

**СОФИЙСКИ УНИВЕРСИТЕТ „СВ.КЛИМЕНТ ОХРИДСКИ“**

*Факултет по математика и информатика*

**КУРСОВ ПРОЕКТ**

*по Бази от данни на тема*

***Склад***

**Изготвен от:** Димана Вятрова - №62516

Моника Митрева - №62522

Радослав Велков - №62528

Владимир Радев - №62530

|  |  |
| --- | --- |
| **Задача** | **Фак. №** |
| 1. Създаване на схемите и релациите | 62530 |
| 2. Добавяне на съдържание | 62530 |
| 3. Примерни прости заявки и заявки върху две и повече релации | 62522 |
| 4. Примери с подзаявки | 62522 |
| 5. Примери със съединения | 62516 |
| 6. Примери с групиране и агрегация | 62516 |
| 7. Примери с ограничения | 62530 |
| 8. Примери с изгледи и индекси | 62528 |
| 9. Примери с тригери | 62528 |
| 10. Подготовка на презентацията | 62522 |

**Проверен от:** ас. Веселин Илиев

*летен семестър 2021/2022*

*гр. София*

**Съдържание**

1. Описание на предметната област и на заданието ………………….. 3

2. Схема на релациите ………………………………………………………………. 4

3. Добавяне на примерно съдържание за доказване

работоспособността на системата чрез примери ………………….. 9

4. Примерни заявки и техните резултати …………………………………. 12

4.1. Прости заявки (5 бр.) ……………………………………………………. 12

4.2. Заявки върху две и повече релации (5 бр.) ………………….. 15

4.3. Подзаявки (5 бр.) …………………………………………………………. 18

4.4. Съединения (5 бр.) ………………………………………………………. 21

4.5. Групиране и агрегация (10 бр.)…………………………………….. 24

5. Доразвиване на системата с добавяне на изгледи –

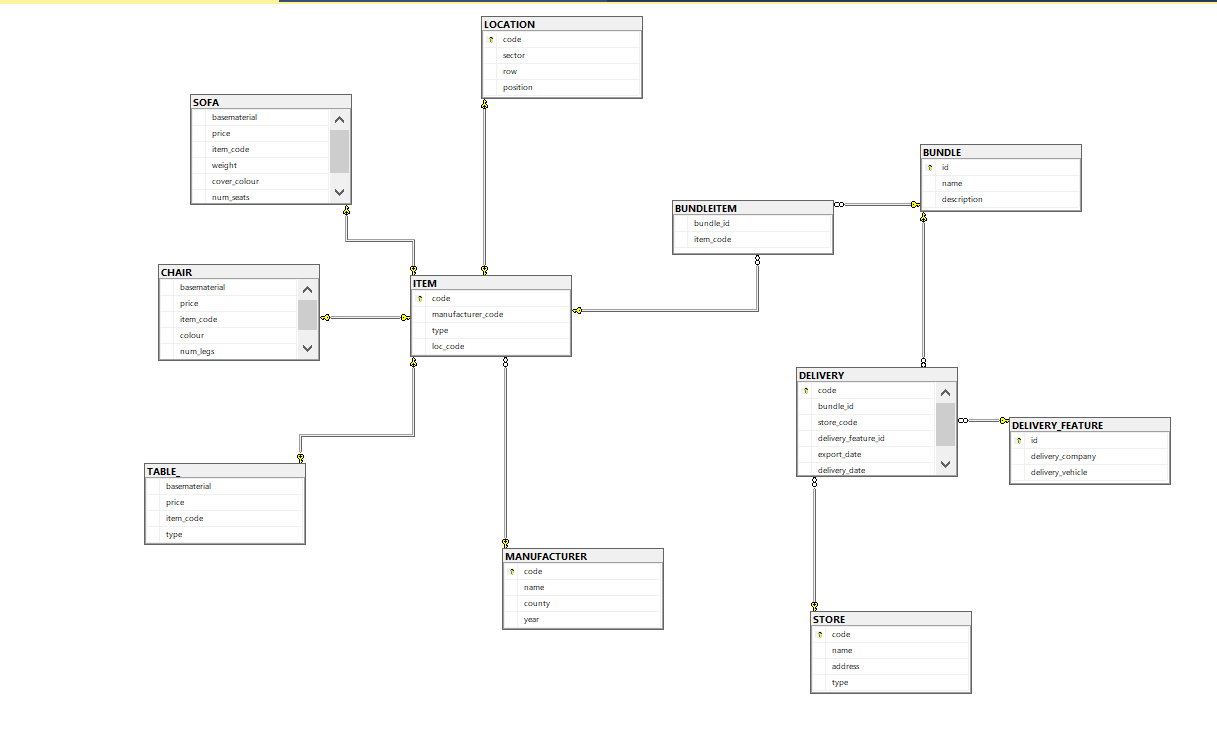
индекси и тригери………………………………………………………………… 29

***1. ОПИСАНИЕ НА ПРЕДМЕТНАТА ОБЛАСТ И НА ЗАДАНИЕТО***

Целта на създаденият от екипът ни проект е да покажем една съвсем базова интерпретация на склад. Подобни бази от данни намират широко приложение в работата с големи вериги магазини от всякакъв тип, на които е нужен бърз и синтезиран достъп до доставените и наличните артикули, предлагани от тях.

В създадената от нас база от данни се съхранява информация за:

* производителите на определени типове артикули (таблицата MANUFACTURER);
* съхраняваните продукти в склада (таблицата ITEM) – тук чрез трите допълнителни таблици SOFA, CHAIR, TABLE сме показали взаимодействието между нашият склад и мебелни магазини - ние пазим данните за предлаганите от тях мебели;
* тяхното точно местоположение (таблицата LOCATION) – основното предназначение на релацията и нейните атрибути е бързи достъп и справка за наличността на определен продукт;
* веригите магазини, с които работим (таблицата STORE);
* извършените от нас доставки (таблицата DELIVERY) – релацията е предназначена да контролира процеса на доставяне, съпровождан от дата на изпращане на пратката от склада ни и дата на получаване, като по-този начин може да се осигури бърза и своевременна реакция при установени нарушения или разминавания;
* куриерските фирми, с които работим, за да извършваме доставките до нашите търговски партньори (таблицата DELIVERY\_FEATURE);
* наличните пакети от продукти (таблицата BUNDLE) – чрез тази релация искаме да покажем „вътрешно разпределение“ за самия склад. Всеки един пакет е като отделно звено, съхраняващо точно определен тип артикули (таблицата BUNDLEITEM). По този начин по-лесно се осъществява достъпа и търсенето на наличните артикули.

******

***2. СХЕМА НА РЕЛАЦИИТЕ***

USE master

GO

if exists (select \* from sysdatabases where name = 'warehouse')

DROP DATABASE warehouse

GO

CREATE DATABASE warehouse

GO

USE warehouse

GO

----- Create Tables -----

CREATE TABLE MANUFACTURER (

code CHAR(10) NOT NULL CONSTRAINT pk\_manufacturer PRIMARY KEY,

name VARCHAR(50) NOT NULL,

county VARCHAR(50) NOT NULL,

year CHAR(4) NOT NULL

);

CREATE TABLE LOCATION (

code VARCHAR(20) NOT NULL CONSTRAINT pk\_location\_code PRIMARY KEY,

sector CHAR(1) NOT NULL,

row INT NOT NULL,

position INT NOT NULL,

CONSTRAINT pk\_location\_combination UNIQUE(sector, row, position),

CHECK (code like 'S%R%P%')

);

CREATE TABLE ITEM (

code CHAR(10) NOT NULL CONSTRAINT pk\_item PRIMARY KEY,

manufacturer\_code CHAR(10) NOT NULL CONSTRAINT fk\_item\_to\_manufacturer FOREIGN KEY

REFERENCES MANUFACTURER(code) ,

type VARCHAR(50) NOT NULL,

loc\_code VARCHAR(20) NOT NULL UNIQUE CONSTRAINT fk\_item\_to\_location FOREIGN KEY

REFERENCES LOCATION(code),

CHECK (type in ('Sofa', 'Chair', 'Table'))

);

CREATE TABLE SOFA (

basematerial VARCHAR(50) NOT NULL,

price DECIMAL(10,2) NOT NULL,

item\_code CHAR(10) NOT NULL UNIQUE CONSTRAINT fk\_sofa\_to\_item FOREIGN KEY REFERENCES ITEM(code),

weight DECIMAL(10,2) NOT NULL,

cover\_colour VARCHAR(50),

num\_seats INT NOT NULL

CHECK (num\_seats >= 1)

);

CREATE TABLE CHAIR (

basematerial VARCHAR(50) NOT NULL,

price DECIMAL(10, 2) NOT NULL,

item\_code CHAR(10) NOT NULL UNIQUE CONSTRAINT fk\_chair\_to\_item FOREIGN KEY REFERENCES ITEM(code),

colour VARCHAR(50) NOT NULL,

num\_legs INT

CHECK (num\_legs in (3, 4) OR num\_legs is null)

);

CREATE TABLE TABLE\_(

basematerial VARCHAR(50) NOT NULL,

price DECIMAL(10,2) NOT NULL,

item\_code CHAR(10) NOT NULL UNIQUE CONSTRAINT fk\_table\_to\_item FOREIGN KEY REFERENCES ITEM(code),

type VARCHAR(11) NOT NULL

CHECK (type IN ('extendable', 'dining', 'oval'))

);

CREATE TABLE BUNDLE (

id INT NOT NULL CONSTRAINT pk\_bundle PRIMARY KEY,

name VARCHAR(20) NOT NULL,

description VARCHAR(100)

);

CREATE TABLE BUNDLEITEM (

bundle\_id INT NOT NULL CONSTRAINT fk\_bundleitem\_to\_bundle FOREIGN KEY

REFERENCES BUNDLE(id),

item\_code CHAR(10) NOT NULL CONSTRAINT fk\_bundleitem\_to\_item FOREIGN KEY

REFERENCES ITEM(code)

);

CREATE TABLE STORE (

code VARCHAR(20) NOT NULL CONSTRAINT pk\_store PRIMARY KEY,

name VARCHAR(50) NOT NULL,

address VARCHAR(50) NOT NULL,

type VARCHAR(20) NOT NULL

CHECK (type IN ('department store', 'furniture store', 'cake shop'))

);

CREATE TABLE DELIVERY\_FEATURE (

id INT NOT NULL CONSTRAINT pk\_delivery\_feature PRIMARY KEY,

delivery\_company VARCHAR(20) NOT NULL,

delivery\_vehicle VARCHAR(20) NOT NULL,

CHECK (delivery\_company IN ('DHL', 'EKONT', 'SPEEDY') AND delivery\_vehicle IN ('Van', 'Mini Truck', 'Truck')),

CONSTRAINT uq\_delivery\_feature\_combination UNIQUE(delivery\_company,delivery\_vehicle)

);

CREATE TABLE DELIVERY (

code CHAR(10) NOT NULL CONSTRAINT pk\_delivery PRIMARY KEY,

bundle\_id INT NOT NULL CONSTRAINT fk\_delivery\_to\_bundle FOREIGN KEY

REFERENCES BUNDLE(id),

store\_code VARCHAR(20) NOT NULL,

delivery\_feature\_id INT NOT NULL CONSTRAINT fk\_delivery\_to\_delivery\_feature FOREIGN KEY

REFERENCES DELIVERY\_FEATURE(id),

export\_date DATE NOT NULL,

delivery\_date DATE NOT NULL,

CONSTRAINT fk\_delivery\_to\_store FOREIGN KEY(store\_code)

REFERENCES STORE(code),

CHECK (export\_date <= delivery\_date)

);

***Забележка:*** *Подобренията с добавяне на ограничения на ниво таблица са*

*въведени при създаването на релациите и въвеждането на стойности.*

Таблицата ***Manufacturer*** съдържа информация за производители. Атрибутът *code* образува първичен ключ.

code - код на производителя;

name - име на производителя;

country - име на държавата производител;

year - година на създаване на производителя;

Таблицата ***Location*** съдържа информация за мястото, на което се намира продуктът. Атрибутът *code* и уникалната тройка *sector, row, position* образуват първични ключове.

code - уникален код на локацията, отговарящ на шаблона 'S%R%P%', където на първо място стои

секторът, на второ - редът, а на трето – позицията, на които е разположен даден артикул;

sector - единична буква отговаряща за секторът;

row - номер на ред;

position - номер на позиция;

Таблицата ***Item*** съдържа информация за продукт. Атрибутът *code* образуват първичен ключ, а атрибутите *manufacturer\_code* и *loc\_code* образуват външен ключ съответно към таблиците *Manufacturer* и *Location.*

code - уникален код на продукта;

manufacturer\_code - код на производителя на този продукт;

type - един от трите типа 'Sofa' | 'Chair' | 'Table';

loc\_code - код на локацията, където се намира продукта в склада;

Таблицата ***Sofa*** съдържа информация за продукт от тип Sofa. Атрибутът *item\_code* образува уникален външен ключ към таблицата *Item.*

basematerial – материалът, от който е направен диванът;

price - цена на диванът;

item\_code - 10 символен код на продукта;

cover\_colour - външен цвят на дивана;

num\_seats - колко местен е диванът;

Таблицата ***Chair*** съдържа информация за продукт от тип Chair. Атрибутът *item\_code* образува уникален външен ключ към таблицата *Item.*

basematerial – материалът, от който е направен столът;

price - цена на столът;

item\_code - 10 символен код на продукта;

colour - външен цвят на столът;

num\_legs - колко крака има столът (3 или 4);

Таблицата ***Table\_*** съдържа информация за продукт от тип Table. Атрибутът *item\_code* образува уникален външен ключ към таблицата *Item.*

basematerial – материалът, от който е направена масата;

price - цена на масата;

item\_code - 10 символен код на продукта;

type - тип на масата ('extendable', 'dining', 'oval');

Таблицата ***Bundle*** съдържа информация за пакет от много продукти, който се използва при извършване на поръчка. Атрибутът *id* образува първичен ключ.

id - цяло число, указващо идентификационен номер на пакет;

name - име на пакет;

description – описание, придружаващо този пакет (не е задължително (null));

Таблицата ***BundleItem*** съдържа информация за id на bundle и съответно code на продукт, който е в него. Атрибутите *bundle\_id* и *item\_code* образуват външен ключ съответно към таблиците *Bundle* и *Item*.

bundle\_id - цяло число, номер на пакет;

item\_code - 10 символен низ, уникален код на продукт;

Таблицата ***STORE*** съдържа информация за клиентските магазини, с които складът извършва доставка на пакети от продукти. Атрибутът *code* образува първичен ключ.

code - 20 символен низ, указващ идентификационен код на магазина;

name - име на магазина;

address - адрес на магазина;

type - тип на магазина (3 типа магазини: 'department store', 'furniture store', 'cake shop');

Таблицата ***DELIVERY\_FEATURE*** съдържа информация за начинът, по който дадена пратка ще бъде доставена до адрес на магазин. Атрибутите *id* и уникалната двойка *delivery\_company, delivery\_vehicle* заедно формират

първичен ключ.

id - цяло число;

delivery\_company - една от три възможни компании за доставка ('DHL', 'ECONT', 'SPEEDY');

delivery\_vehicle - едно от три възможни превозни средства ('Van', 'Mini Truck', 'Truck');

Таблицата ***DELIVERY*** съдържа информация за доставки на пакети от продукти до магазини. Атрибутът *code* образува първичен ключ, а атрибутите *bundle\_id, store\_code , delivery\_feature\_id* образуват външни ключове съответно към таблиците *BUNDLE, STORE, DELIVERY\_FEATURE.*

code - 10 символен низ, уникален за доставка;

bundle\_id - цяло число, отговарящо на номер на пакет, в който са продуктите;

store\_code - 20 символен низ, уникален за магазина, за който е изпратена тази пратка;

delivery\_feature\_id - цяло число, отговарящо за начинът, по който е изпратена пратката и с коя компания;

export\_date - дата на изпращане на пратката;

delivery\_date - дата на доставка на пратката в магазина;

***3. ДОБАВЯНЕ НА ПРИМЕРНО СЪДЪРЖАНИЕ ЗА ДОКАЗВАНЕ РАБОТОСПОСОБНОСТТА НА СИСТЕМАТА ЧРЕЗ ПРИМЕРИ***

INSERT INTO MANUFACTURER

VALUES('6db4od3p68','Qvor','Bulgaria','1992'),

('x5hbimdoph','Zora','Bulgaria','2001'),

('mhozm5dw0d','Koritarov','Bulgaria','1980'),

('qnpmyzn1ng','Kedur','Bulgaria','1998'),

('23yw7nc7j5','Progres','Bulgaria','2008'),

('e93izv7om3','Green Furniture Concept','Sweden','1978'),

('7w5cd2rivc','Noremax','Norway','2019'),

('4ixnhov07f','Wood factory Germany','Germany','2008'),

('9y02x1k9jb','AviNordic','Norway','2007'),

('nfrkgziiwn','P&M furniture','Romania','1989')

INSERT INTO LOCATION

VALUES ('SAR22P1','A',22,1),

('SAR22P12','A',22,12),

('SBR3P5','B',3,5),

('SCR3P1','C',3,1),

('SBR3P4','B',3,4),

('SDR42P9','D',42,9),

('SZR6P1','Z',6,1),

('SWR5P2','W',5,2),

('SWR2P2','W',2,2),

('SBR3P9','B',3,9),

('SCR3P21','C',3,21),

('SBR13P4','B',13,4),

('SDR2P9','D',2,9),

('SZR56P1','Z',56,1),

('SWR3P2','W',3,2),

('SWR6P1','W',6,1),

('SRR4P2','R',4,2)

INSERT INTO ITEM

VALUES ('7z4u1wxWDM','23yw7nc7j5','Sofa','SAR22P1'),

('vzN9UJ27Rx','23yw7nc7j5','Sofa','SAR22P12'),

('I80VAvwP4W','x5hbimdoph','Chair','SBR3P5'),

('Frgk1OlOrE','6db4od3p68','Sofa','SCR3P1'),

('EAavlpQpF6','e93izv7om3','Chair','SBR3P4'),

('6mC5Hvwmo6','x5hbimdoph','Chair','SDR42P9'),

('vEVHOYogtZ','6db4od3p68','Sofa','SZR6P1'),

('J2CB4qX7by','9y02x1k9jb','Chair','SWR5P2'),

('PPi4acu83n','23yw7nc7j5','Chair','SWR2P2'),

('BaQTZkmAWf','9y02x1k9jb','Chair','SBR3P9'),

('fSdg9dEzaX','mhozm5dw0d','Table','SCR3P21'),

('ZvCfjJlha2','9y02x1k9jb','Sofa','SBR13P4'),

('92oI7HReFW','x5hbimdoph','Table','SDR2P9'),

('CzfhY1iZJo','e93izv7om3','Sofa','SZR56P1'),

('pXM84Qs0UZ','mhozm5dw0d','Table','SWR3P2')

INSERT INTO SOFA

VALUES ('Leather',239.99,'7z4u1wxWDM',29.5,'Lemon',2),

('Leather',122,'CzfhY1iZJo',15,'Forest',1),

('Polyester',99.99,'Frgk1OlOrE',14.5,'Brown',1),

('Linen',659.99,'vEVHOYogtZ',39.8,'Gold',6),

('Woolen',445.99,'vzN9UJ27Rx',25.2,'Red',5),

('Polyester',139,'ZvCfjJlha2',19,'Lemon',4)

INSERT INTO CHAIR

VALUES ('Leather',256.99,'6mC5Hvwmo6','Gold',4),

('Vinyl',45,'BaQTZkmAWf','Red',4),

('Fabric',69.99,'EAavlpQpF6','Yellow',3),

('Fabric',89.99,'I80VAvwP4W','Grey',3),

('Mesh',300,'J2CB4qX7by','Red',4),

('Mesh',249,'PPi4acu83n','Black',4)

INSERT INTO TABLE\_

VALUES ('Solid wood',240,'92oI7HReFW','dining'),

('Wood',56,'fSdg9dEzaX','extendable'),

('Butcher block',69.99,'pXM84Qs0UZ','oval')

INSERT INTO BUNDLE(id,name)

VALUES (1,'bulgarian\_chairs'),

(4,'sweden\_sofas'),

(7,'norway\_sofas'),

(8,'bulgarian\_tables')

INSERT INTO BUNDLE

VALUES (2,'norway\_bundle','It contains all types of furniture created in Norway'),

(3, 'three\_legged\_chairs','All three legged chairs'),

(5,'bulgarian\_bundle','It contains all types of furniture created in Bulgaria'),

(6,'sweden\_bundle','It contains all types of furniture created in Sweden'),

(9,'all\_in','It contains all furniture stored in the warehouse')

INSERT INTO BUNDLEITEM

VALUES (1,'6mC5Hvwmo6'),

(1,'BaQTZkmAWf'),

(1,'EAavlpQpF6'),

(1,'I80VAvwP4W'),

(1,'J2CB4qX7by'),

(1,'PPi4acu83n'),

(2,'BaQTZkmAWf'),

(2,'J2CB4qX7by'),

(2,'ZvCfjJlha2'),

(3,'EAavlpQpF6'),

(3,'I80VAvwP4W'),

(4,'CzfhY1iZJo'),

(5,'fSdg9dEzaX'),

(5,'pXM84Qs0UZ'),

(5,'7z4u1wxWDM'),

(5,'PPi4acu83n'),

(5,'vzN9UJ27Rx'),

(5,'Frgk1OlOrE'),

(5,'vEVHOYogtZ'),

(5,'6mC5Hvwmo6'),

(5,'92oI7HReFW'),

(5,'I80VAvwP4W'),

(6,'CzfhY1iZJo'),

(6,'EAavlpQpF6'),

(8,'fSdg9dEzaX'),

(8,'pXM84Qs0UZ'),

(8,'92oI7HReFW')

INSERT INTO STORE

VALUES ('VID\_EOOD','Mebeli Videnov','Sofia ul.stradjanka 57','furniture store'),

('ZOR\_ED','Zora','Sofia ul.Elisaveta Bagrqna','furniture store'),

('MOS\_AD','Mall of sofia','Sofia al.stamboliyski','department store'),

('NED\_OOD','Sladkarnica Nedelq','Sofia manastirski livadi - zapad','cake shop')

INSERT INTO DELIVERY\_FEATURE

VALUES (1,'DHL','VAN'),

(2,'DHL','Truck'),

(3,'SPEEDY','VAN'),

(4,'EKONT','Mini truck')

INSERT INTO DELIVERY

VALUES ('MvlbUA0tO3',3,'NED\_OOD',1,'2022-05-12','2022-05-13'),

('pr9HILbFSL',8,'VID\_EOOD',4,'2022-04-23','2022-05-1'),

('oKhYVT8jJl',5,'VID\_EOOD',2,'2020-09-12','2020-09-25'),

('TQuO36i30j',4,'MOS\_AD',4,'2021-06-26','2022-07-08'),

('c7vUCF0Ms5',6,'ZOR\_ED',3,'2020-01-12','2020-01-13'),

('jg5Wj9Caah',2,'ZOR\_ED',3,'2020-02-12','2022-02-13')

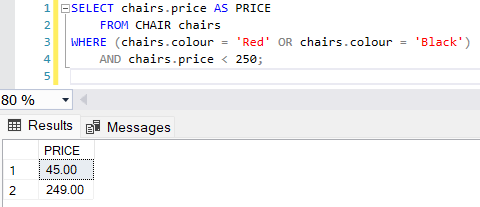
***4. ПРИМЕРНИ ЗАЯВКИ И ТЕХНИТЕ РЕЗУЛТАТИ***

*4.1. Прости заявки*

*Картина, която съдържа текст

Описанието е генерирано автоматично****№1.*** *Да се изведат кодът и цената на всички дивани, които са под 500 лева.*

***№2.*** *Да се изведат цените на всички столове, които са червени или черни на*

*цвят и струват по-малко от 250 лева.*

***№3.*** *Да се изведе информацията за всички мебелни магазини, с които складът*

*Картина, която съдържа текст

Описанието е генерирано автоматичноработи.*

***№4.*** *Да се изведат имената (и годината) на всчики български производители,*

*Картина, която съдържа маса

Описанието е генерирано автоматичноосновани след 1990 година.*

*Картина, която съдържа маса

Описанието е генерирано автоматично****№5.*** *Да се изведе информацията за всички локации, чийто код започва с "SB".*

*4.2. Заявки върху две и повече релации*

***№1****. Да се изведе кодът за доставка на всички пратки, доставени от фирмите ECONT*

*Картина, която съдържа текст

Описанието е генерирано автоматичнои SPEEDY.*

***№2.*** *Да се изведат номерата на пакетите, съдържащи продукти, които са изпратени*

*Картина, която съдържа текст

Описанието е генерирано автоматичноза мебелните магазини, партньори на склада.*

***Картина, която съдържа текст

Описанието е генерирано автоматично№3.*** *Да се изведат името и кодът на производителите на фотьойли/столове.*

***№4.*** *Да се изведат имената на всички магазини, получили доставка от склада преди*

*Картина, която съдържа текст

Описанието е генерирано автоматично15.05.2022 година, както и датата на доставката.*

*Картина, която съдържа текст

Описанието е генерирано автоматично****№5.*** *Да се изведат цените на всички дивани, произведени от Progres.*

*4.3. Подзаявки*

***Картина, която съдържа текст

Описанието е генерирано автоматично№1.*** *Да се изведе кодът на производителя на всички кръгли маси.*

***№2.*** *Да се изведат името, държавата и годината на основаване на фирмите,*

*Картина, която съдържа маса

Описанието е генерирано автоматичнопроизводители на столове.*

*Картина, която съдържа текст

Описанието е генерирано автоматично****№3.*** *Да се изведе вида на най-скъпия артикул в склада и неговата цена.*

*Картина, която съдържа текст

Описанието е генерирано автоматично****№4.*** *Да се изведе името на магазина, получил последна доставка от склада.*

*Картина, която съдържа текст

Описанието е генерирано автоматично****№5.*** *Да се изведат номерата на пакетите, в които има кожени двуместни дивани.*

*4.4. Съединения*

***№1.*** *Да се изведе цената на всички продукти и типа продукти, които са*

*Картина, която съдържа текст

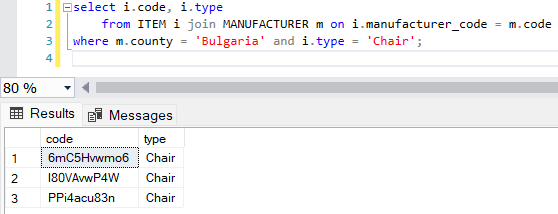
Описанието е генерирано автоматичнопроизведени в България.*

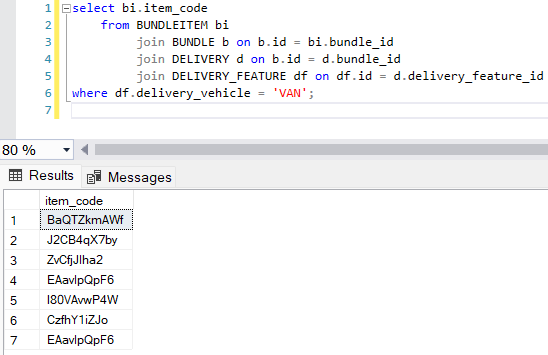
***№2.*** *Да се изведе информацията за всички производители, заедно с всички данни*

*за продуктите, произведени от тях. Ако не са произвели нищо, на стойностите за*

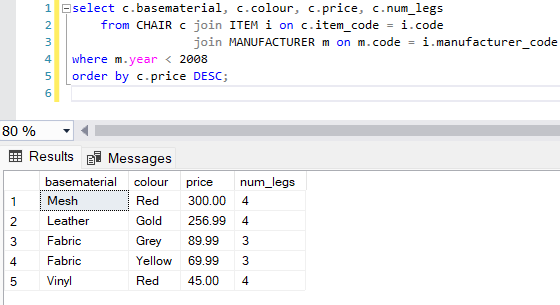
***Картина, която съдържа маса

Описанието е генерирано автоматично****продукт да се извежда NULL.*

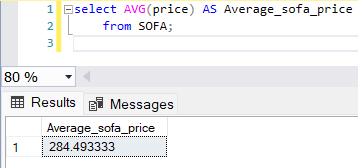
*****№3.*** *Да се изведат всички столове, произведени в България.*

*****№4.*** *Да се изведат кодовете на продуктите, доставени с ван.*

***№5.*** *Да се изведат всички данни за столовете, които са произведени*

*от компания, основана преди 2008 година.*

*4.5. Групиране и агрегация*

*****№1.*** *Да се изведе средната цена на фотьойлите.*

***№2.*** *Да се изведе средната цена на фотьойлите, на база материалът, Картина, която съдържа текст

Описанието е генерирано автоматичноот който*

*са направени.*

***Картина, която съдържа текст

Описанието е генерирано автоматично№3.*** *Да се изведе броят на мебелите спрямо вида им.*

***№4.*** *Да се изведе броят пъти, в които се среща всяко описание, което Картина, която съдържа текст

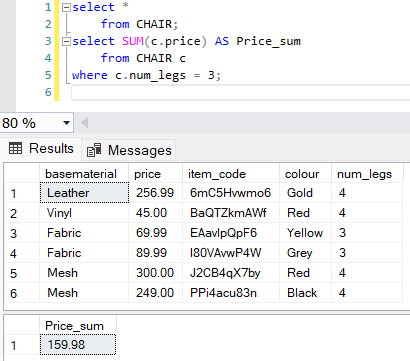
Описанието е генерирано автоматичноне е празно.*

*Картина, която съдържа маса

Описанието е генерирано автоматично****№5.*** *Да се изведат заетите позиции на всеки ред.*

***Картина, която съдържа текст

Описанието е генерирано автоматично№6.*** *Да се изведе средната цена на всички продукти.*

*****№7.*** *Да се намери общата цена на всички столове с 3 крака.*

***№8.*** *Да се изведе идентификационният код на всеки пакет, който съдържа по-малко*

*Картина, която съдържа текст

Описанието е генерирано автоматичноот 5 артикула.*

***№9.*** *Да се изведат секторите, в които има фотьойли, както и броя на фотьойлите*

*Картина, която съдържа текст

Описанието е генерирано автоматично във всеки сектор.*

***№10.*** *Да се изведе максималната цена на столовете, произведени в България*

*спрямо броят крака, които имат.*

*Картина, която съдържа текст

Описанието е генерирано автоматично*

***5. ДОРАЗВИВАНЕ НА СИСТЕМАТА С ДОБАВЯНЕ НА ИЗГЛЕДИ – ИНДЕКСИ И ТРИГЕРИ***

***№1.*** *Изгледи:*

- извеждащ кода на всеки един продукт, както и цената му:

CREATE VIEW v\_allItemsPrices

AS

SELECT item.code, price

FROM item, table\_

WHERE item.code=table\_.item\_code

UNION

SELECT item.code, price

FROM item, sofa

WHERE item.code=sofa.item\_code

UNION

SELECT item.code, price

FROM item, chair

WHERE item.code=chair.item\_code

- извеждащ средната цена за всеки вид продукт:

CREATE VIEW v\_AvgPricesOfProducts

AS

SELECT type, AVG(price) AS avg\_price

FROM item, chair

WHERE item.code=chair.item\_code

GROUP BY type

UNION

SELECT type, AVG(price) AS avg\_price

FROM item, sofa

WHERE item.code=sofa.item\_code

GROUP BY type

UNION

SELECT item.type, AVG(price) AS avg\_price

FROM item, table\_

WHERE item.code=table\_.item\_code

GROUP BY item.type

- извеждащ кодове и материал на всички дивани, струващи над 100 (with check option):

CREATE VIEW v\_sofasAbove100

AS

SELECT item\_code, basematerial, price

FROM sofa

WHERE price>=100

WITH CHECK OPTION;

- извеждащ код, материал и цвят на всички 4-краки стола (with check option):

CREATE VIEW v\_all\_4legged\_Chairs

AS

SELECT item\_code, basematerial, colour

FROM chair

WHERE num\_legs=4

WITH CHECK OPTION;

- извеждащ най-ниска и най-висока цена на стол за всеки вид материал:

CREATE VIEW v\_min\_max\_prices\_allMaterials

AS

SELECT basematerial, MIN(price) AS min\_price, MAX(price) AS max\_price

FROM chair

GROUP BY basematerial

- извеждащ материал и тип на всички дървени маси (името на материала съдържа ‚‚wood‘‘)

(with check option):

CREATE VIEW v\_wooden\_tables

AS

SELECT basematerial, type

FROM table\_

WHERE basematerial LIKE '%wood%'

WITH CHECK OPTION;

- извеждащ всичката информация за всеки български производители (with check option):

CREATE VIEW v\_Bulgarian\_manufacturers

AS

SELECT \*

FROM manufacturer

WHERE county='Bulgaria'

WITH CHECK OPTION;

- извеждащ всички български пакети, които имат въведено описание (with check option):

CREATE VIEW v\_bulgarianBundles\_withDescription

AS

SELECT \*

FROM bundle

WHERE name LIKE '%bulgarian%' AND description IS NOT NULL

WITH CHECK OPTION;

- извеждащ ID на пакет, ID на доставка и дата на доставка за всички пакети, които трябва да бъдат доставени преди 2022 година (with check option):

CREATE VIEW v\_deliveriesBefore2022

AS

SELECT bundle\_id, delivery\_feature\_id, delivery\_date

FROM delivery

WHERE delivery\_date<'2022-01-01'

WITH CHECK OPTION;

- извеждащ името на всеки магазин и колко пакета трябва да му бъдат доставени:

CREATE VIEW v\_allStores\_numberOfBundles

AS

SELECT store.name, COUNT(delivery.bundle\_id) AS num\_bundles

FROM store, delivery

WHERE store.code=delivery.store\_code

GROUP BY store.name

- извеждащ всеки магазин (код, име, вид), като за всеки такъв се извеждат компанията/компаниите, чрез която/които ще се осъществи доставка, както и превозното средство:

CREATE VIEW v\_stores\_deliveries

AS

SELECT DISTINCT s.code, s.name, s.type, df.delivery\_company,

df.delivery\_vehicle

FROM store s, delivery d, delivery\_feature df

WHERE s.code=d.store\_code AND d.delivery\_feature\_id=df.id

***№2.*** *Индекси:*

За таблицата **Item**:

*- обединяващ типа на продукта и кода му:*

CREATE INDEX idx\_type\_code

ON item(type, code);

*- обединяващ типа и кода на продукта и кода на производителя:*

CREATE INDEX idx\_type\_code\_manufacturer

ON item(type, code, manufacturer\_code);

За таблицата **Table\_**:

*- обединяващ типа на масата и цената й:*

CREATE INDEX idx\_type\_price

ON table\_(type, price);

*- обединяващ материала на масата и цената й:*

CREATE INDEX idx\_basematerial\_price

ON table\_(basematerial, price);

*- обединяващ типа и материала на масата:*

CREATE INDEX idx\_type\_basematerial

ON table\_(type, basematerial);

За таблицата **Chair**:

*- обединяващ материала на стола и цената му:*

CREATE INDEX idx\_basematerial\_price

ON chair(basematerial, price);

*- за броя на краката на стола:*

CREATE INDEX idx\_numLegs

ON chair(num\_legs);

За таблицата **Sofa**:

*- обединяващ материала на дивана и цената му:*

CREATE INDEX idx\_basematerial\_price

ON sofa(basematerial, price);

*- за броя на местата за сядане на дивана:*

CREATE INDEX idx\_numSeats

ON sofa(num\_seats);

За таблицата **Manufacturer**:

*- за името на производителя:*

CREATE INDEX idx\_name

ON manufacturer(name);

*- обединяващ името на производителя и държавата, от която е:*

CREATE INDEX idx\_name\_county

ON manufacturer(name, county);

За таблицата **Location**:

*- обединяващ сектор, ред и позиция:*

CREATE INDEX idx\_sector\_row\_position

ON location(sector, row, position);

За таблицата **Bundle**:

*- обединяващ ID и име на пакет:*

CREATE INDEX idx\_id\_name

ON bundle(id,name);

За таблицата **Delivery**:

- *обединяващ ID на пакет и код на магазин:*

CREATE INDEX idx\_bundleID\_storeCode

ON delivery(bundle\_id,store\_code);

За таблицата **Store**:

- *обединяващ име на магазина и адреса му:*

CREATE INDEX idx\_name\_address

ON store(name, address);

- *обединяващ тип на магазина и името му:*

CREATE INDEX idx\_type\_name

ON store(type, name);

- *обединяващ код на магазина и името му:*

CREATE INDEX idx\_code\_name

ON store(code, name);

***№3.*** *Тригери:*

*- Добавя нова колона „num\_items“ към таблицата* ***manufacturer****, която ще съдържа броя на артикулите, които всеки производител е произвел (с check ограничение):*

ALTER TABLE manufacturer ADD num\_items INT CHECK(num\_items>=0);

*- Тригер за таблицата* ***item****, който да се задейства при вмъкване на нов продукт (стол, диван или маса) в таблицата и да увеличава с единица броя на артикулите за таблицата* ***manufacturer****:*

CREATE TRIGGER tr\_increase\_num\_of\_items

ON item

FOR INSERT

AS

UPDATE manufacturer

SET num\_items = isnull(num\_items,0)+(SELECT count(1)

FROM inserted

WHERE manufacturer\_code=manufacturer.code)

WHERE manufacturer.code IN (SELECT manufacturer\_code FROM inserted);

*- Тригер за таблицата* ***item****, който да се задейства при изтриване на продукт (стол, диван или маса) от таблицата и да намалява с единица броя на артикулите за таблицата* ***manufacturer****:*

CREATE TRIGGER tr\_decrease\_num\_of\_items

ON item

FOR DELETE

AS

UPDATE manufacturer

SET num\_items = num\_items - (SELECT count(1)

FROM deleted

WHERE manufacturer\_code=manufacturer.code)

WHERE manufacturer.code IN (SELECT manufacturer\_code FROM deleted);

