Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет

информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра вычислительных машин систем и сетей

Дисциплина: Базы данных

ОТЧЕТ

по лабораторной работе №3

Студент: гр. 450504

Павец Н. К.

Руководитель: старший преподаватель

Калабухов Е. В.

Минск, 2017

Проект

Интернет-магазин компьютерной техники

Сущности

- City
- Country
- Brand
- Notebook
- Customer
- Image
- Provider

SQL DDM Реляционная диаграмма

city

P	* id	NUMBER (9)
	* title	VARCHAR2 (25!
F	country_id	NUMBER (9)

City PK (id)

country

P	* id	NUMBER (9)
	* title	VARCHAR2 (255 CHAI

Country PK (id)

brand

P	* id	NUMBER (9)
	* title	VARCHAR2 (25!
F	country_id	NUMBER (9)

Brand PK (id)

basket

PF	* customer_id	NUMBER (9)
PF	* notebook_id	NUMBER (9)
	quantity	NUMBER (9)
	discount	NUMBER (9,2)

basket__IDX (customer_id, note

customer

P	* id	NUMBER (9)
	* first_name	VARCHAR2 (25
	last_name	VARCHAR2 (25
	* email	VARCHAR2 (25
	* password	VARCHAR2 (25
F	city_id	NUMBER (9)

Notebook PK (id)

review

PF	* customer_id	NUMBER (9)
PF	* notebook_id	NUMBER (9)
	mark	NUMBER (9)
	review	CLOB

review__IDX (customer_id, note

notebook

P	* id	NUMBER (9)
	* title	VARCHAR2 (25
	description	CLOB
	config	CLOB
	* price	NUMBER (9,2)
F	brand_id	NUMBER (9)
F	provider_id	NUMBER (9)

Notebook PK (id)

image

P	* id	NUMBER (9)
	* link	VARCHAR2 (2
	* type	SMALLINT
F	notebook_id	NUMBER (9)

Image PK (id)

Provider

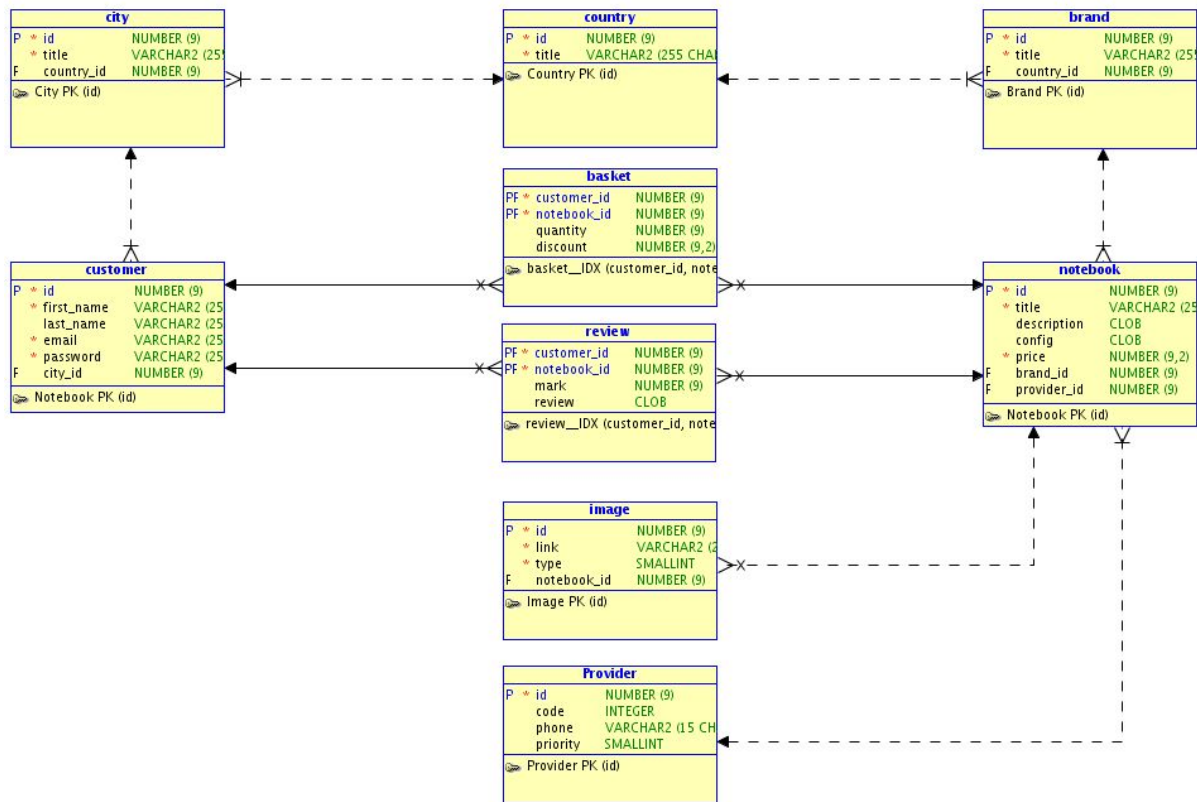|		
P	* id	NUMBER (9)
	code	INTEGER
	phone	VARCHAR2 (15 CH
	priority	SMALLINT

Provider PK (id)

Рисунок 1 - SQL DDM диаграмма в нотации Баркера
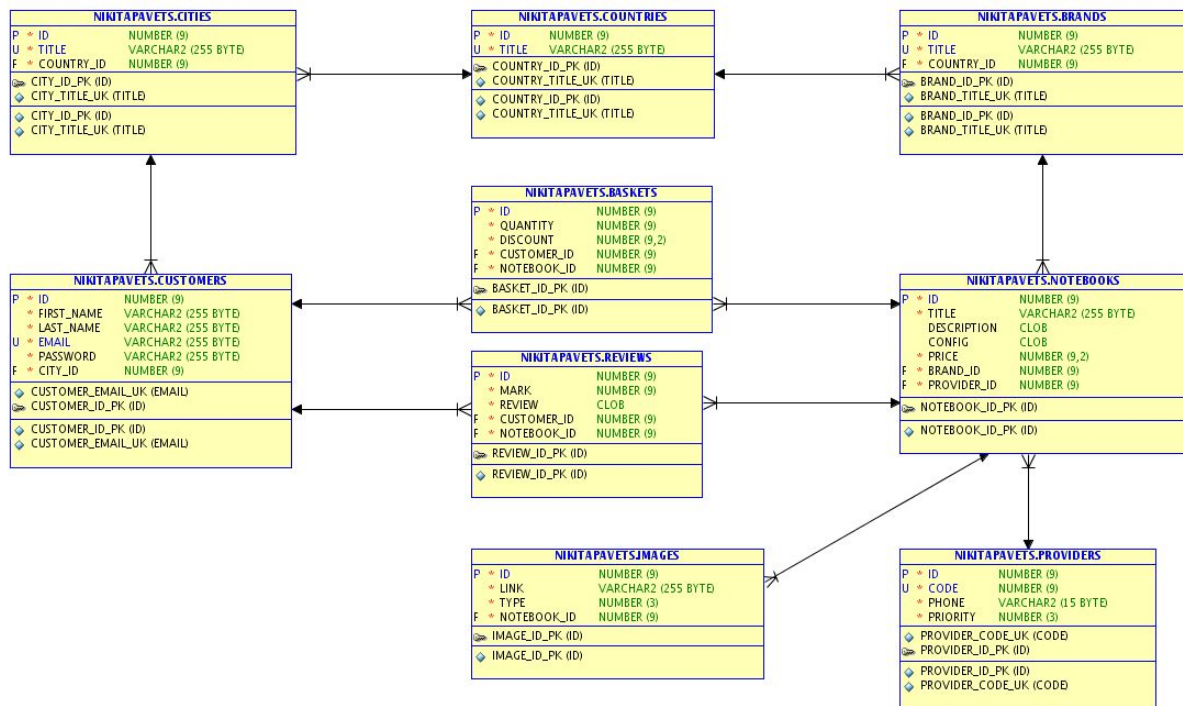
SQL Реляционная диаграмма (сгенерированная)

Рисунок 2 - SQL Реляционная диаграмма (сгенерированная)

В результате сравнения, получились практически идентичные реализации, за исключением того, что, при генерировании через sql-команды, были использованы операторы позволяющие добавить схеме бизнес-логику.

SQL-команды

```
-- COUNTRY ENTITY

ALTER TABLE CITIES
  DROP CONSTRAINT CITY_COUNTRY_FK;

ALTER TABLE BRANDS
  DROP CONSTRAINT BRAND_COUNTRY_FK;

DROP TABLE COUNTRIES;

CREATE TABLE COUNTRIES
(
 ID NUMBER(9) NOT NULL,
 TITLE VARCHAR2(255) NOT NULL,

 CONSTRAINT COUNTRY_TITLE_UK UNIQUE (TITLE),
 CONSTRAINT COUNTRY_ID_PK PRIMARY KEY (ID)
);

-- CITY ENTITY

ALTER TABLE CUSTOMERS
  DROP CONSTRAINT CUSTOMER_CITY_FK;

DROP TABLE CITIES;

CREATE TABLE CITIES
(
 ID NUMBER(9) NOT NULL,
 TITLE VARCHAR2(255) NOT NULL,
 COUNTRY_ID NUMBER(9) NOT NULL,

 CONSTRAINT CITY_TITLE_UK UNIQUE (TITLE),
 CONSTRAINT CITY_ID_PK PRIMARY KEY (ID),
 CONSTRAINT CITY_COUNTRY_FK
  FOREIGN KEY (COUNTRY_ID)
  REFERENCES COUNTRIES
  ON DELETE CASCADE
);
```

```sql
-- BRAND ENTITY

ALTER TABLE NOTEBOOKS
  DROP CONSTRAINT NOTEBOOK_BRAND_FK;

DROP TABLE BRANDS;

CREATE TABLE BRANDS
(
  ID NUMBER(9) NOT NULL,
  TITLE VARCHAR2(255) NOT NULL,
  COUNTRY_ID NUMBER(9) NOT NULL,

  CONSTRAINT BRAND_TITLE_UK UNIQUE (TITLE),
  CONSTRAINT BRAND_ID_PK PRIMARY KEY (ID),
  CONSTRAINT BRAND_COUNTRY_FK
    FOREIGN KEY (COUNTRY_ID)
    REFERENCES COUNTRIES
);

-- CUSTOMER ENTITY

ALTER TABLE BASKETS
  DROP CONSTRAINT BASKET_CUSTOMER_FK;

ALTER TABLE REVIEWS
  DROP CONSTRAINT REVIEW_CUSTOMER_FK;

DROP TABLE CUSTOMERS;

CREATE TABLE CUSTOMERS
(
  ID NUMBER(9) NOT NULL,
  FIRST_NAME VARCHAR2(255) NOT NULL,
  LAST_NAME VARCHAR2(255) NOT NULL,
  EMAIL VARCHAR2(255) NOT NULL,
  PASSWORD VARCHAR2(255) NOT NULL,
  CITY_ID NUMBER(9) NOT NULL,

  CONSTRAINT CUSTOMER_EMAIL_UK UNIQUE (EMAIL),
  CONSTRAINT CUSTOMER_ID_PK PRIMARY KEY (ID),
```

```
    CONSTRAINT CUSTOMER_CITY_FK FOREIGN KEY (CITY_ID) REFERENCES CITIES
);

-- PROVIDER ENTITY

ALTER TABLE NOTEBOOKS
  DROP CONSTRAINT NOTEBOOK_PROVIDER_FK;

DROP TABLE PROVIDERS;

CREATE TABLE PROVIDERS
(
  ID NUMBER(9) NOT NULL,
  CODE NUMBER(9) NOT NULL,
  PHONE VARCHAR2(15) NOT NULL,
  PRIORITY NUMBER(3) NOT NULL,

  CONSTRAINT PROVIDER_CODE_UK UNIQUE (CODE),
  CONSTRAINT PROVIDER_ID_PK PRIMARY KEY (ID)
);

-- NOTEBOOK ENTITY

ALTER TABLE IMAGES
  DROP CONSTRAINT IMAGE_NOTEBOOK_FK;

ALTER TABLE BASKETS
  DROP CONSTRAINT BASKET_NOTEBOOK_FK;

ALTER TABLE REVIEWS
  DROP CONSTRAINT REVIEW_NOTEBOOK_FK;

DROP TABLE NOTEBOOKS;

CREATE TABLE NOTEBOOKS
(
  ID NUMBER(9) NOT NULL,
  TITLE VARCHAR2(255) NOT NULL,
  DESCRIPTION CLOB,
  CONFIG CLOB,
  PRICE NUMBER(9,2) NOT NULL,
  BRAND_ID NUMBER(9) NOT NULL,
```

6

```
  PROVIDER_ID NUMBER(9) NOT NULL,

  CONSTRAINT NOTEBOOK_PRICE_MIN CHECK (PRICE > 0),
  CONSTRAINT NOTEBOOK_ID_PK PRIMARY KEY (ID),
  CONSTRAINT NOTEBOOK_BRAND_FK FOREIGN KEY (BRAND_ID) REFERENCES BRANDS,
  CONSTRAINT NOTEBOOK_PROVIDER_FK FOREIGN KEY (PROVIDER_ID) REFERENCES PROVIDERS
);

-- IMAGE ENTITY

DROP TABLE IMAGES;

CREATE TABLE IMAGES
(
  ID NUMBER(9) NOT NULL,
  LINK VARCHAR2(255) NOT NULL,
  TYPE NUMBER(3) NOT NULL,
  NOTEBOOK_ID NUMBER(9) NOT NULL,

  CONSTRAINT IMAGE_ID_PK PRIMARY KEY (ID),
  CONSTRAINT IMAGE_NOTEBOOK_FK
    FOREIGN KEY (NOTEBOOK_ID)
    REFERENCES NOTEBOOKS
    ON DELETE CASCADE
);

-- BASKET ENTITY

DROP TABLE BASKETS;

CREATE TABLE BASKETS
(
  ID NUMBER(9) NOT NULL,
  QUANTITY NUMBER(9) NOT NULL,
  DISCOUNT NUMBER(9,2) NOT NULL,
  CUSTOMER_ID NUMBER(9) NOT NULL,
  NOTEBOOK_ID NUMBER(9) NOT NULL,

  CONSTRAINT BASKET_QUANTITY_MIN CHECK (QUANTITY > 0),
  CONSTRAINT BASKET_DISCOUNT_MIN CHECK (DISCOUNT > -1),
  CONSTRAINT BASKET_ID_PK PRIMARY KEY (ID),
  CONSTRAINT BASKET_NOTEBOOK_FK
```

```
    FOREIGN KEY (NOTEBOOK_ID)
    REFERENCES NOTEBOOKS
    ON DELETE CASCADE,
  CONSTRAINT BASKET_CUSTOMER_FK
    FOREIGN KEY (CUSTOMER_ID)
    REFERENCES CUSTOMERS
    ON DELETE CASCADE
);

-- REVIEW ENTITY

DROP TABLE REVIEWS;

CREATE TABLE REVIEWS
(
  ID NUMBER(9) NOT NULL,
  MARK NUMBER(9) NOT NULL,
  REVIEW CLOB NOT NULL,
  CUSTOMER_ID NUMBER(9) NOT NULL,
  NOTEBOOK_ID NUMBER(9) NOT NULL,

  CONSTRAINT REVIEW_ID_PK PRIMARY KEY (ID),
  CONSTRAINT REVIEW_NOTEBOOK_FK
    FOREIGN KEY (NOTEBOOK_ID)
    REFERENCES NOTEBOOKS
    ON DELETE CASCADE,
  CONSTRAINT REVIEW_CUSTOMER_FK
    FOREIGN KEY (CUSTOMER_ID)
    REFERENCES CUSTOMERS
    ON DELETE CASCADE
);
```

При заполнении структуры данных, ошибок не возникло, база спроектированная правильно.