

Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение высшего
образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Отчет

по лабораторной работе №3 «Создание БД в СУБД PostgreSQL. Резервное копирование
и восстановление БД»

по дисциплине **«Проектирование и реализация баз данных»**

Автор: Зеленин Денис Сергеевич

Факультет: ИКТ

Группа: K3241

Преподаватель: Говорова М.М.



Санкт-Петербург 2023

Оглавление

Цель работы	3
Практическое задание	3
Вариант 14. БД «Служба заказа такси»	3
Выполнение	5
Вывод	28

Цель работы

Овладеть практическими навыками установки СУБД PostgreSQL и создания базы данных в pgadmin 4, создания таблиц базы данных PostgreSQL 1X, назначение ограничений на данные, заполнения таблиц рабочими данными, резервного копирования и восстановления БД.

Практическое задание

1. Установить СУБД PostgreSQL, pgAdmin4.
2. Создать базу данных с использованием pgAdmin4.
3. Создать схему, таблицы в базе данных.
4. Установить ограничения на данные: Primary Key, Unique, Check, Foreign Key.
5. Заполнить таблицы БД рабочими данными.
6. Создать резервную копию БД.
7. Восстановить резервную копию БД.

Вариант 14. БД «Служба заказа такси»

Описание предметной области: Система должна фиксировать все вызовы такси и распределять их между водителями.

Каждому водителю ежедневно начисляется заработная плата в зависимости от количества вызовов и их тарифа (50% от заработанной им суммы). Автомобили могут быть собственностью компании или таксиста.

Заказ принимает дежурный администратор и передает его водителю. В заказе фиксируется тип оплаты – наличными или онлайн. Если заказ оплачивается онлайн, то в системе хранится привязка к карте, с которой была совершена оплата.

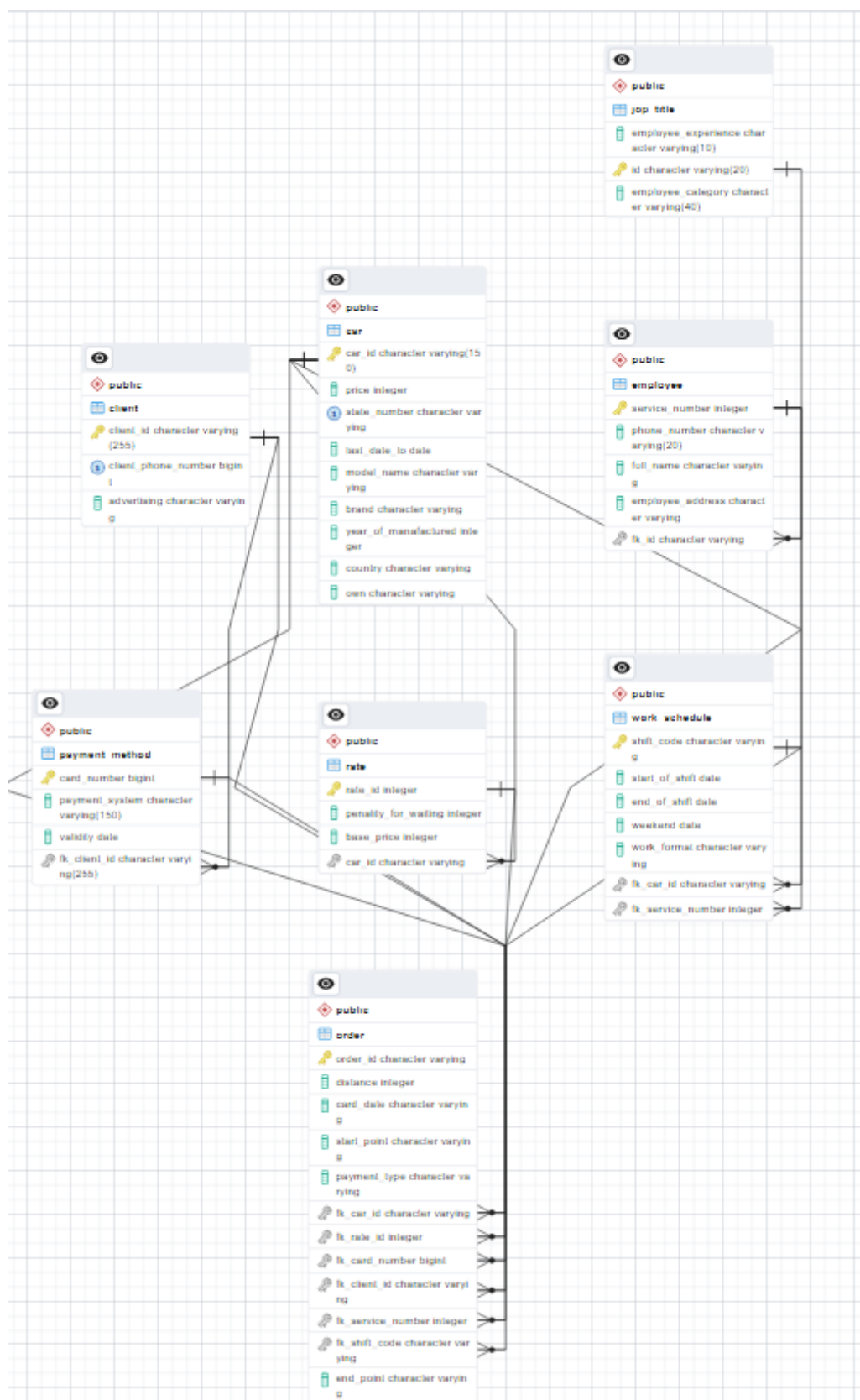
В системе необходимо хранить график работы водителей.

Ежедневно действуют базовые тарифы на тип предоставляемых авто, но в зависимости от времени суток и ситуации на дорогах, цена может корректироваться.

БД должна содержать следующий минимальный набор сведений: Код сотрудника. ФИО сотрудника. Адрес сотрудника. № телефона сотрудника. Паспортные данные сотрудника. Должность сотрудника. Категория сотрудника. Наименование модели и марки автомобиля. Технические характеристики. Стран-производитель.

Стоимость. Код тарифа. Наименование тарифа. Цена за километр. Код автомобиля. Госномер автомобиля. Год выпуска. Пробег. Дата последнего ТО. Дата вызова. Время посадки пассажира. Время высадки пассажира. Номер телефона пассажира. Откуда. Куда. Расстояние. Штраф за время ожидания (в минутах). Оплата (онлайн (при заказе) или наличными). Рекламация клиента на вызов.

Выполнение



```
2. -- PostgreSQL database dump
3. --
4.
5. -- Dumped from database version 14.9
6. -- Dumped by pg_dump version 14.9
7.
8. -- Started on 2023-10-27 15:29:28
9.
10. SET statement_timeout = 0;
11. SET lock_timeout = 0;
12. SET idle_in_transaction_session_timeout = 0;
13. SET client_encoding = 'UTF8';
14. SET standard_conforming_strings = on;
15. SELECT pg_catalog.set_config('search_path', '', false);
16. SET check_function_bodies = false;
17. SET xmloption = content;
18. SET client_min_messages = warning;
19. SET row_security = off;
20.
21. --
22. -- TOC entry 221 (class 1255 OID 16549)
23. -- Name: set_or_order_id(); Type: FUNCTION; Schema: public; Owner: postgres
24. --
25.
26. CREATE FUNCTION public.set_or_order_id() RETURNS trigger
27.   LANGUAGE plpgsql
28.   AS $$
29. BEGIN
30.   NEW.order_id = 'or' || NEXTVAL('or_id_sequence');
31.   RETURN NEW;
32. END;
33. $$;
34.
35.
36. ALTER FUNCTION public.set_or_order_id() OWNER TO postgres;
37.
38. SET default_tablespace = '';
39.
40. SET default_table_access_method = heap;
41.
42. --
43. -- TOC entry 212 (class 1259 OID 16428)
44. -- Name: car; Type: TABLE; Schema: public; Owner: postgres
45. --
46.
47. CREATE TABLE public.car (
48.   car_id character varying(150) NOT NULL,
49.   price integer NOT NULL,
```

```

50. state_number character varying NOT NULL,
51. last_date_to date NOT NULL,
52. model_name character varying NOT NULL,
53. brand character varying,
54. year_of_manufactured integer NOT NULL,
55. country character varying,
56. own character varying NOT NULL,
57. CONSTRAINT check_manufactured_year CHECK ((year_of_manufactured >= 2010)),
58. CONSTRAINT check_price_positive CHECK ((price > 0))
59. );
60.
61.
62. ALTER TABLE public.car OWNER TO postgres;
63.
64. --
65. -- TOC entry 209 (class 1259 OID 16400)
66. -- Name: client; Type: TABLE; Schema: public; Owner: postgres
67. --
68.
69. CREATE TABLE public.client (
70. client_id character varying(255) NOT NULL,
71. client_phone_number bigint NOT NULL,
72. advertising character varying
73. );
74.
75.
76. ALTER TABLE public.client OWNER TO postgres;
77.
78. --
79. -- TOC entry 214 (class 1259 OID 16447)
80. -- Name: employee; Type: TABLE; Schema: public; Owner: postgres
81. --
82.
83. CREATE TABLE public.employee (
84. service_number integer NOT NULL,
85. phone_number character varying(20),
86. full_name character varying NOT NULL,
87. employee_address character varying NOT NULL,
88. fk_id character varying,
89. CONSTRAINT check_phone_number_format CHECK (((phone_number)::text ~
    '^[0-9]{10}$'::text))
90. );
91.
92.
93. ALTER TABLE public.employee OWNER TO postgres;
94.
95. --
96. -- TOC entry 213 (class 1259 OID 16442)

```

```

97. -- Name: jop_title; Type: TABLE; Schema: public; Owner: postgres
98. --
99.
100. CREATE TABLE public.jop_title (
101.     employee_experience character varying(10),
102.     id character varying(20) NOT NULL,
103.     employee_category character varying(40),
104.     CONSTRAINT check_age_format CHECK (((employee_experience)::text ~
        '^\\d+\\s+(лет|года)$'::text))
105. );
106.
107.
108. ALTER TABLE public.jop_title OWNER TO postgres;
109.
110. --
111. -- TOC entry 220 (class 1259 OID 16548)
112. -- Name: or_id_sequence; Type: SEQUENCE; Schema: public; Owner: postgres
113. --
114.
115. CREATE SEQUENCE public.or_id_sequence
116.     START WITH 1
117.     INCREMENT BY 1
118.     NO MINVALUE
119.     NO MAXVALUE
120.     CACHE 1;
121.
122.
123. ALTER TABLE public.or_id_sequence OWNER TO postgres;
124.
125. --
126. -- TOC entry 216 (class 1259 OID 16476)
127. -- Name: order_; Type: TABLE; Schema: public; Owner: postgres
128. --
129.
130. CREATE TABLE public.order_ (
131.     order_id character varying NOT NULL,
132.     distance integer DEFAULT 0 NOT NULL,
133.     card_date character varying NOT NULL,
134.     start_point character varying NOT NULL,
135.     payment_type character varying NOT NULL,
136.     fk_car_id character varying,
137.     fk_rate_id integer,
138.     fk_card_number bigint,
139.     fk_client_id character varying,
140.     fk_service_number integer,
141.     fk_shift_code character varying,
142.     end_point character varying NOT NULL
143. );

```



```
144.
145.
146. ALTER TABLE public.order_ OWNER TO postgres;
147.
148. --
149. -- TOC entry 219 (class 1259 OID 16547)
150. -- Name: order_id; Type: SEQUENCE; Schema: public; Owner: postgres
151. --
152.
153. CREATE SEQUENCE public.order_id
154.     START WITH 1
155.     INCREMENT BY 1
156.     NO MINVALUE
157.     NO MAXVALUE
158.     CACHE 1;
159.
160.
161. ALTER TABLE public.order_id OWNER TO postgres;
162.
163. --
164. -- TOC entry 210 (class 1259 OID 16407)
165. -- Name: payment_method; Type: TABLE; Schema: public; Owner: postgres
166. --
167.
168. CREATE TABLE public.payment_method (
169.     card_number bigint NOT NULL,
170.     payment_system character varying(150) NOT NULL,
171.     validity date NOT NULL,
172.     fk_client_id character varying(255),
173.     CONSTRAINT check_payment_system CHECK (((payment_system)::text = ANY
        ((ARRAY['visa'::character varying, 'masterCard'::character varying, 'мир'::character
        varying])::text[])))
174. );
175.
176.
177. ALTER TABLE public.payment_method OWNER TO postgres;
178.
179. --
180. -- TOC entry 218 (class 1259 OID 16543)
181. -- Name: rate_id; Type: SEQUENCE; Schema: public; Owner: postgres
182. --
183.
184. CREATE SEQUENCE public.rate_id
185.     START WITH 1
186.     INCREMENT BY 1
187.     NO MINVALUE
188.     NO MAXVALUE
189.     CACHE 1;
```

```

190.
191.
192. ALTER TABLE public.rate_id OWNER TO postgres;
193.
194. --
195. -- TOC entry 211 (class 1259 OID 16423)
196. -- Name: rate; Type: TABLE; Schema: public; Owner: postgres
197. --
198.
199. CREATE TABLE public.rate (
200.     rate_id integer DEFAULT nextval('public.rate_id'::regclass) NOT NULL,
201.     penalty_for_waiting integer DEFAULT 0,
202.     base_price integer NOT NULL,
203.     car_id character varying
204. );
205.
206.
207. ALTER TABLE public.rate OWNER TO postgres;
208.
209. --
210. -- TOC entry 217 (class 1259 OID 16541)
211. -- Name: shift_code_sequence; Type: SEQUENCE; Schema: public; Owner: postgres
212. --
213.
214. CREATE SEQUENCE public.shift_code_sequence
215.     START WITH 1
216.     INCREMENT BY 1
217.     NO MINVALUE
218.     NO MAXVALUE
219.     CACHE 1;
220.
221.
222. ALTER TABLE public.shift_code_sequence OWNER TO postgres;
223.
224. --
225. -- TOC entry 215 (class 1259 OID 16459)
226. -- Name: work_schedule; Type: TABLE; Schema: public; Owner: postgres
227. --
228.
229. CREATE TABLE public.work_schedule (
230.     shift_code character varying DEFAULT nextval('public.shift_code_sequence'::regclass)
231.     NOT NULL,
232.     start_of_shift date NOT NULL,
233.     end_of_shift date NOT NULL,
234.     weekend date,
235.     work_format character varying,
236.     fk_car_id character varying NOT NULL,
237.     fk_service_number integer NOT NULL

```

```

237. );
238.
239.
240. ALTER TABLE public.work_schedule OWNER TO postgres;
241.
242. --
243. -- TOC entry 3381 (class 0 OID 16428)
244. -- Dependencies: 212
245. -- Data for Name: car; Type: TABLE DATA; Schema: public; Owner: postgres
246. --
247.
248. COPY public.car (car_id, price, state_number, last_date_to, model_name, brand,
    year_of_manufactured, country, own) FROM stdin;
249. C1 25000 AB123CD 2023-10-15 Camry Toyota 2019 Japan taxi
250. C2 28000 XY789ZA 2023-09-25 Civic Honda 2018 Japan taxi
251. C3 30000 ZZ456XY 2023-11-05 Accord Honda 2020 Japan taxi
252. C4 35000 JK321LM 2023-08-12 3 Series BMW 2017 Germany taxi
253. C5 40000 MM111NN 2023-12-01 A4 Audi 2021 Germany taxi
254. C6 27000 PL777OP 2023-07-18 Fusion Ford 2016 USA taxi
255. C7 32000 AA222BB 2023-06-20 Altima Nissan 2019 Japan частная
256. C8 29000 CC333DD 2023-09-10 Optima Kia 2017 South Korea частная
257. C9 26000 DD444EE 2023-08-05 Impreza Subaru 2020 Japan частная
258. C10 31000 EE555FF 2023-10-30 Civic Honda 2018 Japan taxi
259. \.
260.
261.
262. --
263. -- TOC entry 3378 (class 0 OID 16400)
264. -- Dependencies: 209
265. -- Data for Name: client; Type: TABLE DATA; Schema: public; Owner: postgres
266. --
267.
268. COPY public.client (client_id, client_phone_number, advertising) FROM stdin;
269. C1 1234567890 Проблемы с оплатой
270. C2 1987654321
271. C3 1122334455
272. C4 9876543210
273. C5 6543210987 Потерянный предмет в такси
274. C6 5435350543
275. C7 89817656505
276. C8 8539298439
277. C9 4392424893
278. C10 9434939204
279. \.
280.
281.
282. --
283. -- TOC entry 3383 (class 0 OID 16447)

```

```

284. -- Dependencies: 214
285. -- Data for Name: employee; Type: TABLE DATA; Schema: public; Owner: postgres
286. --
287.
288. COPY public.employee (service_number, phone_number, full_name, employee_address,
    fk_id) FROM stdin;
289. 3681 1234567890 Иванов Иван Иванович ул. Пушкина, 1 11
290. 3682 9876543210 Петров Петр Петрович ул. Лермонтова, 2 12
291. 3683 1112233445 Сидоров Сидор Сидорович ул. Толстого, 3 4
292. 3684 5556667778 Кузнецова Анна Владимировна ул. Гоголя, 4 1
293. 3685 4443332221 Павлов Павел Павлович ул. Чехова, 5 6
294. 3686 7778889990 Михайлов Михаил Михайлович ул. Достоевского, 6 2
295. 3687 9990001112 Орлова Ольга Анатольевна ул. Тургенева, 7 5
296. 3688 1213141516 Григорьев Григорий Григорьевич ул. Шекспира, 8 11
297. 3689 1718192021 Николаев Николай Николаевич ул. Достоевского, 9 12
298. 36810 2324252627 Федоров Федор Федорович ул. Пастернака, 10 2
299. \.
300.
301.
302. --
303. -- TOC entry 3382 (class 0 OID 16442)
304. -- Dependencies: 213
305. -- Data for Name: job_title; Type: TABLE DATA; Schema: public; Owner: postgres
306. --
307.
308. COPY public.job_title (employee_experience, id, employee_category) FROM stdin;
309. 2 года 10 Сервисный работник
310. 3 лет 2 Старший диспетчер
311. 4 лет 3 Менеджер
312. 2 лет 4 Администратор
313. 4 лет 5 Старший администратор
314. 2 лет 6 Технический специалист
315. 4 лет 7 Инструктор
316. 6 лет 9 Старший менеджер
317. 5 лет 1 Водитель 2 категории
318. 1 года 8 Механик
319. 1 года 11 Водитель 1 категории
320. 15 лет 12 Водитель 3 категории
321. \.
322.
323.
324. --
325. -- TOC entry 3385 (class 0 OID 16476)
326. -- Dependencies: 216
327. -- Data for Name: order_; Type: TABLE DATA; Schema: public; Owner: postgres
328. --
329.

```

```

330. COPY public.order_ (order_id, distance, card_date, start_point, payment_type, fk_car_id,
    fk_rate_id, fk_card_number, fk_client_id, fk_service_number, fk_shift_code, end_point)
    FROM stdin;
331. or11 5 2023-10-26 Улица Ленина, 123 Наличные C1 1
    4156025011921310 C5 3681 1 Улица Гагарина, 45
332. or12 4 2023-10-27 Улица Пушкина, 67 Карта C1 1
    4293027718376115 C10 3681 1 Улица Чехова, 89
333. or13 7 2023-10-28 Улица Толстого, 21 Наличные C3 3
    4330943363667464 C7 3684 3 Улица Достоевского, 14
334. or14 5 2023-10-29 Улица Герцена, 37 Карта C4 4
    4449178559155484 C8 3688 5 Улица Лермонтова, 56
335. or15 7 2023-10-30 Улица Пастернака, 98 Наличные C5 5
    4465991831387584 C4 3689 4 Улица Булгакова, 72
336. or16 4 2023-10-31 Улица Шолохова, 5 Карта C6 6
    4576580834247303 C9 3685 6 Улица Солженицына, 19
337. or17 4 2023-11-01 Улица Ахматовой, 33 Наличные C1 1
    4621965225399430 C6 3681 7 Улица Бродского, 78
338. or18 2 2023-11-02 Улица Пастернака, 10 Карта C7 7
    4649963217198504 C3 36810 8 Улица Маяковского, 55
339. or19 7 2023-11-03 Улица Гоголя, 29 Наличные C8 8
    4804374615104832 C1 3681 9 Улица Чайковского, 17
340. or20 4 2023-11-04 Улица Дюма, 94 Карта C10 9 4863642573169460
    C2 3687 10 Улица Гюго, 7
341. \.
342.
343.
344. --
345. -- TOC entry 3379 (class 0 OID 16407)
346. -- Dependencies: 210
347. -- Data for Name: payment_method; Type: TABLE DATA; Schema: public; Owner: postgres
348. --
349.
350. COPY public.payment_method (card_number, payment_system, validity, fk_client_id)
    FROM stdin;
351. 4804374615104832 visa 2026-05-01 C1
352. 4863642573169460 visa 2026-09-01 C2
353. 4649963217198504 visa 2028-01-01 C3
354. 4465991831387584 visa 2029-04-01 C4
355. 4156025011921310 visa 2024-04-01 C5
356. 4621965225399430 visa 2024-11-01 C6
357. 4330943363667464 visa 2024-06-01 C7
358. 4449178559155484 visa 2025-12-01 C8
359. 4576580834247303 visa 2026-08-01 C9
360. 4293027718376115 visa 2026-02-01 C10
361. \.
362.
363.
364. --

```

```

365. -- TOC entry 3380 (class 0 OID 16423)
366. -- Dependencies: 211
367. -- Data for Name: rate; Type: TABLE DATA; Schema: public; Owner: postgres
368. --
369.
370. COPY public.rate (rate_id, penalty_for_waiting, base_price, car_id) FROM stdin;
371. 1    60    400    C1
372. 2     0    550    C2
373. 3     0    480    C3
374. 4     0    490    C4
375. 5    110    510    C5
376. 6    140    520    C6
377. 7    160    530    C7
378. 8    125    495    C8
379. 9    135    505    C9
380. 10   145    515    C10
381. \.
382.
383.
384. --
385. -- TOC entry 3384 (class 0 OID 16459)
386. -- Dependencies: 215
387. -- Data for Name: work_schedule; Type: TABLE DATA; Schema: public; Owner: postgres
388. --
389.
390. COPY public.work_schedule (shift_code, start_of_shift, end_of_shift, weekend,
    work_format, fk_car_id, fk_service_number) FROM stdin;
391. 1    2023-10-26    2023-10-26    \N    2/2    C1    3681
392. 2    2023-10-27    2023-10-27    \N    2/2    C1    3681
393. 5    2023-10-30    2023-10-30    \N    2/2    C2    3686
394. 6    2023-10-30    2023-11-01    \N    5/2    C10   3687
395. 7    2023-11-01    2023-11-01    \N    2/2    C4    3688
396. 8    2023-11-02    2023-11-02    \N    5/2    C5    3689
397. 9    2023-11-02    2023-11-03    \N    2/2    C7    36810
398. 10   2023-11-03    2023-11-05    2023-11-03    5/2    C8    3683
399. 3    2023-10-28    2023-10-28    2023-10-29    2/2    C3    3684
400. 4    2023-10-28    2023-10-29    2023-10-30    5/2    C6    3685
401. \.
402.
403.
404. --
405. -- TOC entry 3395 (class 0 OID 0)
406. -- Dependencies: 220
407. -- Name: or_id_sequence; Type: SEQUENCE SET; Schema: public; Owner: postgres
408. --
409.
410. SELECT pg_catalog.setval('public.or_id_sequence', 20, true);
411.

```

```

412.
413. --
414. -- TOC entry 3396 (class 0 OID 0)
415. -- Dependencies: 219
416. -- Name: order_id; Type: SEQUENCE SET; Schema: public; Owner: postgres
417. --
418.
419. SELECT pg_catalog.setval('public.order_id', 1, false);
420.
421.
422. --
423. -- TOC entry 3397 (class 0 OID 0)
424. -- Dependencies: 218
425. -- Name: rate_id; Type: SEQUENCE SET; Schema: public; Owner: postgres
426. --
427.
428. SELECT pg_catalog.setval('public.rate_id', 10, true);
429.
430.
431. --
432. -- TOC entry 3398 (class 0 OID 0)
433. -- Dependencies: 217
434. -- Name: shift_code_sequence; Type: SEQUENCE SET; Schema: public; Owner: postgres
435. --
436.
437. SELECT pg_catalog.setval('public.shift_code_sequence', 10, true);
438.
439.
440. --
441. -- TOC entry 3214 (class 2606 OID 16434)
442. -- Name: car Car_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres
443. --
444.
445. ALTER TABLE ONLY public.car
446.     ADD CONSTRAINT "Car_pkey" PRIMARY KEY (car_id);
447.
448.
449. --
450. -- TOC entry 3220 (class 2606 OID 16453)
451. -- Name: employee Employee_pkey; Type: CONSTRAINT; Schema: public; Owner:
postgres
452. --
453.
454. ALTER TABLE ONLY public.employee
455.     ADD CONSTRAINT "Employee_pkey" PRIMARY KEY (service_number);
456.
457.
458. --

```

```

459. -- TOC entry 3218 (class 2606 OID 16446)
460. -- Name: jop_title Jop_title_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres
461. --
462.
463. ALTER TABLE ONLY public.jop_title
464.     ADD CONSTRAINT "Jop_title_pkey" PRIMARY KEY (id);
465.
466.
467. --
468. -- TOC entry 3212 (class 2606 OID 16427)
469. -- Name: rate Rate_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres
470. --
471.
472. ALTER TABLE ONLY public.rate
473.     ADD CONSTRAINT "Rate_pkey" PRIMARY KEY (rate_id);
474.
475.
476. --
477. -- TOC entry 3224 (class 2606 OID 16465)
478. -- Name: work_schedule Work schedule_pkey; Type: CONSTRAINT; Schema: public;
    Owner: postgres
479. --
480.
481. ALTER TABLE ONLY public.work_schedule
482.     ADD CONSTRAINT "Work schedule_pkey" PRIMARY KEY (shift_code);
483.
484.
485. --
486. -- TOC entry 3206 (class 2606 OID 16579)
487. -- Name: client client_phone_number; Type: CONSTRAINT; Schema: public; Owner:
    postgres
488. --
489.
490. ALTER TABLE ONLY public.client
491.     ADD CONSTRAINT client_phone_number UNIQUE (client_phone_number) INCLUDE
    (client_phone_number);
492.
493.
494. --
495. -- TOC entry 3226 (class 2606 OID 16482)
496. -- Name: order_order_pkey; Type: CONSTRAINT; Schema: public; Owner: postgres
497. --
498.
499. ALTER TABLE ONLY public.order_
500.     ADD CONSTRAINT order_pkey PRIMARY KEY (order_id);
501.
502.
503. --

```



```

504. -- TOC entry 3210 (class 2606 OID 16530)
505. -- Name: payment_method pk_card_number; Type: CONSTRAINT; Schema: public;
      Owner: postgres
506. --
507.
508. ALTER TABLE ONLY public.payment_method
509.     ADD CONSTRAINT pk_card_number PRIMARY KEY (card_number);
510.
511.
512. --
513. -- TOC entry 3208 (class 2606 OID 16406)
514. -- Name: client pk_client_id; Type: CONSTRAINT; Schema: public; Owner: postgres
515. --
516.
517. ALTER TABLE ONLY public.client
518.     ADD CONSTRAINT pk_client_id PRIMARY KEY (client_id);
519.
520.
521. --
522. -- TOC entry 3222 (class 2606 OID 16576)
523. -- Name: employee uk_service_number; Type: CONSTRAINT; Schema: public; Owner:
      postgres
524. --
525.
526. ALTER TABLE ONLY public.employee
527.     ADD CONSTRAINT uk_service_number UNIQUE (service_number);
528.
529.
530. --
531. -- TOC entry 3216 (class 2606 OID 16565)
532. -- Name: car uk_state_number; Type: CONSTRAINT; Schema: public; Owner: postgres
533. --
534.
535. ALTER TABLE ONLY public.car
536.     ADD CONSTRAINT uk_state_number UNIQUE (state_number);
537.
538.
539. --
540. -- TOC entry 3238 (class 2620 OID 16550)
541. -- Name: order_set_or_order_id_trigger; Type: TRIGGER; Schema: public; Owner:
      postgres
542. --
543.
544. CREATE TRIGGER set_or_order_id_trigger BEFORE INSERT ON public.order_ FOR EACH
      ROW EXECUTE FUNCTION public.set_or_order_id();
545.
546.
547. --

```

```

548. -- TOC entry 3228 (class 2606 OID 16437)
549. -- Name: rate fk_car_id; Type: FK CONSTRAINT; Schema: public; Owner: postgres
550. --
551.
552. ALTER TABLE ONLY public.rate
553.     ADD CONSTRAINT fk_car_id FOREIGN KEY (car_id) REFERENCES public.car(car_id) NOT
    VALID;
554.
555.
556. --
557. -- TOC entry 3230 (class 2606 OID 16466)
558. -- Name: work_schedule fk_car_id; Type: FK CONSTRAINT; Schema: public; Owner:
    postgres
559. --
560.
561. ALTER TABLE ONLY public.work_schedule
562.     ADD CONSTRAINT fk_car_id FOREIGN KEY (fk_car_id) REFERENCES public.car(car_id)
    NOT VALID;
563.
564.
565. --
566. -- TOC entry 3232 (class 2606 OID 16483)
567. -- Name: order_fk_car_id; Type: FK CONSTRAINT; Schema: public; Owner: postgres
568. --
569.
570. ALTER TABLE ONLY public.order_
571.     ADD CONSTRAINT fk_car_id FOREIGN KEY (fk_car_id) REFERENCES public.car(car_id)
    NOT VALID;
572.
573.
574. --
575. -- TOC entry 3237 (class 2606 OID 16551)
576. -- Name: order_fk_card_number; Type: FK CONSTRAINT; Schema: public; Owner:
    postgres
577. --
578.
579. ALTER TABLE ONLY public.order_
580.     ADD CONSTRAINT fk_card_number FOREIGN KEY (fk_card_number) REFERENCES
    public.payment_method(card_number) NOT VALID;
581.
582.
583. --
584. -- TOC entry 3227 (class 2606 OID 16418)
585. -- Name: payment_method fk_client_id; Type: FK CONSTRAINT; Schema: public; Owner:
    postgres
586. --
587.
588. ALTER TABLE ONLY public.payment_method

```

```

589.      ADD CONSTRAINT fk_client_id FOREIGN KEY (fk_client_id) REFERENCES
        public.client(client_id) NOT VALID;
590.
591.
592.  --
593.  -- TOC entry 3234 (class 2606 OID 16498)
594.  -- Name: order_ fk_client_id; Type: FK CONSTRAINT; Schema: public; Owner: postgres
595.  --
596.
597.  ALTER TABLE ONLY public.order_
598.      ADD CONSTRAINT fk_client_id FOREIGN KEY (fk_client_id) REFERENCES
        public.client(client_id) NOT VALID;
599.
600.
601.  --
602.  -- TOC entry 3229 (class 2606 OID 16454)
603.  -- Name: employee fk_id; Type: FK CONSTRAINT; Schema: public; Owner: postgres
604.  --
605.
606.  ALTER TABLE ONLY public.employee
607.      ADD CONSTRAINT fk_id FOREIGN KEY (fk_id) REFERENCES public.jop_title(id) NOT
        VALID;
608.
609.
610.  --
611.  -- TOC entry 3233 (class 2606 OID 16488)
612.  -- Name: order_ fk_rate_id; Type: FK CONSTRAINT; Schema: public; Owner: postgres
613.  --
614.
615.  ALTER TABLE ONLY public.order_
616.      ADD CONSTRAINT fk_rate_id FOREIGN KEY (fk_rate_id) REFERENCES
        public.rate(rate_id) NOT VALID;
617.
618.
619.  --
620.  -- TOC entry 3231 (class 2606 OID 16471)
621.  -- Name: work_schedule fk_service_number; Type: FK CONSTRAINT; Schema: public;
        Owner: postgres
622.  --
623.
624.  ALTER TABLE ONLY public.work_schedule
625.      ADD CONSTRAINT fk_service_number FOREIGN KEY (fk_service_number) REFERENCES
        public.employee(service_number) NOT VALID;
626.
627.
628.  --
629.  -- TOC entry 3236 (class 2606 OID 16508)

```

```

630.  -- Name: order_fk_service_number; Type: FK CONSTRAINT; Schema: public; Owner:
      postgres
631.  --
632.
633.  ALTER TABLE ONLY public.order_
634.      ADD CONSTRAINT fk_service_number FOREIGN KEY (fk_service_number) REFERENCES
      public.employee(service_number) NOT VALID;
635.
636.
637.  --
638.  -- TOC entry 3235 (class 2606 OID 16503)
639.  -- Name: order_fk_shift_code; Type: FK CONSTRAINT; Schema: public; Owner: postgres
640.  --
641.
642.  ALTER TABLE ONLY public.order_
643.      ADD CONSTRAINT fk_shift_code FOREIGN KEY (fk_shift_code) REFERENCES
      public.work_schedule(shift_code) NOT VALID;
644.
645.
646.  -- Completed on 2023-10-27 15:29:29
647.
648.  --
649.  -- PostgreSQL database dump complete
650.  --
651.
652.

```

Вывод

В процессе выполнения лабораторной работы я приобрел навыки создания, заполнения и восстановления баз данных в PostgreSQL, используя программу pgAdmin4. В рамках задания была создана база данных с соответствующими таблицами, настроены необходимые связи и ограничения. Таблицы были заполнены актуальными данными, обеспечивая полноту информации.