

Министерство науки и высшего образования Российской Федерации
Федеральное государственное автономное образовательное учреждение
высшего образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»
Факультет инфокоммуникационных технологий

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 1.2

по теме: Создание таблиц базы данных postgresql. Заполнение
таблиц рабочими данными.

по дисциплине: Проектирование и реализация баз данных

Специальность:

09.03.03 Мобильные и сетевые технологии

Проверил:

Говорова М.М. _____

Дата: «__» _____ 20__ г.

Оценка _____

Выполнил:

студент группы К3241

Еремеева А. А.

Санкт-Петербург
2022

ЦЕЛЬ РАБОТЫ

Овладеть практическими навыками создания таблиц базы данных PostgreSQL 1X, заполнения их рабочими данными, резервного копирования и восстановления БД.

ПРАКТИЧЕСКОЕ ЗАДАНИЕ

Оборудование: компьютерный класс.

Программное обеспечение: СУБД PostgreSQL 1X, pgAdmin 4.

Практическое задание:

1. Создать базу данных с использованием pgAdmin 4 (согласно индивидуальному заданию).
2. Создать схему в составе базы данных.
3. Создать таблицы базы данных.
4. Установить ограничения на данные: *Primary Key, Unique, Check, Foreign Key*.
5. Заполнить таблицы БД рабочими данными.
6. Создать резервную копию БД. *Указание:*

Создать две резервные копии:

- с расширением *CUSTOM* для восстановления БД;
 - с расширением *PLAIN* для листинга (в отчете);
 - при создании резервных копий БД настроить параметры *Dump options* для *Type of objects* и *Queries*.
7. Восстановить БД.

Вариант 9. БД «Оптовая база»

Описание предметной области: Оптовая база закупает товары у компаний-поставщиков и поставляет их компаниям – покупателям. Доход оптовой базы составляет не менее 5% от стоимости товара, проданного компании – покупателю. Один и тот же товар может доставляться несколькими поставщиками, и один и тот же поставщик может поставлять несколько видов товаров. Цены поставки товара у разных поставщиков могут отличаться. Поставки и заказы обслуживают менеджеры по работе с клиентами (по поставкам и продажам).

БД должна содержать следующий минимальный набор сведений: Табельный номер. Код сотрудника. Паспортные данные сотрудника. Должность. Код товара. Название товара. Единица измерения товара. Количество товара. Запас товара на базе. Стоимость единицы товара. Код

поставки. Дата поставки на базу. Количество поставки. Примечание – описание товара. Код поставщика. Название компании поставщика. Адрес поставщика. Дата поставки. Количество товара в партии. Номер счета. Код организации – покупателя. Название компании покупателя. Адрес покупателя. Дата заказа. Дата вывоза. Номер партии. Продажная цена товара.

ХОД РАБОТЫ

1) **Наименование БД:** wholesale_base

2) **Схема логической модели:**



3) **Dump, содержащий скрипты работы с БД**

```

PGDMP
zwholesale_base13.613.6A_
00ENCODINGENCODINGSET client_encoding = 'UTF8';
false
00

falseh
00

falseh
126216394wholesale_baseDATABASEYCREATE DATABASE wholesale_base WITH TEMPLATE = template0 ENCODING = 'UTF8' LOCALE = 'C';

name character varying(30) NOT NULL,
address character varying(100) NOT NULL,
account_number integer NOT NULL,
id_customer integer NOT NULL
);

SEQUENCE NAME wholesale_base.customer_id_customer_seg
START WITH 1
INCREMENT BY 1
NO MINVALUE
NO MAXVALUE
CACHE 1
);

id_employee integer NOT NULL,
post character varying(50) NOT NULL,
name character varying(30) NOT NULL,
passport_series integer,
passport_number integer,
CONSTRAINT employee_passport_number_check CHECK ((passport_number <= 999999)),
CONSTRAINT employee_passport_series_check CHECK ((passport_series <= 9999))
);

SEQUENCE NAME wholesale_base.employee_service_number_seg
START WITH 1
INCREMENT BY 1
NO MINVALUE
NO MAXVALUE
CACHE 1
);

id_order integer NOT NULL,
id_employee integer NOT NULL,
id_customer integer NOT NULL,
order_date date NOT NULL,
delivery_date date NOT NULL,
order_status character varying(20) NOT NULL,
order_pay character varying(20) NOT NULL,
CONSTRAINT order_check CHECK ((order_date < delivery_date))
);

SEQUENCE NAME wholesale_base.order_id_order_seg
START WITH 1
INCREMENT BY 1
NO MINVALUE
NO MAXVALUE
CACHE 1
);

name character varying NOT NULL,
product_price integer NOT NULL,
unit_measurement character varying(5) NOT NULL,
product_description character varying(500) NOT NULL,
type_product integer NOT NULL,
id_product integer NOT NULL,
CONSTRAINT product_product_price_check CHECK ((product_price > 0))
);

SEQUENCE NAME wholesale_base.product_id_product_seg
START WITH 1
INCREMENT BY 1
NO MINVALUE
NO MAXVALUE

```

```

    CACHE 1
);

    name character varying(30) NOT NULL,
    address character varying(100) NOT NULL,
    account_number integer NOT NULL,
    id_provider integer NOT NULL
);

    AS integer
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1;

wholesale_basepostgresfalse6204K
00provider_id_provider_seqSEQUENCE OWNED BYALTER SEQUENCE wholesale_base.provider_id_provider_seq OWNED BY wholesale_base.provider.id_provider;

    batch_number integer NOT NULL,
    order_price integer NOT NULL,
    id_product integer NOT NULL,
    id_order integer NOT NULL,
    amount_product integer NOT NULL,
    CONSTRAINT scope_of_order_amount_product_check CHECK ((amount_product > 0)),
    CONSTRAINT scope_of_order_order_price_check CHECK ((order_price > 0))
);

    SEQUENCE NAME wholesale_base.scope_of_order_batch_number_seq
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1
);

    batch_number integer NOT NULL,
    shipment_price integer NOT NULL,
    id_product integer NOT NULL,
    id_shipment integer NOT NULL,
    amount_shipment integer NOT NULL,
    CONSTRAINT scope_of_shipment_amount_shipment_check CHECK ((amount_shipment > 0)),
    CONSTRAINT scope_of_shipment_batch_price_check CHECK ((shipment_price > 0))
);

    SEQUENCE NAME wholesale_base.scope_of_shipment_batch_number_seq
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1
);

    id_provider integer NOT NULL,
    id_employee integer NOT NULL,
    shipment_date date NOT NULL,
    contract_date date NOT NULL,
    status_pay character varying(20) NOT NULL,
    status_shipment character varying(20) NOT NULL,
    id_shipment integer NOT NULL,
    CONSTRAINT shipment_check CHECK ((shipment_date >= contract_date))
);

    SEQUENCE NAME wholesale_base.shipment_id_shipment_seq
    START WITH 1
    INCREMENT BY 1
    NO MINVALUE
    NO MAXVALUE
    CACHE 1
);

    id_product integer NOT NULL,
    storage_date date NOT NULL,
    count_in_storage integer NOT NULL,
    product_sold integer NOT NULL,
    product_received integer NOT NULL,
    id_storage integer NOT NULL,
    CONSTRAINT storage_check CHECK ((product_received >= (product_sold + count_in_storage)))
);

```

```

AS integer
START WITH 1
INCREMENT BY 1
NO MINVALUE
NO MAXVALUE
CACHE 1;

wholesale_basepostgresfalse206k
00storage_id_storage_seqSEQUENCE OWNED BYALTER SEQUENCE wholesale_base.storage_id_storage_seq OWNED BY wholesale_base.storage_id_storage;
wholesale_basepostgresfalse211
260416527provider_id_providerDEFAULTALTER TABLE ONLY wholesale_base.provider ALTER COLUMN id_provider SET DEFAULT nextval('wholesale_base.provider_id_provider_seq'::regclass);

wholesale_basepostgresfalse210204
260416577storage_id_storageDEFAULTALTER TABLE ONLY wholesale_base.storage ALTER COLUMN id_storage SET DEFAULT nextval('wholesale_base.storage_id_storage_seq'::regclass);

wholesale_basepostgresfalse211206w
016436customer

wholesale_basepostgresfalse209NZu
016396employee

wholesale_basepostgresfalse201MJA
016431order

wholesale_basepostgresfalse208Rvc
016416product

wholesale_basepostgresfalse2055JE
016411provider

wholesale_basepostgresfalse2044_i
016426scope_of_order

wholesale_basepostgresfalse207i`J
016406scope_of_shipment

wholesale_basepostgresfalse203ar
016401shipment

wholesale_basepostgresfalse202PaI
016421storage

wholesale_basepostgresfalse206,as
00customer_id_customer_seq
SEQUENCE SETSELECT pg_catalog.setval('wholesale_base.customer_id_customer_seq', 5, true);
wholesale_basepostgresfalse217
00employee_service_number_seq
SEQUENCE SETSELECT pg_catalog.setval('wholesale_base.employee_service_number_seq', 18, true);
wholesale_basepostgresfalse214-
00order_id_order_seq
SEQUENCE SETSELECT pg_catalog.setval('wholesale_base.order_id_order_seq', 26, true);
wholesale_basepostgresfalse212"
00product_id_product_seq
SEQUENCE SETSELECT pg_catalog.setval('wholesale_base.product_id_product_seq', 5, true);
wholesale_basepostgresfalse216"
00provider_id_provider_seq
SEQUENCE SETSELECT pg_catalog.setval('wholesale_base.provider_id_provider_seq', 5, true);
wholesale_basepostgresfalse210"
00scope_of_order_batch_number_seq
SEQUENCE SETSELECT pg_catalog.setval('wholesale_base.scope_of_order_batch_number_seq', 11, true);
wholesale_basepostgresfalse213"
00scope_of_shipment_batch_number_seq
SEQUENCE SETSELECT pg_catalog.setval('wholesale_base.scope_of_shipment_batch_number_seq', 10, true);
wholesale_basepostgresfalse215+
00shipment_id_shipment_seq
SEQUENCE SETSELECT pg_catalog.setval('wholesale_base.shipment_id_shipment_seq', 8, true);
wholesale_basepostgresfalse218,
00storage_id_storage_seq
SEQUENCE SETSELECT pg_catalog.setval('wholesale_base.storage_id_storage_seq', 5, true);
wholesale_basepostgresfalse211
260616504customer accountCHECK CONSTRAINTgALTER TABLE wholesale_base.customer
ADD CONSTRAINT account CHECK ((account_number > 0)) NOT VALID;

wholesale_basepostgresfalse209209)
260616693customer customer_pkey

ADD CONSTRAINT customer_pkey PRIMARY KEY (id_customer);

wholesale_basepostgresfalse209
2606167075employee employee_passport_series_passport_number_key

ADD CONSTRAINT employee_passport_series_passport_number_key UNIQUE (passport_series, passport_number);

```

```

wholesale_basepostgresfalse201201
260616400employee employee_pkey

    ADD CONSTRAINT employee_pkey PRIMARY KEY (id_employee);

wholesale_basepostgresfalse201'
260616435order order_pkey

    ADD CONSTRAINT order_pkey PRIMARY KEY (id_order);

wholesale_basepostgresfalse208!
260616669product product_pkey

    ADD CONSTRAINT product_pkey PRIMARY KEY (id_product);

wholesale_basepostgresfalse205
260616533provider provider_pkey

    ADD CONSTRAINT provider_pkey PRIMARY KEY (id_provider);

wholesale_basepostgresfalse204%
260616430"scope_of_order scope_of_order_pkey

    ADD CONSTRAINT scope_of_order_pkey PRIMARY KEY (batch_number);

wholesale_basepostgresfalse207
260616410(scope_of_shipment scope_of_shipment_pkey

    ADD CONSTRAINT scope_of_shipment_pkey PRIMARY KEY (batch_number);

wholesale_basepostgresfalse203
260616636shipment shipment_pkey

    ADD CONSTRAINT shipment_pkey PRIMARY KEY (id_shipment);

wholesale_basepostgresfalse202#
260616582storage storage_pkey

    ADD CONSTRAINT storage_pkey PRIMARY KEY (id_storage);

wholesale_basepostgresfalse206+
260616550shipment_id_employee_fkey

    ADD CONSTRAINT id_employee_fkey FOREIGN KEY (id_employee) REFERENCES wholesale_base.employee(id_employee) NOT VALID;

wholesale_basepostgresfalse3097202201*
260616545shipment_id_provider_fkey

    ADD CONSTRAINT id_provider_fkey FOREIGN KEY (id_provider) REFERENCES wholesale_base.provider(id_provider) NOT VALID;

wholesale_basepostgresfalse31032042022
260616694order order_id_customer_fkey

    ADD CONSTRAINT order_id_customer_fkey FOREIGN KEY (id_customer) REFERENCES wholesale_base.customer(id_customer) NOT VALID;

wholesale_basepostgresfalse31132082091
260616593order order_id_employee_fkey

    ADD CONSTRAINT order_id_employee_fkey FOREIGN KEY (id_employee) REFERENCES wholesale_base.employee(id_employee) NOT VALID;

wholesale_basepostgresfalse2012083007/
260616616+scope_of_order scope_of_order_id_order_fkey

    ADD CONSTRAINT scope_of_order_id_order_fkey FOREIGN KEY (id_order) REFERENCES wholesale_base."order"(id_order) NOT VALID;

wholesale_basepostgresfalse20820731110
260616681-scope_of_order scope_of_order_id_product_fkey

    ADD CONSTRAINT scope_of_order_id_product_fkey FOREIGN KEY (id_product) REFERENCES wholesale_base.product(id_product) NOT VALID;

wholesale_basepostgresfalse3105207205-
2606166713scope_of_shipment scope_of_shipment_id_product_fkey

    ADD CONSTRAINT scope_of_shipment_id_product_fkey FOREIGN KEY (id_product) REFERENCES wholesale_base.product(id_product) NOT VALID;

wholesale_basepostgresfalse2053105203,
2606166494scope_of_shipment scope_of_shipment_id_shipment_fkey

    ADD CONSTRAINT scope_of_shipment_id_shipment_fkey FOREIGN KEY (id_shipment) REFERENCES wholesale_base.shipment(id_shipment) NOT VALID;

wholesale_basepostgresfalse2032023099,
260616676storage storage_id_product_fkey

    ADD CONSTRAINT storage_id_product_fkey FOREIGN KEY (id_product) REFERENCES wholesale_base.product(id_product) NOT VALID;

    ADD CONSTRAINT scope_of_shipment_id_shipment_fkey FOREIGN KEY (id_shipment) REFERENCES wholesale_base.shipment(id_shipment) NOT VALID;

wholesale_basepostgresfalse2032023099,
260616676storage storage_id_product_fkey

    ADD CONSTRAINT storage_id_product_fkey FOREIGN KEY (id_product) REFERENCES wholesale_base.product(id_product) NOT VALID;

```

Вывод:

В ходе выполнения работы:

- была создана база данных в PostgreSQL
- логическая схема
- созданы таблицы и заданы ограничения на данные: PK, Unique, Check, FK
- в БД были внесены данные
- создана резервная копия
- произведено восстановление базы данных