Санкт-Петербургский национальный исследовательский университет информационных технологий, механики и оптики

# Лабораторная работа№ 1.2

«Создание таблиц базы данных POSTGRESQL. Заполнение таблиц рабочими данными»

Выполнил: Галиновский Роман Андреевич

Группа: K3240 Преподаватель: Говорова Марина Михайловна

Санкт-Петербург 2022

**Цель работы:** овладеть практическими навыками создания таблиц базы данных PostgreSQL 1Х, заполнения их рабочими данными, резервного копирования и восстановления БД.

**Оборудование**: компьютерный класс.

**Программное обеспечение:** СУБД PostgreSQL 1Х, 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 .*

1. Восстановить БД.

## ТЕХНОЛОГИЯ ВЫПОЛНЕНИЯ РАБОТЫ:

### Название БД

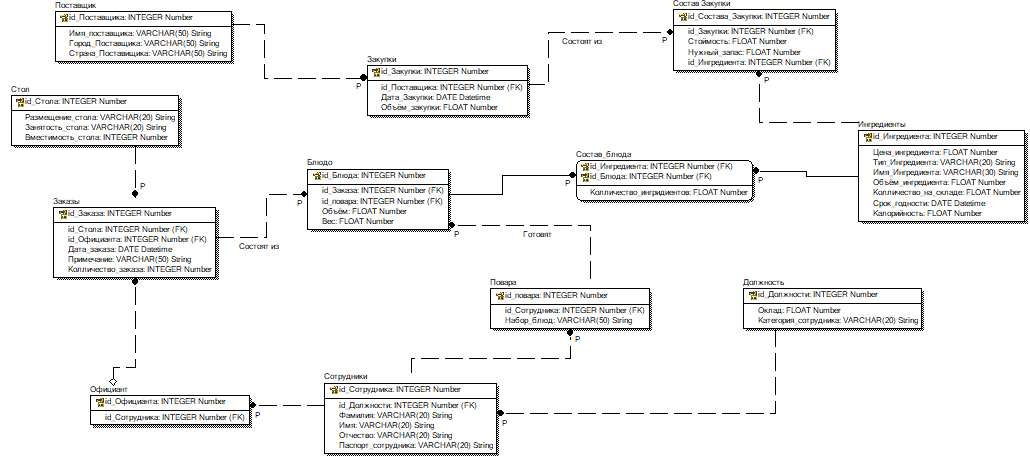
**Вариант 13. «Ресторан»**

Описание предметной области: Сотрудники ресторана – повара и официанты. За каждым официантом закреплены определенные столы. Каждый повар готовит определенный набор блюд. Запас продуктов на складе не должен быть ниже заданного значения. Цена заказа складывается из стоимости ингредиентов и наценки, которая составляет 40% стоимости ингредиентов. БД должна содержать следующий минимальный набор сведений: ФИО сотрудника. Паспортные данные сотрудника. Категория сотрудника. Должность сотрудника. Оклад сотрудника. Наименование ингредиента. Код ингредиента. Дата закупки. Объем закупки. Количество продукта на складе. Необходимый запас продукта. Срок годности. Цена ингредиента. Поставщик. Наименование блюда. Код блюда. Объем ингредиента. Номер стола. Дата заказа. Код заказа. Количество. Название блюда. Ингредиенты, входящие в блюдо. Тип ингредиента.

### Состав реквизитов сущностей:

* + 1. **Ингредиент (**ID ингредиента, цена ингредиента, тип ингредиента, название ингредиента, объём ингредиента, количество на складе, срок годности, калорийность)
    2. **Поставщик** (ID поставщика, имя поставщика, город поставщика, страна поставщика)
    3. **Стол** (ID стола, размещение стола, занятость стола, вместимость стола)
    4. **Должность** (ID должности, оклад, категория сотрудника)
    5. **Закупки** (ID закупки, ID поставщика, дата закупки, объём закупки)
    6. **Состав закупки** (ID состава закупки, ID закупки, ID ингредиента, стоймость, нужный запас)
    7. **Сотрудники** (ID сотрудника, ID должности, Фамилия, имя, отчество, паспорт сотрудника)
    8. **Официант** (ID Официанта, ID сотрудника)
    9. **Заказы** (ID заказа, ID стола, ID Официанта, Дата заказа, Примечание, Количество заказа)
    10. **Повара** (ID Повара, ID сотрудника, набор блюд)
    11. **Блюдо** (ID блюда, ID заказа, ID повара, объём, вес)
    12. **Состав Блюда** (ID ингредиента, ID блюда, количество ингредиентов)

## Схема логической модели БД в нотации IDEF1X:

****

**Заполнение таблиц рабочими данными** Для заполнения использовался скрипт: *INSERT INTO (модификатор доступа).”Имя таблицы”(столбцы) VALUES (значения)*

Изображение выглядит как стол

Автоматически созданное описание

Изображение выглядит как стол

Автоматически созданное описание

**ДАМП СО СКРИПТАМИ:**

Создаем базу данных:

CREATE DATABASE courses WITH TEMPLATE = template0 ENCODING = 'UTF8' LOCALE = 'Russian\_Russia.1251';

ALTER DATABASE VAR13 OWNER TO postgres;

\connect VAR13

SET statement\_timeout = 0; SET lock\_timeout = 0;

SET idle\_in\_transaction\_session\_timeout = 0; SET client\_encoding = 'UTF8';

SET standard\_conforming\_strings = on;

SELECT pg\_catalog.set\_config('search\_path', '', false); SET check\_function\_bodies = false;

SET xmloption = content;

SET client\_min\_messages = warning; SET row\_security = off;

--

-- Name: courses; Type: SCHEMA; Schema: -; Owner: postgres

Создаем схему:

CREATE SCHEMA VAR13;

ALTER SCHEMA VAR13 OWNER TO postgres; SET default\_tablespace = '';

SET default\_table\_access\_method = heap;

Создаем таблицы:

--

-- TOC entry 209 (class 1259 OID 49173)

-- Name: Cook; Type: TABLE; Schema: public; Owner: postgres

--

CREATE TABLE public."Cook" (

id\_cook integer NOT NULL,

set\_of\_dishes character varying(50) NOT NULL,

id\_employer integer NOT NULL

);

ALTER TABLE public."Cook" OWNER TO postgres;

--

-- TOC entry 210 (class 1259 OID 49176)

-- Name: Dish\_composition; Type: TABLE; Schema: public; Owner: postgres

--

CREATE TABLE public."Dish\_composition" (

id\_ingredient integer NOT NULL,

id\_dish integer NOT NULL,

count\_ingredient real NOT NULL

);

ALTER TABLE public."Dish\_composition" OWNER TO postgres;

--

-- TOC entry 211 (class 1259 OID 49179)

-- Name: Dishes; Type: TABLE; Schema: public; Owner: postgres

--

CREATE TABLE public."Dishes" (

id\_dish integer NOT NULL,

id\_order integer NOT NULL,

id\_cook integer NOT NULL,

volume real NOT NULL,

weight real NOT NULL

);

ALTER TABLE public."Dishes" OWNER TO postgres;

--

-- TOC entry 212 (class 1259 OID 49182)

-- Name: Employer; Type: TABLE; Schema: public; Owner: postgres

--

CREATE TABLE public."Employer" (

id\_employer integer NOT NULL,

id\_position integer NOT NULL,

family character varying(20) NOT NULL,

name character varying(20) NOT NULL,

patronomyc character varying(20) NOT NULL,

employer\_passport character varying(20) NOT NULL

);

ALTER TABLE public."Employer" OWNER TO postgres;

--

-- TOC entry 213 (class 1259 OID 49185)

-- Name: Ingredient; Type: TABLE; Schema: public; Owner: postgres

--

CREATE TABLE public."Ingredient" (

id\_ingredient integer NOT NULL,

ingredient\_price real NOT NULL,

ingredient\_type character varying(20) NOT NULL,

ingredient\_name character varying(20) NOT NULL,

ingredient\_volume real NOT NULL,

count\_on\_warehouse real NOT NULL,

ingredient\_bbd date NOT NULL,

calories real NOT NULL

);

ALTER TABLE public."Ingredient" OWNER TO postgres;

--

-- TOC entry 214 (class 1259 OID 49188)

-- Name: Orders; Type: TABLE; Schema: public; Owner: postgres

--

CREATE TABLE public."Orders" (

id\_order integer NOT NULL,

id\_table integer NOT NULL,

id\_waiter integer NOT NULL,

date\_order date NOT NULL,

notion character varying(50),

count\_order integer NOT NULL

);

ALTER TABLE public."Orders" OWNER TO postgres;

--

-- TOC entry 215 (class 1259 OID 49191)

-- Name: Platen; Type: TABLE; Schema: public; Owner: postgres

--

CREATE TABLE public."Platen" (

id\_table integer NOT NULL,

table\_occupation character varying(20),

table\_capacity integer NOT NULL,

place\_of\_table character varying(20)

);

ALTER TABLE public."Platen" OWNER TO postgres;

--

-- TOC entry 216 (class 1259 OID 49194)

-- Name: Position; Type: TABLE; Schema: public; Owner: postgres

--

CREATE TABLE public."Position" (

id\_position integer NOT NULL,

salary real NOT NULL,

employer\_category character varying(20) NOT NULL

);

ALTER TABLE public."Position" OWNER TO postgres;

--

-- TOC entry 217 (class 1259 OID 49197)

-- Name: Purchase; Type: TABLE; Schema: public; Owner: postgres

--

CREATE TABLE public."Purchase" (

id\_purchase integer NOT NULL,

purchase\_date date,

purchase\_volume real NOT NULL,

id\_supplier integer NOT NULL

);

ALTER TABLE public."Purchase" OWNER TO postgres;

--

-- TOC entry 218 (class 1259 OID 49200)

-- Name: Purchase\_composition; Type: TABLE; Schema: public; Owner: postgres

--

CREATE TABLE public."Purchase\_composition" (

id\_purchase\_composition integer NOT NULL,

id\_purchase integer NOT NULL,

cost real NOT NULL,

right\_stock real NOT NULL,

id\_ingredient integer NOT NULL

);

ALTER TABLE public."Purchase\_composition" OWNER TO postgres;

--

-- TOC entry 219 (class 1259 OID 49203)

-- Name: Supplier; Type: TABLE; Schema: public; Owner: postgres

--

CREATE TABLE public."Supplier" (

id\_supplier integer NOT NULL,

supplier\_name character varying(50),

supplier\_city character varying(50),

supplier\_country character varying(50)

);

ALTER TABLE public."Supplier" OWNER TO postgres;

--

-- TOC entry 220 (class 1259 OID 49206)

-- Name: Supplier\_id\_supplier\_seq; Type: SEQUENCE; Schema: public; Owner: postgres

--

ALTER TABLE public."Supplier" ALTER COLUMN id\_supplier ADD GENERATED ALWAYS AS IDENTITY (

SEQUENCE NAME public."Supplier\_id\_supplier\_seq"

START WITH 1

INCREMENT BY 1

NO MINVALUE

NO MAXVALUE

CACHE 1

);

--

-- TOC entry 221 (class 1259 OID 49207)

-- Name: Table\_id\_table\_seq; Type: SEQUENCE; Schema: public; Owner: postgres

--

ALTER TABLE public."Platen" ALTER COLUMN id\_table ADD GENERATED ALWAYS AS IDENTITY (

SEQUENCE NAME public."Table\_id\_table\_seq"

START WITH 1

INCREMENT BY 1

NO MINVALUE

NO MAXVALUE

CACHE 1

);

--

-- TOC entry 222 (class 1259 OID 49208)

-- Name: Waiter; Type: TABLE; Schema: public; Owner: postgres

--

CREATE TABLE public."Waiter" (

id\_waiter integer NOT NULL,

id\_employer integer NOT NULL

);

ALTER TABLE public."Waiter" OWNER TO postgres;

--

-- TOC entry 223 (class 1259 OID 49211)

-- Name: Waiter\_id\_waiter\_seq; Type: SEQUENCE; Schema: public; Owner: postgres

--

ALTER TABLE public."Waiter" ALTER COLUMN id\_waiter ADD GENERATED ALWAYS AS IDENTITY (

SEQUENCE NAME public."Waiter\_id\_waiter\_seq"

START WITH 1

INCREMENT BY 1

NO MINVALUE

NO MAXVALUE

CACHE 1

);

Задаем ограничения Checks и Foreign key:

COPY public."Cook" (id\_cook, set\_of\_dishes, id\_employer) FROM stdin;

111141 fried/stewed 111123

111142 stewed/boiled 111124

111143 boiled/fried 111125

\.

--

-- TOC entry 3436 (class 0 OID 49176)

-- Dependencies: 210

-- Data for Name: Dish\_composition; Type: TABLE DATA; Schema: public; Owner: postgres

--

COPY public."Dish\_composition" (id\_ingredient, id\_dish, count\_ingredient) FROM stdin;

111181 111171 1

111182 111171 1

111186 111171 3

111186 111172 2

111184 111172 2

111188 111172 4

111183 111173 2

111188 111173 2

111182 111173 1

111185 111174 1

111181 111174 2

111186 111174 2

111184 111175 3

111181 111175 2

111187 111175 2

111183 111176 3

111182 111176 2

111187 111176 2

111187 111177 4

111188 111177 4

111186 111177 4

111187 111178 4

111188 111178 4

111181 111179 3

111182 111179 2

111181 111800 3

\.

--

-- TOC entry 3437 (class 0 OID 49179)

-- Dependencies: 211

-- Data for Name: Dishes; Type: TABLE DATA; Schema: public; Owner: postgres

--

COPY public."Dishes" (id\_dish, id\_order, id\_cook, volume, weight) FROM stdin;

111171 111161 111141 150.5 420

111172 111162 111142 90.3 115.2

111173 111163 111143 40.6 50

111174 111164 111142 45.2 55.7

111175 111165 111143 75.5 90.1

111176 111166 111141 80 100.2

111177 111167 111143 85.3 125.2

111178 111168 111142 160.3 330.7

111179 111169 111141 36.6 78.8

111800 111700 111142 455.5 774.7

\.

--

-- TOC entry 3438 (class 0 OID 49182)

-- Dependencies: 212

-- Data for Name: Employer; Type: TABLE DATA; Schema: public; Owner: postgres

--

COPY public."Employer" (id\_employer, id\_position, family, name, patronomyc, employer\_passport) FROM stdin;

111121 111114 Nagiev Dmitriy Vladimirovich 6716 53355

111122 111111 Victoryia Lazareva Sergeevna 6728 22837

111123 111113 Dobroslave Shiryaev Nurzupaevicg 6111 11111

111124 111113 Ismail Darzaev Umarshapaevich 6111 195851

111125 111113 Maxim Prihodko Tatianovich 6111 133337

111126 111112 Ksenya Sergeeva Sergeevna 11 124566

111127 111112 Mihail Romanov Yanovich 6666 133728

111128 111112 Mihail Zaxarov Yanovich 6636 135628

\.

--

-- TOC entry 3439 (class 0 OID 49185)

-- Dependencies: 213

-- Data for Name: Ingredient; Type: TABLE DATA; Schema: public; Owner: postgres

--

COPY public."Ingredient" (id\_ingredient, ingredient\_price, ingredient\_type, ingredient\_name, ingredient\_volume, count\_on\_warehouse, ingredient\_bbd, calories) FROM stdin;

111181 100 no vegan eggs 50 400 2022-08-31 115.5

111182 45 vegan bread 35.5 40 2022-06-30 45.5

111183 170 no vegan chicken 400 150 2022-07-15 170.5

111184 300 no vegan beef 450 250 2022-07-10 245.5

111185 250 no vegan pig 445 225 2022-07-10 235.5

111186 70 vegan grain 60 500 2023-01-01 0.85

111187 80 vegan fruits 70 500 2022-06-30 15.8

111188 80 vegan vegetables 70 500 2022-06-30 15.8

\.

--

-- TOC entry 3440 (class 0 OID 49188)

-- Dependencies: 214

-- Data for Name: Orders; Type: TABLE DATA; Schema: public; Owner: postgres

--

COPY public."Orders" (id\_order, id\_table, id\_waiter, date\_order, notion, count\_order) FROM stdin;

111161 111151 111131 2022-05-31 no 5

111162 111151 111133 2022-06-01 no salt 6

111163 111153 111132 2022-06-02 no chilly 3

111164 111155 111131 2022-06-03 no sugar 4

111165 111157 111132 2022-06-04 no 4

111166 111158 111133 2022-06-05 no 4

111167 111158 111132 2022-06-07 no 5

111168 111159 111133 2022-06-08 no 2

111169 111153 111131 2022-06-09 no 3

111700 111159 111132 2022-06-10 no sauce 2

\.

--

-- TOC entry 3441 (class 0 OID 49191)

-- Dependencies: 215

-- Data for Name: Platen; Type: TABLE DATA; Schema: public; Owner: postgres

--

COPY public."Platen" (id\_table, table\_occupation, table\_capacity, place\_of\_table) FROM stdin;

111151 armored 5 exit

111152 free 5 exit

111153 armored 2 kitchen

111155 armored 4 middle room

111154 free 20 middle room

111156 free 10 kitchen

111157 armored 4 middle room

111158 armored 4 exit

111159 armored 2 middle room

\.

--

-- TOC entry 3442 (class 0 OID 49194)

-- Dependencies: 216

-- Data for Name: Position; Type: TABLE DATA; Schema: public; Owner: postgres

--

COPY public."Position" (id\_position, salary, employer\_category) FROM stdin;

111111 30000 manager

111112 25000 waiter

111113 60000 cook

111114 75000 director

111115 15000 cleaner

111116 20000 barmen

\.

--

-- TOC entry 3443 (class 0 OID 49197)

-- Dependencies: 217

-- Data for Name: Purchase; Type: TABLE DATA; Schema: public; Owner: postgres

--

COPY public."Purchase" (id\_purchase, purchase\_date, purchase\_volume, id\_supplier) FROM stdin;

111101 2022-04-28 272625 111191

111102 2022-04-29 70000 111193

111103 2022-04-30 51491 111192

\.

--

-- TOC entry 3444 (class 0 OID 49200)

-- Dependencies: 218

-- Data for Name: Purchase\_composition; Type: TABLE DATA; Schema: public; Owner: postgres

--

COPY public."Purchase\_composition" (id\_purchase\_composition, id\_purchase, cost, right\_stock, id\_ingredient) FROM stdin;

111004 111102 40000 500 111187

111005 111102 40000 500 111188

111001 111101 25500 400 111183

111002 111101 75000 450 111184

111003 111101 56250 445 111185

111006 111103 40000 400 111181

111007 111103 1800 40 111182

111008 111103 35000 500 111186

\.

--

-- TOC entry 3445 (class 0 OID 49203)

-- Dependencies: 219

-- Data for Name: Supplier; Type: TABLE DATA; Schema: public; Owner: postgres

--

COPY public."Supplier" (id\_supplier, supplier\_name, supplier\_city, supplier\_country) FROM stdin;

111191 Soslan MoscowCity Moscow

111192 Seraphim Pittsburg Russia

111193 Ramzan Chechnya Kavkaz

\.

--

-- TOC entry 3448 (class 0 OID 49208)

-- Dependencies: 222

-- Data for Name: Waiter; Type: TABLE DATA; Schema: public; Owner: postgres

--

COPY public."Waiter" (id\_waiter, id\_employer) FROM stdin;

111131 111126

111132 111127

111133 111128

\.

--

-- TOC entry 3455 (class 0 OID 0)

-- Dependencies: 220

-- Name: Supplier\_id\_supplier\_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres

--

SELECT pg\_catalog.setval('public."Supplier\_id\_supplier\_seq"', 1, false);

--

-- TOC entry 3456 (class 0 OID 0)

-- Dependencies: 221

-- Name: Table\_id\_table\_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres

--

SELECT pg\_catalog.setval('public."Table\_id\_table\_seq"', 1, false);

--

-- TOC entry 3457 (class 0 OID 0)

-- Dependencies: 223

-- Name: Waiter\_id\_waiter\_seq; Type: SEQUENCE SET; Schema: public; Owner: postgres

--

SELECT pg\_catalog.setval('public."Waiter\_id\_waiter\_seq"', 1, false);

--

-- TOC entry 3210 (class 2606 OID 49212)

-- Name: Cook Cook\_id\_cook\_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE public."Cook"

ADD CONSTRAINT "Cook\_id\_cook\_check" CHECK ((id\_cook > 0)) NOT VALID;

--

-- TOC entry 3229 (class 2606 OID 49214)

-- Name: Cook Cook\_id\_cook\_id\_cook1\_key; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Cook"

ADD CONSTRAINT "Cook\_id\_cook\_id\_cook1\_key" UNIQUE (id\_cook) INCLUDE (id\_cook);

--

-- TOC entry 3231 (class 2606 OID 49216)

-- Name: Cook Cook\_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Cook"

ADD CONSTRAINT "Cook\_pkey" PRIMARY KEY (id\_cook);

--

-- TOC entry 3211 (class 2606 OID 49217)

-- Name: Dish\_composition Dish\_composition\_count\_ingredient\_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE public."Dish\_composition"

ADD CONSTRAINT "Dish\_composition\_count\_ingredient\_check" CHECK ((count\_ingredient > (0)::double precision)) NOT VALID;

--

-- TOC entry 3212 (class 2606 OID 49218)

-- Name: Dish\_composition Dish\_composition\_id\_dish\_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE public."Dish\_composition"

ADD CONSTRAINT "Dish\_composition\_id\_dish\_check" CHECK ((id\_dish > 0)) NOT VALID;

--

-- TOC entry 3213 (class 2606 OID 49219)

-- Name: Dish\_composition Dish\_composition\_id\_ingredient\_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE public."Dish\_composition"

ADD CONSTRAINT "Dish\_composition\_id\_ingredient\_check" CHECK ((id\_ingredient > 0)) NOT VALID;

--

-- TOC entry 3234 (class 2606 OID 49221)

-- Name: Dish\_composition Dish\_composition\_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Dish\_composition"

ADD CONSTRAINT "Dish\_composition\_pkey" PRIMARY KEY (id\_ingredient, id\_dish);

--

-- TOC entry 3214 (class 2606 OID 49222)

-- Name: Dishes Dishes\_id\_dish\_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE public."Dishes"

ADD CONSTRAINT "Dishes\_id\_dish\_check" CHECK ((id\_dish > 0)) NOT VALID;

--

-- TOC entry 3237 (class 2606 OID 49224)

-- Name: Dishes Dishes\_id\_dish\_id\_dish1\_key; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Dishes"

ADD CONSTRAINT "Dishes\_id\_dish\_id\_dish1\_key" UNIQUE (id\_dish) INCLUDE (id\_dish);

--

-- TOC entry 3239 (class 2606 OID 49226)

-- Name: Dishes Dishes\_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Dishes"

ADD CONSTRAINT "Dishes\_pkey" PRIMARY KEY (id\_dish);

--

-- TOC entry 3215 (class 2606 OID 49227)

-- Name: Dishes Dishes\_volume\_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE public."Dishes"

ADD CONSTRAINT "Dishes\_volume\_check" CHECK ((volume > (0)::double precision)) NOT VALID;

--

-- TOC entry 3216 (class 2606 OID 49228)

-- Name: Dishes Dishes\_weight\_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE public."Dishes"

ADD CONSTRAINT "Dishes\_weight\_check" CHECK ((weight > (0)::double precision)) NOT VALID;

--

-- TOC entry 3217 (class 2606 OID 49229)

-- Name: Employer Employer\_id\_employer\_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE public."Employer"

ADD CONSTRAINT "Employer\_id\_employer\_check" CHECK ((id\_employer > 0)) NOT VALID;

--

-- TOC entry 3243 (class 2606 OID 49231)

-- Name: Employer Employer\_id\_employer\_id\_employer1\_key; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Employer"

ADD CONSTRAINT "Employer\_id\_employer\_id\_employer1\_key" UNIQUE (id\_employer) INCLUDE (id\_employer);

--

-- TOC entry 3245 (class 2606 OID 49233)

-- Name: Employer Employer\_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Employer"

ADD CONSTRAINT "Employer\_pkey" PRIMARY KEY (id\_employer);

--

-- TOC entry 3218 (class 2606 OID 49234)

-- Name: Ingredient Ingredient\_id\_ingredient\_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE public."Ingredient"

ADD CONSTRAINT "Ingredient\_id\_ingredient\_check" CHECK ((id\_ingredient > 0)) NOT VALID;

--

-- TOC entry 3248 (class 2606 OID 49236)

-- Name: Ingredient Ingredient\_id\_ingredient\_id\_ingredient1\_key; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Ingredient"

ADD CONSTRAINT "Ingredient\_id\_ingredient\_id\_ingredient1\_key" UNIQUE (id\_ingredient) INCLUDE (id\_ingredient);

--

-- TOC entry 3250 (class 2606 OID 49238)

-- Name: Ingredient Ingredient\_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Ingredient"

ADD CONSTRAINT "Ingredient\_pkey" PRIMARY KEY (id\_ingredient);

--

-- TOC entry 3219 (class 2606 OID 49239)

-- Name: Orders Orders\_id\_order\_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE public."Orders"

ADD CONSTRAINT "Orders\_id\_order\_check" CHECK ((id\_order > 0)) NOT VALID;

--

-- TOC entry 3252 (class 2606 OID 49241)

-- Name: Orders Orders\_id\_order\_id\_order1\_key; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Orders"

ADD CONSTRAINT "Orders\_id\_order\_id\_order1\_key" UNIQUE (id\_order) INCLUDE (id\_order);

--

-- TOC entry 3254 (class 2606 OID 49243)

-- Name: Orders Orders\_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Orders"

ADD CONSTRAINT "Orders\_pkey" PRIMARY KEY (id\_order);

--

-- TOC entry 3220 (class 2606 OID 49244)

-- Name: Platen Platen\_id\_table\_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE public."Platen"

ADD CONSTRAINT "Platen\_id\_table\_check" CHECK ((id\_table > 0)) NOT VALID;

--

-- TOC entry 3258 (class 2606 OID 49246)

-- Name: Platen Platen\_id\_table\_key; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Platen"

ADD CONSTRAINT "Platen\_id\_table\_key" UNIQUE (id\_table);

--

-- TOC entry 3260 (class 2606 OID 49248)

-- Name: Platen Platen\_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Platen"

ADD CONSTRAINT "Platen\_pkey" PRIMARY KEY (id\_table);

--

-- TOC entry 3221 (class 2606 OID 49249)

-- Name: Platen Platen\_table\_capacity\_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE public."Platen"

ADD CONSTRAINT "Platen\_table\_capacity\_check" CHECK ((table\_capacity > 0)) NOT VALID;

--

-- TOC entry 3222 (class 2606 OID 49250)

-- Name: Position Position\_id\_position\_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE public."Position"

ADD CONSTRAINT "Position\_id\_position\_check" CHECK ((id\_position > 0)) NOT VALID;

--

-- TOC entry 3262 (class 2606 OID 49252)

-- Name: Position Position\_id\_position\_id\_position1\_key; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Position"

ADD CONSTRAINT "Position\_id\_position\_id\_position1\_key" UNIQUE (id\_position) INCLUDE (id\_position);

--

-- TOC entry 3264 (class 2606 OID 49254)

-- Name: Position Position\_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Position"

ADD CONSTRAINT "Position\_pkey" PRIMARY KEY (id\_position);

--

-- TOC entry 3223 (class 2606 OID 49255)

-- Name: Position Position\_salary\_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE public."Position"

ADD CONSTRAINT "Position\_salary\_check" CHECK ((salary > (0)::double precision)) NOT VALID;

--

-- TOC entry 3225 (class 2606 OID 49256)

-- Name: Purchase\_composition Purchase\_composition\_id\_purchase\_composition\_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE public."Purchase\_composition"

ADD CONSTRAINT "Purchase\_composition\_id\_purchase\_composition\_check" CHECK ((id\_purchase\_composition > 0)) NOT VALID;

--

-- TOC entry 3271 (class 2606 OID 49258)

-- Name: Purchase\_composition Purchase\_composition\_id\_purchase\_composition\_id\_purchase\_co\_key; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Purchase\_composition"

ADD CONSTRAINT "Purchase\_composition\_id\_purchase\_composition\_id\_purchase\_co\_key" UNIQUE (id\_purchase\_composition) INCLUDE (id\_purchase\_composition);

--

-- TOC entry 3273 (class 2606 OID 49260)

-- Name: Purchase\_composition Purchase\_composition\_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Purchase\_composition"

ADD CONSTRAINT "Purchase\_composition\_pkey" PRIMARY KEY (id\_purchase\_composition);

--

-- TOC entry 3224 (class 2606 OID 49261)

-- Name: Purchase Purchase\_id\_purchase\_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE public."Purchase"

ADD CONSTRAINT "Purchase\_id\_purchase\_check" CHECK ((id\_purchase > 0)) NOT VALID;

--

-- TOC entry 3266 (class 2606 OID 49263)

-- Name: Purchase Purchase\_id\_purchase\_id\_purchase1\_key; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Purchase"

ADD CONSTRAINT "Purchase\_id\_purchase\_id\_purchase1\_key" UNIQUE (id\_purchase) INCLUDE (id\_purchase);

--

-- TOC entry 3268 (class 2606 OID 49265)

-- Name: Purchase Purchase\_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Purchase"

ADD CONSTRAINT "Purchase\_pkey" PRIMARY KEY (id\_purchase);

--

-- TOC entry 3226 (class 2606 OID 49266)

-- Name: Supplier Supplier\_id\_supplier\_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE public."Supplier"

ADD CONSTRAINT "Supplier\_id\_supplier\_check" CHECK ((id\_supplier > 0)) NOT VALID;

--

-- TOC entry 3277 (class 2606 OID 49268)

-- Name: Supplier Supplier\_id\_supplier\_key; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Supplier"

ADD CONSTRAINT "Supplier\_id\_supplier\_key" UNIQUE (id\_supplier);

--

-- TOC entry 3279 (class 2606 OID 49270)

-- Name: Supplier Supplier\_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Supplier"

ADD CONSTRAINT "Supplier\_pkey" PRIMARY KEY (id\_supplier);

--

-- TOC entry 3227 (class 2606 OID 49271)

-- Name: Waiter Waiter\_id\_waiter\_check; Type: CHECK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE public."Waiter"

ADD CONSTRAINT "Waiter\_id\_waiter\_check" CHECK ((id\_waiter > 0)) NOT VALID;

--

-- TOC entry 3281 (class 2606 OID 49273)

-- Name: Waiter Waiter\_id\_waiter\_id\_waiter1\_key; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Waiter"

ADD CONSTRAINT "Waiter\_id\_waiter\_id\_waiter1\_key" UNIQUE (id\_waiter) INCLUDE (id\_waiter);

--

-- TOC entry 3283 (class 2606 OID 49275)

-- Name: Waiter Waiter\_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Waiter"

ADD CONSTRAINT "Waiter\_pkey" PRIMARY KEY (id\_waiter);

--

-- TOC entry 3240 (class 1259 OID 49276)

-- Name: fki\_Cook\_id\_cook\_fkey; Type: INDEX; Schema: public; Owner: postgres

--

CREATE INDEX "fki\_Cook\_id\_cook\_fkey" ON public."Dishes" USING btree (id\_cook);

--

-- TOC entry 3235 (class 1259 OID 49277)

-- Name: fki\_Dishes\_id\_dish\_fkey; Type: INDEX; Schema: public; Owner: postgres

--

CREATE INDEX "fki\_Dishes\_id\_dish\_fkey" ON public."Dish\_composition" USING btree (id\_dish);

--

-- TOC entry 3232 (class 1259 OID 49278)

-- Name: fki\_Employer\_id\_employer\_fkey; Type: INDEX; Schema: public; Owner: postgres

--

CREATE INDEX "fki\_Employer\_id\_employer\_fkey" ON public."Cook" USING btree (id\_employer);

--

-- TOC entry 3274 (class 1259 OID 49279)

-- Name: fki\_Ingredient\_id\_ingredient\_fkey; Type: INDEX; Schema: public; Owner: postgres

--

CREATE INDEX "fki\_Ingredient\_id\_ingredient\_fkey" ON public."Purchase\_composition" USING btree (id\_ingredient);

--

-- TOC entry 3241 (class 1259 OID 49280)

-- Name: fki\_Orders\_id\_order\_fkey; Type: INDEX; Schema: public; Owner: postgres

--

CREATE INDEX "fki\_Orders\_id\_order\_fkey" ON public."Dishes" USING btree (id\_order);

--

-- TOC entry 3246 (class 1259 OID 49281)

-- Name: fki\_Position\_id\_position\_fkey; Type: INDEX; Schema: public; Owner: postgres

--

CREATE INDEX "fki\_Position\_id\_position\_fkey" ON public."Employer" USING btree (id\_position);

--

-- TOC entry 3275 (class 1259 OID 49282)

-- Name: fki\_Purchase\_id\_purchase\_fkey; Type: INDEX; Schema: public; Owner: postgres

--

CREATE INDEX "fki\_Purchase\_id\_purchase\_fkey" ON public."Purchase\_composition" USING btree (id\_purchase);

--

-- TOC entry 3269 (class 1259 OID 49283)

-- Name: fki\_Purchase\_id\_supplier\_fkey; Type: INDEX; Schema: public; Owner: postgres

--

CREATE INDEX "fki\_Purchase\_id\_supplier\_fkey" ON public."Purchase" USING btree (id\_purchase);

--

-- TOC entry 3255 (class 1259 OID 49284)

-- Name: fki\_Table\_id\_table\_fkey; Type: INDEX; Schema: public; Owner: postgres

--

CREATE INDEX "fki\_Table\_id\_table\_fkey" ON public."Orders" USING btree (id\_table);

--

-- TOC entry 3256 (class 1259 OID 49285)

-- Name: fki\_Waiter\_id\_waiter\_fkey; Type: INDEX; Schema: public; Owner: postgres

--

CREATE INDEX "fki\_Waiter\_id\_waiter\_fkey" ON public."Orders" USING btree (id\_waiter);

--

-- TOC entry 3287 (class 2606 OID 49286)

-- Name: Dishes Cook\_id\_cook\_fkey; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Dishes"

ADD CONSTRAINT "Cook\_id\_cook\_fkey" FOREIGN KEY (id\_cook) REFERENCES public."Cook"(id\_cook) ON UPDATE RESTRICT ON DELETE RESTRICT DEFERRABLE INITIALLY DEFERRED;

--

-- TOC entry 3285 (class 2606 OID 49291)

-- Name: Dish\_composition Dishes\_id\_dish\_fkey; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Dish\_composition"

ADD CONSTRAINT "Dishes\_id\_dish\_fkey" FOREIGN KEY (id\_dish) REFERENCES public."Dishes"(id\_dish) ON UPDATE RESTRICT ON DELETE RESTRICT DEFERRABLE INITIALLY DEFERRED;

--

-- TOC entry 3284 (class 2606 OID 49296)

-- Name: Cook Employer\_id\_employer\_fkey; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Cook"

ADD CONSTRAINT "Employer\_id\_employer\_fkey" FOREIGN KEY (id\_employer) REFERENCES public."Employer"(id\_employer) ON UPDATE RESTRICT ON DELETE RESTRICT DEFERRABLE INITIALLY DEFERRED;

--

-- TOC entry 3295 (class 2606 OID 49301)

-- Name: Waiter Employer\_id\_employer\_fkey; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Waiter"

ADD CONSTRAINT "Employer\_id\_employer\_fkey" FOREIGN KEY (id\_employer) REFERENCES public."Employer"(id\_employer) ON UPDATE RESTRICT ON DELETE RESTRICT DEFERRABLE INITIALLY DEFERRED;

--

-- TOC entry 3286 (class 2606 OID 49306)

-- Name: Dish\_composition Ingredient\_id\_ingredient\_fkey; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Dish\_composition"

ADD CONSTRAINT "Ingredient\_id\_ingredient\_fkey" FOREIGN KEY (id\_ingredient) REFERENCES public."Ingredient"(id\_ingredient) ON UPDATE RESTRICT ON DELETE RESTRICT DEFERRABLE INITIALLY DEFERRED;

--

-- TOC entry 3293 (class 2606 OID 49311)

-- Name: Purchase\_composition Ingredient\_id\_ingredient\_fkey; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Purchase\_composition"

ADD CONSTRAINT "Ingredient\_id\_ingredient\_fkey" FOREIGN KEY (id\_ingredient) REFERENCES public."Ingredient"(id\_ingredient) ON UPDATE RESTRICT ON DELETE RESTRICT DEFERRABLE INITIALLY DEFERRED;

--

-- TOC entry 3288 (class 2606 OID 49316)

-- Name: Dishes Orders\_id\_order\_fkey; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Dishes"

ADD CONSTRAINT "Orders\_id\_order\_fkey" FOREIGN KEY (id\_order) REFERENCES public."Orders"(id\_order) ON UPDATE RESTRICT ON DELETE RESTRICT DEFERRABLE INITIALLY DEFERRED;

--

-- TOC entry 3290 (class 2606 OID 49321)

-- Name: Orders Platen\_id\_table\_fkey; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Orders"

ADD CONSTRAINT "Platen\_id\_table\_fkey" FOREIGN KEY (id\_table) REFERENCES public."Platen"(id\_table) ON UPDATE RESTRICT ON DELETE RESTRICT DEFERRABLE INITIALLY DEFERRED;

--

-- TOC entry 3289 (class 2606 OID 49326)

-- Name: Employer Position\_id\_position\_fkey; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Employer"

ADD CONSTRAINT "Position\_id\_position\_fkey" FOREIGN KEY (id\_position) REFERENCES public."Position"(id\_position) ON UPDATE RESTRICT ON DELETE RESTRICT DEFERRABLE INITIALLY DEFERRED;

--

-- TOC entry 3294 (class 2606 OID 49331)

-- Name: Purchase\_composition Purchase\_id\_purchase\_fkey; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Purchase\_composition"

ADD CONSTRAINT "Purchase\_id\_purchase\_fkey" FOREIGN KEY (id\_purchase) REFERENCES public."Purchase"(id\_purchase) ON UPDATE RESTRICT ON DELETE RESTRICT DEFERRABLE INITIALLY DEFERRED;

--

-- TOC entry 3292 (class 2606 OID 49336)

-- Name: Purchase Purchase\_id\_supplier\_fkey; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Purchase"

ADD CONSTRAINT "Purchase\_id\_supplier\_fkey" FOREIGN KEY (id\_purchase) REFERENCES public."Purchase"(id\_purchase) ON UPDATE RESTRICT ON DELETE RESTRICT DEFERRABLE INITIALLY DEFERRED;

--

-- TOC entry 3291 (class 2606 OID 49341)

-- Name: Orders Waiter\_id\_waiter\_fkey; Type: FK CONSTRAINT; Schema: public; Owner: postgres

--

ALTER TABLE ONLY public."Orders"

ADD CONSTRAINT "Waiter\_id\_waiter\_fkey" FOREIGN KEY (id\_waiter) REFERENCES public."Waiter"(id\_waiter) ON UPDATE RESTRICT ON DELETE RESTRICT DEFERRABLE INITIALLY DEFERRED;

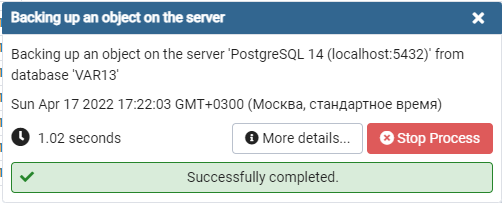
-- Completed on 2022-04-17 17:22:04

--

-- PostgreSQL database dump complete

--

Созданием бэкапа и его восстановление:



Изображение выглядит как текст

Автоматически созданное описание

# Вывод:

PgAdmin – достаточно удобная программа для создания баз данных PostgreSQL, обладающая приемлемо интуитивным интерфейсом, разобраться с которым новичку не доставит великих проблем. Но, к сожалению, программа обладает неявными ограничениями или даже багами, с которыми новичку самостоятельно справиться будет гораздо тяжелее. К примеру, с чем столкнулся Я: невозможность задать ограничение для столбца, если его имя содержит символы верхнего регистра, необходимость использовать скрипты SELECT, INSERT, DELETE, etc., так как программа не воспринимает стандартный метод ввода SQL.