds
```

Functia *get_employees_by_tip_agt* are ca parametru tipul de angajat pe care vrem sa il cautam si, folosind un cursor explicit multiplu fara parametrii punem numele si prenumele angajatilor care indeplinesc conditia intr-o variabila pe care o returnam.

```
BEGIN  
    DBMS_OUTPUT.PUT_LINE(tamplarie.get_employees_by_tip_agt('P'));  
END;
```

```
Bogdan Andrei, Dumitrescu Elena, Ardelean Marcel, Paulescu David, Marinescu Alexandru, Boca Ovidiu, Ionascu Maria,  
  
Statement processed.  
  
0.01 seconds
```

Procedura *get_clienti_by_localitate* preia codul localitatii care ne intereseaza si, folosind un cursor cu parametru creaza o variabila cu numele clientilor din localitatea respectiva. Aceasta variabila este ceea ce noi vedem pe ecran la rulara procedurii.

```
BEGIN  
    tamplarie.get_clienti_by_localitate('ZLU');  
END;
```

```
Verde Paradis,  
  
Statement processed.  
  
0.00 seconds
```

CREATE OR REPLACE PACKAGE tamplarie IS

```
gv_strada VARCHAR2(20) := 'Strada ';  
PROCEDURE schimbare_salariu(id_ang IN NUMBER, salariu_nou IN NUMBER);  
FUNCTION adresa_este(cif_client IN NUMBER) RETURN VARCHAR2;  
FUNCTION get_employees_by_tip_agt(tip IN CHAR) RETURN VARCHAR2;  
PROCEDURE get_clienti_by_localitate(lce IN VARCHAR2);
```

END tamplarie;

CREATE OR REPLACE PACKAGE BODY tamplarie IS

```
PROCEDURE changes_log(id_ang IN NUMBER, old_sal IN NUMBER, new_sal IN NUMBER) IS  
BEGIN  
    INSERT INTO tamplarie_salary_changes (change_id, marca, old_salary, new_salary, change_date)  
VALUES (id_creator.nextval, id_ang, old_sal, new_sal, SYSDATE);  
END changes_log;
```

```
FUNCTION display_adresa(cif_client IN NUMBER) RETURN VARCHAR2 IS  
v_adresa VARCHAR2(255);  
v_strada VARCHAR2(30);  
v_nr VARCHAR2(30);  
v_ap VARCHAR2(30);  
BEGIN  
    SELECT strada, TO_CHAR(nr), TO_CHAR(ap) INTO v_strada, v_nr, v_ap FROM tamplarie_clienti  
WHERE cif = cif_client;  
    v_adresa := gv_strada || v_strada || ', Nr. ' || v_nr || ', Ap. ' || v_ap;  
    RETURN v_adresa;  
END display_adresa;
```

```
FUNCTION adresa_este(cif_client IN NUMBER) RETURN VARCHAR2 IS  
v_denumire VARCHAR2(50);  
v_text VARCHAR(255);  
BEGIN  
    SELECT denumire INTO v_denumire FROM tamplarie_clienti WHERE cif = cif_client;  
    v_text := 'Clientul ' || v_denumire || ' are adresa la ' || display_adresa(cif_client);  
    RETURN v_text;  
END adresa_este;
```

```
PROCEDURE schimbare_salariu(id_ang IN NUMBER, salariu_nou IN NUMBER) IS  
old_sal NUMBER(5,0);  
BEGIN  
    SELECT salariu INTO old_sal FROM tamplarie_angajati WHERE marca = id_ang;  
  
    UPDATE tamplarie_angajati  
SET salariu = salariu_nou  
WHERE marca = id_ang;  
  
    changes_log(id_ang, old_sal, salariu_nou);  
END schimbare_salariu;
```

```
FUNCTION get_employees_by_tip_agt(tip IN CHAR) RETURN VARCHAR2 IS  
CURSOR c_angajati IS  
    SELECT nume, prenume FROM tamplarie_angajati WHERE tip_agt = tip;
```

```
v_num tamplarie_angajati.num TYPE;  
v_prenum tamplarie_angajati.prenum TYPE;
```

```
v_names VARCHAR2(4000);
```

```
BEGIN  
  OPEN c_angajati;  
  LOOP  
    FETCH c_angajati INTO v_num, v_prenum;  
    EXIT WHEN c_angajati%NOTFOUND;  
    v_names := v_names || v_num || ' ' || v_prenum || ', '  
  END LOOP;  
  CLOSE c_angajati;  
  RETURN v_names;  
END get_employees_by_tip_agt;
```

```
PROCEDURE get_clienti_by_localitate(lce IN VARCHAR2) IS  
  CURSOR c_client(lce VARCHAR2) IS  
    SELECT denumire FROM tamplarie_clienti WHERE cod_lce = lce;
```

```
v_denumire VARCHAR2(50);  
v_result VARCHAR2(500);  
BEGIN  
  OPEN c_client(lce);  
  LOOP  
    FETCH c_client INTO v_denumire;  
    EXIT WHEN c_client%NOTFOUND;  
    v_result := v_result || v_denumire || ', '  
  END LOOP;  
  CLOSE c_client;  
  DBMS_OUTPUT.PUT_LINE(v_result);  
END get_clienti_by_localitate;
```

```
END tamplarie;
```

```
CREATE OR REPLACE TRIGGER update_message AFTER UPDATE ON tamplarie_angajati  
BEGIN  
  DBMS_OUTPUT.PUT_LINE('Operatiune efectuata cu succes!');  
END;
```

```
CREATE OR REPLACE TRIGGER insert_message AFTER INSERT ON tamplarie_salary_changes  
BEGIN  
  DBMS_OUTPUT.PUT_LINE('S-a produs o modificare de salariu!');  
END;
```

```
CREATE SEQUENCE id_creator START WITH 1 INCREMENT BY 1 MINVALUE 0 MAXVALUE  
20000 CYCLE;
```

MongoDB

În industria mobilei, gestionarea comenzilor și producției este esențială pentru a răspunde cerințelor clienților și pentru a menține eficiența în fabrică. O aplicație care să ajute la urmărirea comenzilor, producției și stocurilor de materiale este crucială pentru acest domeniu.

Aplicația ajută la gestionarea comenzilor de mobilier, urmărind fiecare produs de la comanda până la livrare. Aceasta va stoca informații despre produse, clienți, comenzi, producții, stocuri de materii prime, prețuri și vânzări.

Baza de date va organiza toate informațiile pe colecții, permitând accesul rapid la datele necesare pentru a optimiza producția și vânzările. Aplicația va ajuta conducerea să ia decizii rapide și să eficientizeze procesul de producție.

Am creat colecția *angajati si echipe* în care am inclus informațiile din tabelele *angajati*, *echipe* și *legitimatii*. Colecțiile *echipe* și *legitimatii* sunt imbricate pentru a simplifica structura și pentru a reduce numărul de colecții.

Colecția *produse si productii* îmbină tabelele din titlu cu tabelul *istorice_pret* din aceleași motive pentru care am creat și colecția anterioară. Colecțiile *productii* și *istorice_pret* sunt imbricate. Am procedat la fel și pentru ultima colecție, *clienți si vânzări*, în care colecția *vânzări* este imbricată.

Codul implementării colecțiilor este mai jos:

Colecția *Angajati si Echipe*, cu object id automat:

```
db["Angajati si Echipe"].insertMany([
  {
    _id: ObjectId("67780254815a5bc4ce9a4def"),
    nume: "Bogdan",
    prenume: "Andrei",
    tip_agt: "P",
    data_ang: "2012-01-01",
    salariu: 4400,
    salariu_orar: null,
    nr_tel: null,
    echipe: [
      {
        id: 11,
        denumire: "Mesterii",
        capacitate: 6,
        cod_pds_spec: 2,
        marca_drr: 1
      }
    ],
    legitimatii: [
      {
        id: 100000,
```

```
    data_elib: "2020-03-12",
    data_exp: "2026-03-12"
  }
],
},
{
  _id: ObjectId("67780432815a5bc4ce9a4df0"),
  nume: "Paulescu",
  prenume: "David",
  tip_agt: "P",
  data_ang: "2019-05-01",
  salariu: 3850,
  salariu_orar: null,
  nr_tel: null,
  echipe: [],
  legitimatii: [
    {
      id: null,
      data_elib: null,
      data_exp: null
    }
  ]
},
{
  _id: ObjectId("67780457815a5bc4ce9a4df1"),
  nume: "Pasca",
  prenume: "Andreea",
  tip_agt: "C",
  data_ang: null,
  salariu: null,
  salariu_orar: null,
  nr_tel: 30,
  echipe: [
    {
      id: 33,
      denumire: "Constructorii",
      capacitate: 3,
      cod_pds_spec: 5,
      marca_drr: 3
    }
  ],
  legitimatii: [
    {
      id: 100002,
      data_elib: "2024-01-23",
      data_exp: "2027-01-23"
    }
  ]
},
{
  _id: ObjectId("6778046b815a5bc4ce9a4df2"),
```

```
    nume: "Ardelean",
    prenume: "Marcel",
    tip_agt: "P",
    data_ang: "2010-07-01",
    salariu: 5500,
    salariu_orar: null,
    nr_tel: null,
    echipe: [
      {
        id: 22,
        denumire: "Artistii",
        capacitate: 4,
        cod_pds_spec: 1,
        marca_drr: 3
      }
    ],
    legitimatii: [
      {
        id: 100003,
        data_elib: "2019-01-01",
        data_exp: "2025-01-01"
      }
    ]
  },
  {
    _id: ObjectId("67780478815a5bc4ce9a4df3"),
    nume: "Boca",
    prenume: "Ovidiu",
    tip_agt: "P",
    data_ang: "2021-05-12",
    salariu: 3300,
    salariu_orar: null,
    nr_tel: null,
    echipe: [
      {
        id: 33,
        denumire: "Constructorii",
        capacitate: 3,
        cod_pds_spec: 5,
        marca_drr: 5
      }
    ],
    legitimatii: [
      {
        id: 100004,
        data_elib: "2021-05-12",
        data_exp: "2030-05-12"
      }
    ]
  },
  {
```



```
_id: ObjectId("67780487815a5bc4ce9a4df4"),
nume: "Ardelean",
prenume: "Cornel",
tip_agt: "C",
data_ang: null,
salariu: null,
salariu_orar: null,
nr_tel: 25,
echipe: [
  {
    id: 44,
    denumire: "Preciziei",
    capacitate: 7,
    cod_pds_spec: 4,
    marca_drr: 6
  }
],
legitimatii: [
  {
    id: 100005,
    data_elib: "2020-03-12",
    data_exp: "2026-03-12"
  }
]
},
{
  _id: ObjectId("67780497815a5bc4ce9a4df5"),
  nume: "Popescu",
  prenume: "Andrei",
  tip_agt: "C",
  data_ang: null,
  salariu: null,
  salariu_orar: null,
  nr_tel: 32,
  echipe: [
    {
      id: 66,
      denumire: "Profesionistilor",
      capacitate: 3,
      cod_pds_spec: 3,
      marca_drr: 10
    }
  ],
  legitimatii: [
    {
      id: 100006,
      data_elib: "2020-03-12",
      data_exp: "2026-03-12"
    }
  ]
},
```

```
{
  _id: ObjectId("6778049e815a5bc4ce9a4df6"),
  nume: "Ionascu",
  prenume: "Maria",
  tip_agt: "P",
  data_ang: "2022-05-12",
  salariu: 3630,
  salariu_orar: null,
  nr_tel: null,
  echipe: [
    {
      id: 55,
      denumire: "Innovatorilor",
      capacitate: 3,
      cod_pds_spec: 6,
      marca_drr: 8
    }
  ],
  legitimatii: [
    {
      id: 100007,
      data_elib: "2023-09-17",
      data_exp: "2029-09-17"
    }
  ]
},
{
  _id: ObjectId("677804ac815a5bc4ce9a4df7"),
  nume: "Dumitrescu",
  prenume: "Elena",
  tip_agt: "P",
  data_ang: "2023-03-11",
  salariu: 3190,
  salariu_orar: null,
  nr_tel: null,
  echipe: [],
  legitimatii: [
    {
      id: 100008,
      data_elib: "2024-01-23",
      data_exp: "2027-01-23"
    }
  ]
},
{
  _id: ObjectId("677804b5815a5bc4ce9a4df8"),
  nume: "Marinescu",
  prenume: "Alexandru",
  tip_agt: "P",
  data_ang: "2019-06-01",
  salariu: 3850,
```

```

salariu_orar: null,
nr_tel: null,
echipe: [
  {
    id: 66,
    denumire: "Profesionistilor",
    capacitate: 3,
    cod_pds_spec: 3,
    marca_drr: 10
  }
],
legitimatii: [
  {
    id: 100009,
    data_elib: "2020-01-01",
    data_exp: "2025-01-01"
  }
]
}
]);

```

Colectia *Produse si Productii*, cu id-ul din baza de date anterioara:

```

db["Produse si productii"].insertMany([
  {
    _id: 1,
    nume: 'Scaun',
    tip: 'M',
    cod_lemn: 'MES',
    productii: [
      {
        data: '2023-05-12',
        numar: 5,
        cost: 70,
        id_eca: 11,
        timp_necesar: null
      }
    ],
    istorice_pret: [
      {
        data_start: '2022-12-05',
        data_final: null,
        pret: 100
      }
    ]
  },
  {
    _id: 2,
    nume: 'Scaun',
    tip: 'M',

```

```
cod_lemn: 'ART',
productii: [
  {
    data: '2024-03-11',
    numar: 8,
    cost: 75,
    id_eca: 66,
    timp_necesar: null
  }
],
istorice_pret: [
  {
    data_start: '2020-10-17',
    data_final: null,
    pret: 120
  }
]
},
{
  _id: 3,
  nume: 'Masă',
  tip: 'M',
  cod_lemn: 'STE',
  productii: [
    {
      data: '2024-05-05',
      numar: 3,
      cost: 350,
      id_eca: 33,
      timp_necesar: null
    }
  ],
  istorice_pret: [
    {
      data_start: '2023-12-12',
      data_final: null,
      pret: 500
    }
  ]
},
{
  _id: 4,
  nume: 'Pat',
  tip: 'M',
  cod_lemn: 'NUC',
  productii: [
    {
      data: '2024-02-04',
      numar: 4,
      cost: 450,
      id_eca: 22,
```

```
    timp_necesar: null
  }
],
istorice_pret: [
  {
    data_start: '2024-01-01',
    data_final: null,
    pret: 600
  }
]
},
{
  _id: 5,
  nume: 'Ramă',
  tip: 'D',
  cod_lemn: 'CIR',
  productii: [
    {
      data: '2020-11-12',
      numar: 15,
      cost: 10,
      id_eca: 55,
      timp_necesar: null
    }
  ],
  istorice_pret: [
    {
      data_start: '2018-10-30',
      data_final: null,
      pret: 30
    }
  ]
},
{
  _id: 6,
  nume: 'Ramă',
  tip: 'D',
  cod_lemn: 'PIN',
  productii: [
    {
      data: '2022-11-11',
      numar: 20,
      cost: 20,
      id_eca: 44,
      timp_necesar: null
    }
  ],
  istorice_pret: [
    {
      data_start: '2021-11-15',
      data_final: null,
```

```
        pret: 35
    }
]
}
]);
```

Colectia *Cienti si Vanzari*, tot cu id-uri din baza de date anterioara:

```
db["Cienti si vanzari"].insertMany([
{
    _id: 666666,
    denumire: "Spatii Vii",
    strada: "Decorului",
    nr: 18,
    ap: null,
    cod_postal: 567890,
    cod_lce: "BMA",
    agent: null,
    mail: null,
    vanzari: [
        {
            data: "2024-01-15",
            cantitate: 5,
            cod_pds: 1,
            cif_cet: 666666
        }
    ]
},
{
    _id: 111111,
    denumire: "Verde Paradis",
    strada: "Gradinarilor",
    nr: 14,
    ap: null,
    cod_postal: 456789,
    cod_lce: "ZLU",
    agent: "Ion Costache",
    mail: null,
    vanzari: [
        {
            data: "2024-03-21",
            cantitate: 3,
            cod_pds: 2,
            cif_cet: 111111
        }
    ]
},
{
    _id: 222222,
    denumire: "Eleganta Interioarelor",
    strada: "Armoniei",
```

```
nr: 27,  
ap: 2,  
cod_postal: 987654,  
cod_lce: "CJN",  
agent: null,  
mail: null,  
vanzari: [  
  {  
    data: "2022-11-06",  
    cantitate: 10,  
    cod_pds: 5,  
    cif_cet: 222222  
  }  
]  
},  
{  
  _id: 333333,  
  denumire: "Casa Moderna",  
  strada: "Arhitectilor",  
  nr: 36,  
  ap: null,  
  cod_postal: 345678,  
  cod_lce: "BTA",  
  agent: null,  
  mail: null,  
  vanzari: [  
    {  
      data: "2022-06-15",  
      cantitate: 12,  
      cod_pds: 6,  
      cif_cet: 333333  
    }  
  ]  
},  
{  
  _id: 444444,  
  denumire: "Ambient",  
  strada: "Stilului",  
  nr: 50,  
  ap: 3,  
  cod_postal: 876543,  
  cod_lce: "ORD",  
  agent: null,  
  mail: null,  
  vanzari: [  
    {  
      data: "2024-03-12",  
      cantitate: 3,  
      cod_pds: 4,  
      cif_cet: 444444  
    }  
  ]  
}
```

```

    ]
  },
  {
    _id: 555555,
    denumire: "Confort si Design",
    strada: "Constructorilor",
    nr: 22,
    ap: null,
    cod_postal: 234567,
    cod_lce: "CRI",
    agent: null,
    mail: null,
    vanzari: [
      {
        data: "2023-11-01",
        cantitate: 7,
        cod_pds: 3,
        cif_cet: 555555
      }
    ]
  }
]
});

```

Mai departe am realizat stergerea unuia dintre angajati, dupa nume si prenume:

```
db["Angajati si Echipe"].deleteOne({ nume: "Ionascu", prenume: "Maria" });
```

De asemenea, am actualizat salariul unui angajat, tot dupa nume si prenume:

```
db["Angajati si Echipe"].updateOne(
  { nume: "Bogdan", prenume: "Andrei" },
  { $set: { salariu: 4500 } });
```

Am creat si un index compus pentru a imbunatati performanta interogarilor care include campurile *tip_agt* si *salariu*, in colectia *Angajati si Echipe*.

```
db["Angajati si Echipe"].createIndex({ tip_agt: 1, salariu: 1 });
```


In cele din urma, vom realiza 5 interogari, dintre care 2 vor folosi framework-ul de agregare:

1. Gasiti angajatii care au salariul mai mare de 4500:

```
db["Angajati si Echipe"].find({salariu: {$gt: 4500}});
```

Aceasta comanda returneaza urmatoarele date:

```
{
  _id: ObjectId('6778046b815a5bc4ce9a4df2'),
  nume: 'Ardelean',
  prenume: 'Marcel',
  tip_agt: 'P',
  data_ang: '2010-07-01',
  salariu: 5500,
  salariu_orar: null,
  nr_tel: null,
  echipe: [
    {
      id: 22,
      denumire: 'Artistii',
      capacitate: 4,
      cod_pds_spec: 1,
      marca_drr: 3
    }
  ],
  legitimatii: [
    {
      id: 100003,
      data_elib: '2019-01-01',
      data_exp: '2025-01-01'
    }
  ]
}
```

2. Gasiti clientii care provin din localitatea Zalau:

```
db["Clienti si vanzari"].find({cod_lce: "ZLU"});
```

Aceasta comanda returneaza urmatoarele date:

```
{
  _id: 111111,
  denumire: 'Verde Paradis',
  strada: 'Gradinarilor',
  nr: 14,
  ap: null,
  cod_postal: 456789,
  cod_lce: 'ZLU',
  agent: 'Ion Costache',
  mail: null,
  vanzari: [
    {
      data: '2024-03-21',
      cantitate: 3,
      cod_pds: 2,
      cif_cet: 111111
    }
  ]
}
```

3. Gasiti produsele care sunt confectionate din lemn de nuc:

```
db["Produse si productii"].find({cod_lemn: "NUC"});
```

Aceasta comanda returneaza urmatoarele date:

```
{
  _id: 4,
  nume: 'Pat',
  tip: 'M',
  cod_lemn: 'NUC',
  productii: [
    {
      data: '2024-02-04',
      numar: 4,
      cost: 450,
      id_eca: 22,
      timp_necesar: null
    }
  ],
  istorice_pret: [
    {
      data_start: '2024-01-01',
      data_final: null,
      pret: 600
    }
  ]
}
```

Urmatoarele 2 interogari se vor folosi de framework-ul de agregare:

4. Gasiti media de salariu din fiecare echipa si sortati-le descrescator:

```
db["Angajati si Echipa"].aggregate([
  {
    $unwind: "$echipe"
  },
  {
    $group: {
      _id: "$echipe.denumire",
      salariuMediu: { $avg: "$salariu" },
      numarAngajati: { $sum: 1 }
    }
  }
],
```

```
{
  $sort: { salariuMediu: -1 }
}
]);
```

Aceasta interogare returneaza urmatoarele date:

```
{
  _id: 'Artistii',
  salariuMediu: 5500,
  numarAngajati: 1
}
{
  _id: 'Mesterii',
  salariuMediu: 4500,
  numarAngajati: 1
}
{
  _id: 'Profesionistilor',
  salariuMediu: 3850,
  numarAngajati: 2
}
{
  _id: 'Constructorii',
  salariuMediu: 3300,
  numarAngajati: 2
}
{
  _id: 'Preciziei',
  salariuMediu: null,
  numarAngajati: 1
},
```

5. Gasiti si sortati clientii descrescator, dupa numarul de produse achizitionate:

```
db["Clienti si vanzari"].aggregate([
  {
    $unwind: "$vanzari"
  },
  {
    $group: {
      _id: "$denumire",
      ProduseCumparate: { $sum: "$vanzari.cantitate" }
    }
  }
])
```

```
}  
},  
{  
    $sort: {ProduseCumparate: -1}  
}  
]);
```

Rezultatul acestei interogari este:

```
{  
  _id: 'Casa Moderna',  
  ProduseCumparate: 12  
}  
{  
  _id: 'Eleganta Interioarelor',  
  ProduseCumparate: 10  
}  
{  
  _id: 'Confort si Design',  
  ProduseCumparate: 7  
}  
{  
  _id: 'Spatii Vii',  
  ProduseCumparate: 5  
}  
{  
  _id: 'Verde Paradis',  
  ProduseCumparate: 3  
}  
{  
  _id: 'Ambient',  
  ProduseCumparate: 3  
}
```