



Universitatea Politehnica București
Facultatea de Automatică și Calculatoare
Departamentul de Automatică și Ingineria Sistemelor

Evidenta unei fabrici de imbracaminte

Student
Cruțeru Raluca-Elena, 331AA

București, 2024

Cuprins

1	Definirea Tabelelor	1
2	Relatiile dintre tabele	3
2.1	Interogari Simple	4
2.2	Interogari Complexe	6

1 Definirea Tabelelor

Fabrica

- Fabrica_ID
- NumeFabrica
- Detalii de contact (NumeContact, Oras, Tara, Adresa, CodPostal)

TipProdus

- TipProdus_ID
- NumeTipProdus
- Detalii despre material (vezi Material)

Categorie

- Categorie_ID
- NumeCategorie

Produs

- Produs_ID
- TipProdus_ID (cheie externă către TipProdus)
- Detalii despre producție (DataOraFabricare, Defectiuni)

MagazineColab

- MagazinColab_ID
- Detalii despre magazin (NumeMagazin, Oras, Tara, Adresa, CodPostal)

Materiale

- Material_ID
- Furnizor_ID (cheie externă către Furnizor)
- NumeMaterial
- PretUnitate

Furnizor

- Furnizor_ID
- NumeFurnizor
- Detalii despre furnizor (Oras, Tara, Adresa, CodPostal)

Fabrica2TipProdus, TipProdus2Material, Fabrica2MagazinColab

- Relații de legătură pentru relațiile many-to-many între Fabrica și TipProdus, TipProdus și Material, Fabrica și MagazineColab, respectiv.

Explicații

O fabrică produce mai multe tipuri de produse (bluze, pantaloni, pulovere). Fiecare categorie de produs conține mai multe tipuri de produs: bluza cu mânecă scurtă, lungă, gri etc. Fiecare produs fabricat este de un tip anume de produs, dar individual este vândut unui singur magazin și poate fi cumpărat dintr-un singur magazin. Fiecare tip de produs este confecționat din același material ca toate produsele din acel tip de produs. Un material poate fi folosit la mai multe tipuri de produs, dar și mai multe produse pot fi fabricate din același material. Mai multe fabrici pot să confecționeze un tip de produs și mai multe tipuri de produse pot fi confecționate de o fabrică. Fiecare material este furnizat de o singură firmă. Fiecare fabrică are propria colaborare cu un magazin, însă un magazin poate să colaboreze cu mai multe fabrici și o fabrică cu mai multe magazine.

2 Relatiile dintre tabele

Relații One to Many

- Produs - TipProdus
- Categorie - Tip Produs
- Furnizor - Material
- Produs - MagazineColab

Relații Many to Many

- Fabrica - TipProdus
- TipProdus - Material
- Fabrica - MagazineColab

Diagrama bazei de date

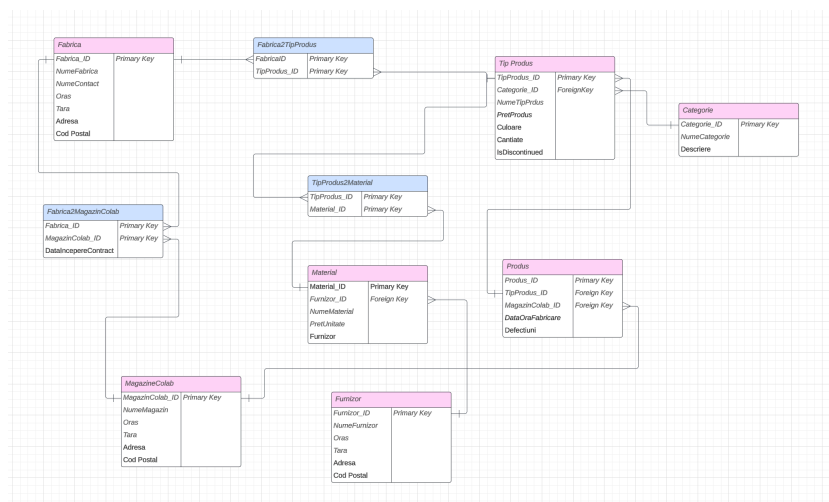


Figura 2.1: diagrama

2.1 Interogari Simple

1 - să furnizeze informații despre tipurile de produse asociate cu o anumită fabrică identificată prin "factory_id"

2 - să se afișeze numele și orașul magazinelor care colaborează cu o anumită fabrică, împreună cu datele contractuale asociate colaborării, unde fabrica are un anumit ID specificat

3 - să furnizeze numele și identificatorul unic (Material_ID) ale materialelor care sunt asociate cu un anumit tip de produs (ProductType_ID) și au un preț unitar (UnityPrice) mai mare de 100

4 - să furnizeze numele categoriei ("CategoryName") asociate cu un anumit tip de produs ("ProductType_ID")

5 - să furnizeze numărul total de produse cu defect ("Defect" este true) pentru un anumit tip de produs ("ProductType_ID").

6 - să furnizeze informații despre furnizorul unui anumit material identificat prin "Material_ID"

```
cur.execute('SELECT "NameProductType", "ProductPrice", "Color", "Quantity", "IsDiscontinued", PT."ProductType_ID"
FROM "ProductType" AS PT
JOIN "Factory2ProductType" AS FP ON PT."ProductType_ID" = FP."ProductType_ID"
WHERE FP."factory_id" = %s', (2,))
```

Figura 2.2: Interogare simpla 1

```
cur.execute('SELECT C."CategoryName"
FROM "Category" AS C
JOIN "ProductType" AS PT ON PT."Categorie_ID" = C."Category_ID"
WHERE PT."ProductType_ID" = %s', (productType_id,))
```

Figura 2.3: Interogare simpla 2

```
search_query = request.args.get('search', default='', type=str)

cur.execute('SELECT M."NameMaterial", M."Material_ID"
FROM "Material" AS M
JOIN "ProductType2Material" AS PT2M ON M."Material_ID" = PT2M."Material_ID"
JOIN "ProductType" AS PT ON PT."ProductType_ID" = PT2M."ProductType_ID"
WHERE PT."ProductType_ID" = %s AND M."UnityPrice" > 100', (productType_id,))

data = cur.fetchall()
```

Figura 2.4: Interogare simpla 3

```
cur.execute('SELECT CS."StoreName", CS."City", F2CS."ContractDate"
FROM "CoLabStores" AS CS
JOIN "Factory2CoLabStores" AS F2CS ON CS."CoLabStores_ID" = F2CS."CoLabStores_ID"
JOIN "Factory" AS F ON F."factory_id" = F2CS."Factory_ID"
WHERE F."factory_id" = %s', (2,))
```

Figura 2.5: Interogare simpla 4

```
cur.execute('SELECT S."NameSupplier", S."City", S."Country"
FROM "Supplier" AS S
JOIN "Material" AS M ON M."Supplier_ID" = S."Supplier_ID"
WHERE M."Material_ID" = %s', (material_id,))
```

Figura 2.6: Interogare simpla 5

```
cur.execute('SELECT COUNT(P."Defect")
FROM "Product" AS P
JOIN "ProductType" AS PT ON PT."ProductType_ID" = P."ProductType_ID"
WHERE PT."ProductType_ID" = %s
AND P."Defect" = true
GROUP BY PT."ProductType_ID"', (productType_id,))
```

Figura 2.7: Interogare simpla 6

```
cur1.execute("""SELECT
MC."StoreName",
(SELECT COUNT(*)
FROM "Product" P
WHERE P."ColabStores_ID" = CB."ColabStores_ID"
AND P."DateFabrication" > FMC."ContractDate") AS NumarProduceFabricate
FROM
"ColabStores" CB
JOIN
"Factory2ColabStores" FMC ON CB."ColabStores_ID"
= FMC."ColabStores_ID"
JOIN
"Factory" F ON FMC."Factory_ID" = F."factory_id"
JOIN
"ColabStores" MC ON FMC."ColabStores_ID" = MC."ColabStores_ID";
""")
```

Figura 2.8: Interogare complexa 1

```
cur1.execute("""SELECT S."NameSupplier",
(SELECT COUNT(M."Material_ID")
FROM "Material" M
JOIN "ProductType2Material" T2M ON M."Material_ID" = T2M."Material_ID"
JOIN "ProductType" TP ON T2M."ProductType_ID" = TP."ProductType_ID"
WHERE TP."ProductType_ID" = %s
AND M."Supplier_ID" = S."Supplier_ID"
) AS NumarMateriale
FROM "Supplier" S
ORDER BY NumarMateriale DESC
LIMIT 1;""", (productType_id,))
```

Figura 2.9: Interogare complexa 2

2.2 Interogari Complexe

1c - să afișeze numele magazinelor (reprezentat prin "StoreName") împreună cu numărul de produse fabricate pentru fiecare magazin în contextul colaborării cu o fabrică specificată

2c - să furnizeze informații despre magazinele colaboratoare ("ColabStores") împreună cu numărul de produse fabricate în fiecare magazin după data semnării contractului cu fabrica asociată ("Factory")

3c - să furnizeze informații despre magazinele colaboratoare ("ColabStores") care au cel mai mare număr de produse de un anumit tip ("ProductType")

4c- furnizează informații despre tipurile de produse, inclusiv numele, prețul de vânzare, costul total al materialelor și profitul obținut pentru fiecare tip de produs

```
cur1 = conn.cursor()
cur.execute("""SELECT CS."StoreName", CS."Address"
FROM "ColabStores" AS CS
JOIN (
SELECT "ColabStores_ID"
FROM "Product"
WHERE "ProductType_ID" = %s
GROUP BY "ColabStores_ID"
ORDER BY COUNT(*) DESC
LIMIT 1
) subquery ON CS."ColabStores_ID" = subquery."ColabStores_ID";""", (productType_id,))
```

Figura 2.10: Interogare complexa 3


```
cur.execute('''SELECT
PT."NameProductType",
PT."ProductPrice",
(SELECT SUM(M."UnityPrice")
FROM "Material" M
JOIN "ProductType2Material" T2M ON M."Material_ID" = T2M."Material_ID"
WHERE T2M."ProductType_ID" = PT."ProductType_ID") AS CostMateriale,
(PT."ProductPrice" - (SELECT SUM(M."UnityPrice")
FROM "Material" M
JOIN "ProductType2Material" T2M ON M."Material_ID" = T2M."Material_ID"
WHERE T2M."ProductType_ID" = PT."ProductType_ID" )) AS Profit
FROM "ProductType" PT; ''')
sb3_data = cur.fetchall()
```

Figura 2.11: Interogare complexa 4