

Федеральное государственное автономное образовательное учреждение  
высшего образования  
«Национальный исследовательский университет «Высшая школа  
экономики»  
Московский институт электроники и математики имени А.Н. Тихонова

Отчет по заданию 2:  
По дисциплине Базы данных  
Вариант 8

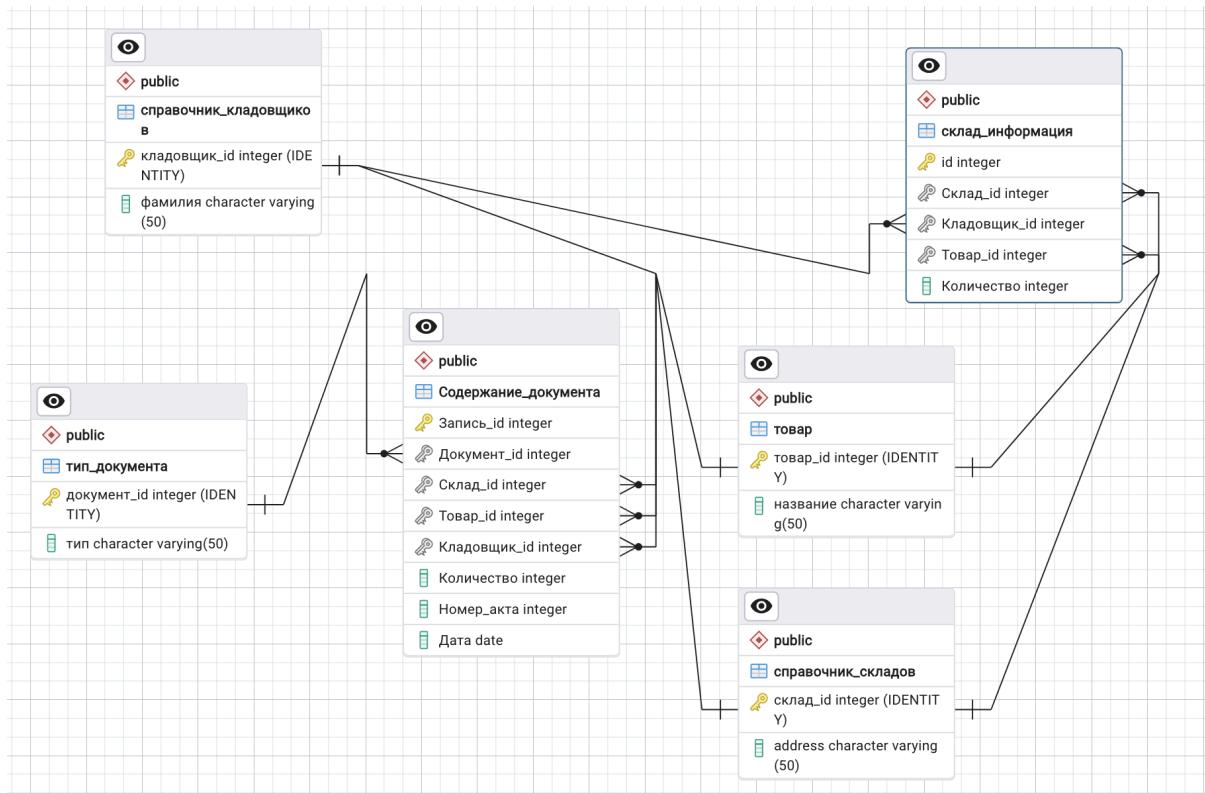
Выполнил:  
Зябкин Даниэль БПМ213

## **Постановка задачи:**

Модель «Склад» должна содержать информацию о складе, товаре, кладовщике, документе (накладная на прием товара, акт на отпуск товара)  
Использовать следующие справочники и классификаторы:

- справочник складов
- справочник кладовщиков
- тип документа

## **1. ERD**



## 2. Описание таблиц

### *Содержание\_документа*

Содержание\_документа

General Columns Advanced Constraints Parameters Security SQL

Import from table(s) Select to import from...

Columns

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	Запись_id	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	Документ_id	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	Склад_id	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	Товар_id	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	Кладовщик_id	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	Количество	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	Номер_акта	integer			<input checked="" type="checkbox"/>	<input type="checkbox"/>	
	Дата	date			<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Первичным ключом данной таблицы является Запись\_id

На следующем скриншоте можно увидеть внешние ключи и то, на что они ссылаются.

Содержание\_документа

General Columns Advanced Constraints Parameters Security SQL

Primary Key Foreign Key Check Unique Exclude

	Name	Columns	Referenced Table
	Документ	(Документ_id) -> (документ_id)	public.тип_документа
	Кладовщик	(Кладовщик_id) -> (кладовщик_id)	public.справочник_кладовщиков
	Склад	(Склад_id) -> (склад_id)	public.справочник_складов
	Товар	(Товар_id) -> (товар_id)	public.товар

### *Тип\_документа*

**тип\_документа**

General Columns Advanced Constraints Parameters Security SQL

Inherited from table(s) Select to inherit from...

**Columns**

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	документ_id	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	тип	character varying	50		<input type="checkbox"/>	<input type="checkbox"/>	

Первичным ключом данной таблицы является документ\_id

### Справочник\_складов

**справочник\_складов**

General Columns Advanced Constraints Parameters Security SQL

Inherited from table(s) Select to inherit from...

**Columns**

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	склад_id	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	address	character varying	50		<input type="checkbox"/>	<input type="checkbox"/>	

Первичным ключом данной таблицы является склад\_id

### Справочник\_кладовщиков

**справочник\_кладовщиков**

General Columns Advanced Constraints Parameters Security SQL

Inherited from table(s) Select to inherit from...

**Columns**

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	кладовщик_id	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	фамилия	character varying	50		<input type="checkbox"/>	<input type="checkbox"/>	

Первичным ключом данной таблицы является кладовщик\_id

### Товар

**товар**

General Columns Advanced Constraints Parameters Security SQL

Inherited from table(s) Select to inherit from...

**Columns**

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	товар_id	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	название	character varying	50		<input type="checkbox"/>	<input type="checkbox"/>	

Первичным ключом данной таблицы является товар\_id

### Склад\_информация

**склад\_информация**

General Columns Advanced Constraints Parameters Security SQL

Inherited from table(s) Select to inherit from...

**Columns**

	Name	Data type	Length/Precision	Scale	Not NULL?	Primary key?	Default
	id	integer			<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
	Склад_id	integer			<input type="checkbox"/>	<input type="checkbox"/>	
	Кладовщик_id	integer			<input type="checkbox"/>	<input type="checkbox"/>	
	Товар_id	integer			<input type="checkbox"/>	<input type="checkbox"/>	
	Количество	integer			<input type="checkbox"/>	<input type="checkbox"/>	

Close Reset Save

Данная таблица содержит информацию о том, какой товар хранится на каком складе, в каком количестве, и кто его отгружал. Первичный ключ - id записи в данной таблице, кроме того, есть и внешние ключи. С ними можно ознакомиться в следующем скриншоте.

**склад\_информация**

General Columns Advanced Constraints Parameters Security SQL

Primary Key Foreign Key Check Unique Exclude

	Name	Columns	Referenced Table
	Кладовщик_id	(Кладовщик_id) -> (кладовщик_id)	public.справочник_кладовщиков
	Склад_id	(Склад_id) -> (склад_id)	public.справочник_складов
	Товар_id	(Товар_id) -> (товар_id)	public.товар

### 3. Запросы

1. Получить список всех товаров, находящихся на всех складах с указанием количества на складе;

The screenshot shows the pgAdmin interface with a query editor and a results viewer. The query is:

```
1 v SELECT "address", "название", "Кол-во" FROM склад_информация
2 JOIN товар ON склад_информация."ТоварID" = товар."товарID"
3 JOIN справочник_складов ON склад_информация."СкладID" = справочник_складов."складID"
```

The results table has columns: address, название, Кол-во. The data is:

	address	название	Кол-во
1	Moscow	Хлеб	250
2	Vladimir	Молоко	700
3	Krasnodar	Соль	200
4	Saint-Petersburg	Вода	500
5	Ryazan	Сахар	600

2. Получить список кладовщиков, принимавших заданный товар, с указанием адреса склада и даты отгрузки;

The screenshot shows the pgAdmin interface with a query editor and a results viewer. The query is:

```
1 v SELECT "фамилия", "address", "Дата" FROM "Содержание_документа"
2 JOIN "справочник_кладовщиков" ON "Содержание_документа"."КладовщикID" = "справочник_кладовщиков"."кладовщикID"
3 JOIN "тип_документа" ON "Содержание_документа"."ДокументID" = "тип_документа"."документID"
4 JOIN "справочник_складов" ON "Содержание_документа"."СкладID" = "справочник_складов"."складID"
5 JOIN "товар" ON "Содержание_документа"."ТоварID" = "товар"."товарID"
6 WHERE "название" = 'Соль' AND "тип" = 'накладная на прием товара'
```

The results table has columns: фамилия, address, Дата. The data is:

	фамилия	address	Дата
1	Кацуба	Saint-Petersburg	2023-04-25

*Если поменять заданный товар, то можно сделать так, что в выводе будет пустая таблица:*

```

1 ▼ SELECT "фамилия", "address", "Дата" FROM "Содержание_документа"
2 JOIN "справочник_кладовщиков" ON "Содержание_документа"."КладовщикID" = "справочник_кладовщиков"."кладовщикID"
3 JOIN "тип_документа" ON "Содержание_документа"."ДокументID" = "тип_документа"."документID"
4 JOIN "справочник_складов" ON "Содержание_документа"."СкладID" = "справочник_складов"."складID"
5 JOIN "товар" ON "Содержание_документа"."ТоварID" = "товар"."товарID"
6 WHERE "название" = 'Хлеб' AND "тип" = 'накладная на прием товара'

```

Data Output Messages Notifications

фамилия character varying (50)	address character varying (50)	Дата date

3. Получить пары (Наименование товара, № акта), отпущенных с заданного склада в заданный день;

```

1 ▼ SELECT "название", "Номер_акта" FROM "Содержание_документа"
2 JOIN "тип_документа" ON "Содержание_документа"."ДокументID" = "тип_документа"."документID"
3 JOIN "товар" ON "Содержание_документа"."ТоварID" = "товар"."товарID"
4 WHERE "Дата" = '2023-07-25' AND "СкладID" = 1 AND "тип" = 'акт на отпуск товара'

```

Data Output Messages Notifications

название character varying (50)	Номер_акта integer
Хлеб	5

4. Получить список складов, на которые был осуществлен прием тех же товаров, что и на заданный склад в указанную дату;

```

1 v SELECT "название" FROM "справочник_складов"
2 JOIN "Содержание_документа" ON "справочник_складов"."складID" = "Содержание_документа"."СкладID"
3 JOIN "тип_документа" ON "Содержание_документа"."ДокументID" = "тип_документа"."документID"
4 JOIN "товар" ON "Содержание_документа"."ТоварID" = "товар"."товарID"
5 WHERE "дата" = '2023-08-25' AND "тип" = 'накладная на прием товара' AND "address" = 'Ryazan'

```

Data Output    Messages    Notifications



	название
1	Вода

- Тут мы поняли, какой товар приняли на заданный склад в указанную дату.

Далее выводим все склады, за исключением заданного, принявшие заданный товар в указанную дату:

```

1 v SELECT "address" FROM "справочник_складов"
2 JOIN "Содержание_документа" ON "справочник_складов"."складID" = "Содержание_документа"."СкладID"
3 JOIN "тип_документа" ON "Содержание_документа"."ДокументID" = "тип_документа"."документID"
4 JOIN "товар" ON "Содержание_документа"."ТоварID" = "товар"."товарID"
5 WHERE "дата" = '2023-08-25' AND "тип" = 'накладная на прием товара' AND "название" = 'Вода' AND "address" != 'Ryazan'

```

Data Output    Messages    Notifications



	address
1	Saint-Petersburg
2	Vladimir

5. Получить номера всех накладных склада на прием тех же товаров, что и принимал заданный кладовщик в указанную дату;

Аналогично, сначала найдем товары, которые заданный кладовщик принимал в указанную дату:

```

1 ✓ SELECT "название" FROM "справочник_кладовщиков"
2 JOIN "Содержание_документа" ON "справочник_кладовщиков"."кладовщикID" = "Содержание_документа"."КладовщикID"
3 JOIN "тип_документа" ON "Содержание_документа"."ДокументID" = "тип_документа"."документID"
4 JOIN "товар" ON "Содержание_документа"."ТоварID" = "товар"."товарID"
5 WHERE "Дата" = '2023-08-25' AND "тип" = 'накладная на прием товара' AND "фамилия" = 'Зябкин'

```

Data Output Messages Notifications

	название	character varying (50)	🔒
1	Вода		

Далее выведем все номера документов на прием этого товара заданного склада:

```

Query Query History
1 ✓ SELECT "Номер_акта" FROM "Содержание_документа"
2 JOIN "товар" ON "Содержание_документа"."ТоварID" = "товар"."товарID"
3 WHERE "название" = 'Вода' AND "СкладID" = 5

```

Data Output Messages Notifications

	Номер_акта	integer	🔒
1		6	
2		9	

6. Получить список (Наименование товара, Номер склада, ФИО кладовщика), принятых по заданной накладной;

```

1 ✓ SELECT "название", "СкладID", "фамилия" FROM "Содержание_документа"
2 JOIN "товар" ON "Содержание_документа"."ТоварID" = "товар"."товарID"
3 JOIN "справочник_кладовщиков" ON "Содержание_документа"."КладовщикID" = "справочник_кладовщиков"."кладовщикID"
4 WHERE "Номер_акта" = 3

```

Data Output Messages Notifications

	название	СкладID	фамилия
	character varying (50)	integer	character varying (50)
1	Молоко	4	Матковский

7. Получить список из трех самых невостребованных товаров на указанном складе (т.е. те, что привозят и отгружают реже всех) с указанием количества;

```
1 ✓ SELECT "название", "Кол-во" FROM "склад_информация"
2 JOIN "товар" ON "склад_информация"."ТоварИД" = "товар"."товарИД"
3 JOIN "справочник_складов" ON "склад_информация"."СкладИД" = "справочник_складов"."складИД"
4 WHERE "address" = 'Moscow'
5 ORDER BY "Кол-во"
6 LIMIT 3
```

Data Output    Messages    Notifications

	название	Кол-во
1	Сахар	50
2	Хлеб	250
3	Соль	300

## Скрипт создания БД:

BEGIN;

```
CREATE TABLE IF NOT EXISTS public."Содержание_документа"
(
    "ЗаписьИД" integer NOT NULL,
    "ДокументИД" integer NOT NULL,
    "СкладИД" integer NOT NULL,
    "ТоварИД" integer NOT NULL,
    "КладовщикИД" integer NOT NULL,
    "Количество" integer NOT NULL,
    "Номер_акта" integer NOT NULL,
    "Дата" date NOT NULL,
    CONSTRAINT "Содержание_документа_pkey" PRIMARY KEY ("ЗаписьИД")
);
```

```
CREATE TABLE IF NOT EXISTS public."склад_информация"
```

```

(
    id integer NOT NULL,
    "СкладИД" integer,
    "ТоварИД" integer,
    "Кол-во" integer,
    CONSTRAINT "склад_информация_pkey" PRIMARY KEY (id)
);

CREATE TABLE IF NOT EXISTS public."справочник_кладовщиков"
(
    "кладовщикИД" integer NOT NULL GENERATED ALWAYS AS IDENTITY (
        INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 2147483647 CACHE 1 ),
    "фамилия" character varying(50) COLLATE pg_catalog."default",
    CONSTRAINT "справочник_кладовщиков_pkey" PRIMARY KEY
    ("кладовщикИД")
);

CREATE TABLE IF NOT EXISTS public."справочник_складов"
(
    "складИД" integer NOT NULL GENERATED ALWAYS AS IDENTITY (
        INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 2147483647 CACHE 1 ),
    address character varying(50) COLLATE pg_catalog."default",
    CONSTRAINT "справочник_складов_pkey" PRIMARY KEY ("складИД")
);

CREATE TABLE IF NOT EXISTS public."тип_документа"
(
    "документИД" integer NOT NULL GENERATED ALWAYS AS IDENTITY (
        INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 2147483647 CACHE 1 ),
    "тип" character varying(50) COLLATE pg_catalog."default",
    CONSTRAINT "тип_документа_pkey" PRIMARY KEY ("документИД")
);

CREATE TABLE IF NOT EXISTS public."товар"
(
    "товарИД" integer NOT NULL GENERATED ALWAYS AS IDENTITY (
        INCREMENT 1 START 1 MINVALUE 1 MAXVALUE 2147483647 CACHE 1 ),
    "название" character varying(50) COLLATE pg_catalog."default",
    CONSTRAINT "товар_pkey" PRIMARY KEY ("товарИД")
);

ALTER TABLE IF EXISTS public."Содержание_документа"
ADD CONSTRAINT "Документ" FOREIGN KEY ("ДокументИД")
REFERENCES public."тип_документа" ("документИД") MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION
NOT VALID;

ALTER TABLE IF EXISTS public."Содержание_документа"
ADD CONSTRAINT "Кладовщик" FOREIGN KEY ("КладовщикИД")

```

```
REFERENCES public."справочник_кладовщиков" ("кладовщикИД") MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION
NOT VALID;
```

```
ALTER TABLE IF EXISTS public."Содержание_документа"
ADD CONSTRAINT "Склад" FOREIGN KEY ("СкладИД")
REFERENCES public."справочник_складов" ("складИД") MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION
NOT VALID;
```

```
ALTER TABLE IF EXISTS public."Содержание_документа"
ADD CONSTRAINT "Товар" FOREIGN KEY ("ТоварИД")
REFERENCES public."товар" ("товарИД") MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION
NOT VALID;
```

```
ALTER TABLE IF EXISTS public."склад_информация"
ADD CONSTRAINT "Склад_id" FOREIGN KEY ("СкладИД")
REFERENCES public."справочник_складов" ("складИД") MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION
NOT VALID;
```

```
ALTER TABLE IF EXISTS public."склад_информация"
ADD CONSTRAINT "Товар_id" FOREIGN KEY ("ТоварИД")
REFERENCES public."товар" ("товарИД") MATCH SIMPLE
ON UPDATE NO ACTION
ON DELETE NO ACTION
NOT VALID;
```

```
END;
```