**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 продука 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);

**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

**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);