

# Создание базы данных(SQL)

На данной странице рассматривается процесс создания Базы данных на основе ER-диаграммы из предыдущей страницы. Также написаны простые запросы для анализа

Автор: Шамсутдинов Вадим

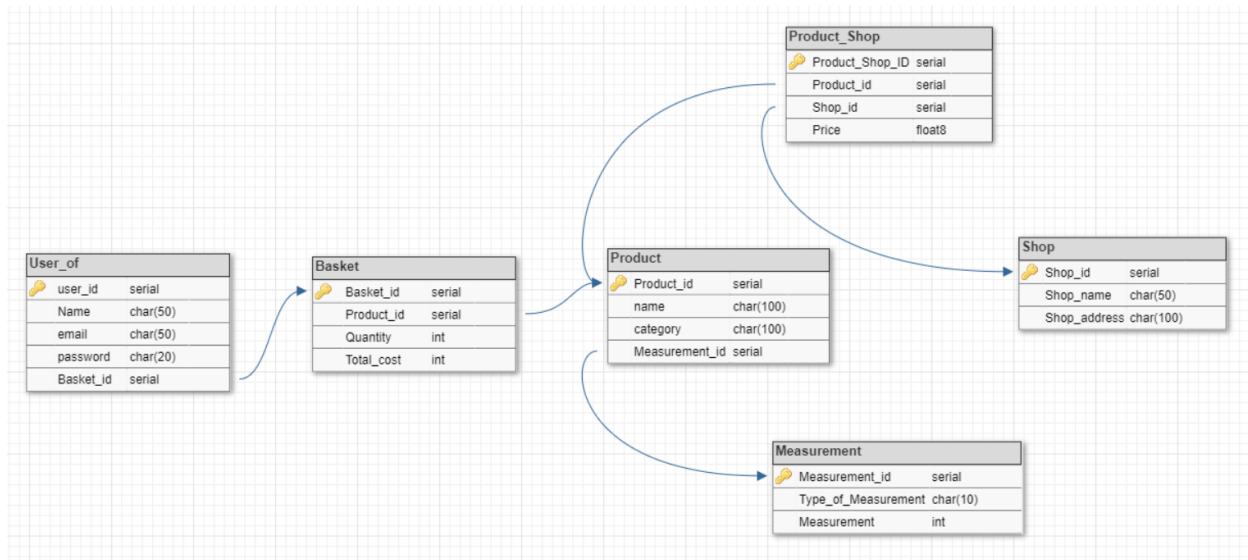
Дата создания: 21 нояб. 2021 г.

Review: Сухов Александр

---

## Физическая модель данных

Создаём физическую модель с помощью конструктора на сайте <https://www.dbdesigner.net/> . Для удобства выберем **PostgreSQL**.



## Скрипт для создания базы данных

Сайт на основе физической модели создает скрипт по которому мы можем создать БД.

```
CREATE TABLE User_of (  
    user_id serial NOT NULL,  
    Name char(50) NOT NULL,  
    email char(50) NOT NULL,
```

```
        password char(20) NOT NULL,  
        CONSTRAINT User_of_pk PRIMARY KEY (user_id)  
);
```

```
CREATE TABLE Basket (  
    Basket_id serial NOT NULL,  
    user_id serial NOT NULL,  
    Product_id serial NOT NULL,  
    Quantity int NOT NULL,  
    Total_cost int,  
    CONSTRAINT Basket_pk PRIMARY KEY (Basket_id)  
);
```

```
CREATE TABLE Product (  
    Product_id serial NOT NULL,  
    name char(100) NOT NULL,  
    category char(100) NOT NULL,  
    Measurement_id serial NOT NULL,  
    CONSTRAINT Product_pk PRIMARY KEY (Product_id)  
);
```

```
CREATE TABLE Measurement (  
    Measurement_id serial NOT NULL,  
    Type_of_Measurement char(10) NOT NULL,  
    Measurement int NOT NULL,  
    CONSTRAINT Measurement_pk PRIMARY KEY (Measurement_id)  
);
```

```
CREATE TABLE Product_Shop (  
    Product_Shop_ID serial NOT NULL,  
    Product_id serial NOT NULL,  
    Shop_id serial NOT NULL,  
    Price float8 NOT NULL,  
    CONSTRAINT Product_Shop_pk PRIMARY KEY (Product_Shop_ID)  
);
```

```
CREATE TABLE Shop (  
    Shop_id serial NOT NULL,  
    Shop_name char(50) NOT NULL,  
    Shop_address char(100) NOT NULL,  
    CONSTRAINT Shop_pk PRIMARY KEY (Shop_id)  
);
```

```
ALTER TABLE Basket ADD CONSTRAINT Basket_fk0 FOREIGN KEY (user_id) REFERENCES  
User_of(user_id);  
ALTER TABLE Basket ADD CONSTRAINT Basket_fk1 FOREIGN KEY (Product_id)  
REFERENCES Product(Product_id);
```

```

ALTER TABLE Product ADD CONSTRAINT Product_fk0 FOREIGN KEY (Measurement_id)
REFERENCES Measurement(Measurement_id);
ALTER TABLE Product_Shop ADD CONSTRAINT Product_Shop_fk0 FOREIGN KEY
(Product_id) REFERENCES Product(Product_id);
ALTER TABLE Product_Shop ADD CONSTRAINT Product_Shop_fk1 FOREIGN KEY
(Shop_id) REFERENCES Shop(Shop_id);

```

## Запросы для заполнения базы данных

Данную базу теперь надо заполнить. Это можно сделать с помощью следующих запросов

```

INSERT INTO User_of(Name, email, password)
VALUES ('Vadim_Shamsutdinov', 'vadimS@mail.ru', 'qwerty123'),
      ('Pasha_Kolyadenko', 'pashaK@mail.ru', 'g2w2bi2212'),
      ('Roma_Ivanov', 'romaI@mail.ru', 'o2e9292b2'),
      ('Vadim_Kasyk', 'vadimk@mail.ru', 'qwefw4323'),
      ('Pauk_Andrew', 'andrewP@mail.ru', 'cnmc29b93c'),
      ('Mark_Barmin', 'markB@mail.ru', 'y2cb23n'),
      ('Rustam_Kadyrov', 'rustamK@mail.ru', 'enw3dwd29'),
      ('Ivan_Evseev', 'ivanE@mail.ru', 'esfwnul22'),
      ('Klim_Dudko', 'klimD@mail.ru', 'iue32b92s'),
      ('Vladimir_Putin', 'creml@mail.ru', 'russia_the_best');

INSERT INTO Measurement(Type_of_Measurement, Measurement)
VALUES ('литр', 1),
      ('грамм', 150),
      ('грамм', 300),
      ('грамм', 100),
      ('штук', 25),
      ('грамм', 500),
      ('грамм', 700),
      ('грамм', 500),
      ('грамм', 500),
      ('литр', 1)
;

INSERT INTO Product(name, category)
VALUES ('Молоко_Красная_цена', 'Молочные изделия'),
      ('Хлопья_Любятово', 'Крупы'),
      ('Майнонез_ЕЖК', 'соусы'),
      ('Миндаль', 'орехи'),
      ('Чай_Lipton', 'чай'),
      ('Печенье_сдобное_Lenta', 'печенье'),
      ('Кетчуп_Ricco', 'соусы'),
      ('Макараны_Makfa', 'бакалея'),
      ('Спаггети_Makfa', 'бакалея'),
      ('Подсолнечное_масло_Слобода', 'бакалея')
;

INSERT INTO Shop(Shop_name, Shop_address)
VALUES ('Ашан_1', 'г. Москва, Ходынский бульвар д. 4, "ТЦ Авиапарк"'),
      ('Lenta_1', 'г. Москва, 7-я Кожуховская ул., 9'),
      ('Metro', 'г. Москва, Складочная ул., 1, стр. 1'),
      ('ОКЕЙ_1', 'г. Москва, Головинское ш., 5, корп. 1, "ТЦ Водный"'),
      ('ОКЕЙ_2', 'г. Москва, Каширское ш., 14, "ТРК ГУД ЗОН"'),
      ('Globus', 'г. Москва, Широкая ул., 2А'),

```

```

('Lenta_2', 'г. Москва, ул. Зои и Александра Космодемьянских, 23'),
('Lenta_3', 'г. Москва, Сокольническая площадь, 4к1-2'),
('Ашан_2', 'г. Москва, ул. Автозаводская д.18, "ТРЦ Ривьера'),
('Ашан_3', 'г. Москва, ул. Бутлерова, д. 3 А')
;
INSERT INTO Product_Shop(Product_id, Shop_id,Price)
VALUES (2, 1, 67.5),
       (2, 2, 77.9),
       (2, 5, 52.5),
       (3, 1, 97.9),
       (3, 6, 107.9),
       (3, 8, 120.9),
       (4, 1, 100),
       (4, 5, 109.9),
       (10, 3, 112.9),
       (9, 1, 44.9),
       (9, 2, 54.9),
       (8, 1, 44.9),
       (6, 2, 119.9),
       (6, 7, 129.9),
       (6, 8, 124.9),
       (5, 3, 87.9),
       (5, 6, 97.5);
INSERT INTO Basket(user_id, Product_id,Quantity )
VALUES (1, 3, 1),
       (1, 2, 2),
       (1, 6, 2),
       (1, 10, 2),
       (1, 7, 1),
       (2, 3, 1),
       (2, 4, 1),
       (2, 10, 1),
       (2, 7, 2),
       (2, 6, 1),
       (2, 2, 5),
       (6, 9, 6),
       (6, 5, 2),
       (6, 10, 2),
       (6, 3, 2),
       (6, 8, 3),
       (6, 2, 3);

```

## Примеры запросов(скриптов) для данный БД

1. Первый запрос выводит список товаров и их измерение в каких-либо единицах и сортирует по алфавиту:

```

select Product.name, Product.category, Measurement.Measurement,
Measurement.Type_of_Measurement from Measurement
inner join Product on Product.Measurement_id =
Measurement.Measurement_id
order by Product.name;

```

Результатом является такой список:

name	category	measurement	type_of_measurement
Кетчуп_Ricco	соусы	700	грамм
Майонез_ЕЖК	соусы	300	грамм
Макараны_Макфа	бакалея	500	грамм
Миндаль	орехи	100	грамм
Молоко_Красная_цена	Молочные изделия	1	литр
Печенье_сдобное_Lenta	печенье	500	грамм
Подсолнечное_масло_Слобода	бакалея	1	литр
Спаггетти_Макфа	бакалея	500	грамм
Хлопья_Любятково	Крупы	150	грамм
Чай_Lipton	чай	25	штук

2. Второй запрос выводит список покупателей, общее количество товаров в корзине и сортирует по убыванию

```
select user_of.name, sum(Basket.Quantity) from User_of
      inner join Basket on User_of.user_id = Basket.user_id
group by User_of.user_id
order by sum(Basket.Quantity) desc;
```

Результатом является такой список:

name	sum
Mark_Barmin	18
Pasha_Kolyadenko	11
Vadim_Shamsutdinov	8

3. Данный запрос выводит список продуктов в разных магазинах(и их адрес) и показывает цены в них.

```
select Product.name, Shop.Shop_name, Shop.Shop_address, Product_Shop.price
from Shop
      inner join Product_Shop on Product_Shop.Shop_id = Shop.Shop_id
      inner join Product on Product.Product_id = Product_Shop.Product_id
where Shop.Shop_name is not NULL;
```

Результат:

name	shop_name	shop_address	price
Хлопья_Любитово	Ашан_1	г. Москва, Ходынский бульвар д. 4, "ТЦ Авиаларс"	67.5
Хлопья_Любитово	Lenta_1	г. Москва, 7-я Кожуховская ул., 9	77.9
Хлопья_Любитово	ОКЕЙ_2	г. Москва, Каширское ш., 14, "ТРК ГУД ЗОН"	52.5
Майонез_ЕЖК	Ашан_1	г. Москва, Ходынский бульвар д. 4, "ТЦ Авиаларс"	97.9
Майонез_ЕЖК	Globus	г. Москва, Широкая ул., 2А	107.9
Майонез_ЕЖК	Lenta_3	г. Москва, Сокольническая площадь, 4к1-2	120.9
Миндаль	Ашан_1	г. Москва, Ходынский бульвар д. 4, "ТЦ Авиаларс"	100
Миндаль	ОКЕЙ_2	г. Москва, Каширское ш., 14, "ТРК ГУД ЗОН"	109.9
Подсолнечное_масло_Слобода	Metro	г. Москва, Сладочная ул., 1, стр. 1	112.9
Сплетни_Makfa	Ашан_1	г. Москва, Ходынский бульвар д. 4, "ТЦ Авиаларс"	44.9
Сплетни_Makfa	Lenta_1	г. Москва, 7-я Кожуховская ул., 9	54.9
Макараном_Makfa	Ашан_1	г. Москва, Ходынский бульвар д. 4, "ТЦ Авиаларс"	44.9
Печенье_сдобное_Lenta	Lenta_1	г. Москва, 7-я Кожуховская ул., 9	119.9
Печенье_сдобное_Lenta	Lenta_2	г. Москва, ул. Зои и Александра Космодемьянских, 23	129.9
Печенье_сдобное_Lenta	Lenta_3	г. Москва, Сокольническая площадь, 4к1-2	124.9
Чай_Lipton	Metro	г. Москва, Сладочная ул., 1, стр. 1	87.9
Чай_Lipton	Globus	г. Москва, Широкая ул., 2А	97.5

4. Этот запрос выводит только те продукты, которые уникальны во всех магазинах(то есть они встречаются только в одном магазине), рядом выводится этот магазин

```
select Product.name, Shop.Shop_name from Product
    inner join Product_Shop on Product.Product_id =
Product_Shop.Product_id
    inner join Shop on Shop.Shop_id = Product_Shop.Shop_id
where Product.product_id in (
    select Product.product_id from Product
        inner join Product_Shop on Product.Product_id =
Product_Shop.Product_id
    group by Product.product_id
    having count(product_shop_id)=1
);
```

Результат:

name	shop_name
Подсолнечное_масло_Слобода	Metro
Макараном_Makfa	Ашан_1

5. Последний запрос показывает, где выгоднее купить продукты из корзины пользователя 'Vadim\_Shamsutdinov'.

```
select distinct Product.name, Shop.Shop_name from User_of
    inner join Basket on User_of.user_id = Basket.user_id
    inner join Product on Product.Product_id = Basket.Product_id
    inner join Product_Shop on Product.Product_id = Product_Shop.Product_id
    inner join Shop on Shop.Shop_id = Product_Shop.Shop_id
where Product_Shop.price in (
    select min(Product_Shop.price) from Product
```

```
inner join Product_Shop on Product.Product_id = Product_Shop.Product_id
inner join Shop on Shop.Shop_id = Product_Shop.Shop_id
group by Product.name
) and user_of.name='Vadim_Shamsutdinov'
order by 1;
```

## Результат:

name	shop_name
Майнонез_ЕЖК	Ашан_1
Печенье_сдобное_Lenta	Lenta_1
Подсолнечное_масло_Слобода	Metro
Хлопья_Любятково	ОКЕЙ_2