

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 № 3
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
Кривошапкина А.С.

Санкт-Петербург 2021 г

1. Цель работы

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

2. 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.

3. Выполнение

- I. **Наименование БД: «Таксопарк»**
- II. **Схема логической модели базы данных, сгенерированная в Generate ERD**

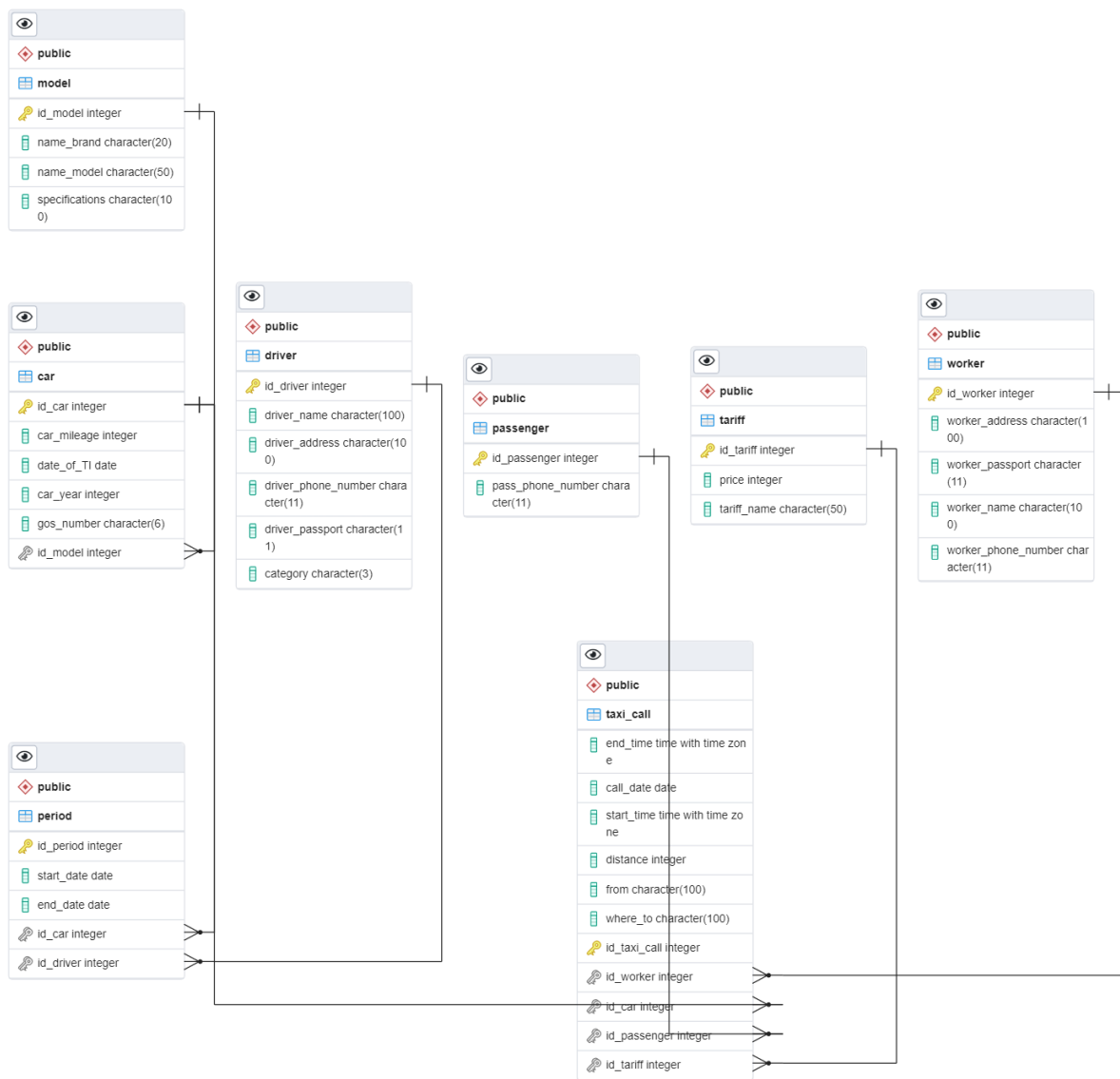


Рис. 1 – Схема логической модели базы данных «Таксопарк»

III. Dump, содержащий скрипты работы с БД

```

1. --
2. -- PostgreSQL database dump
3. --
4.
5. -- Dumped from database version 13.2
6. -- Dumped by pg_dump version 13.2
7.
8. -- Started on 2021-06-01 00:59:20
9.
10. SET statement_timeout = 0;
11. SET lock_timeout = 0;
12. SET idle_in_transaction_session_timeout = 0;
13. SET client_encoding = 'WIN1252';
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 4 (class 2615 OID 2200)
23. -- Name: public; Type: SCHEMA; Schema: -; Owner: postgres
24. --
25. /* Создание схемы */
26. CREATE SCHEMA public;
27.
28.
29. ALTER SCHEMA public OWNER TO postgres;
30.
31. --
32. -- TOC entry 3060 (class 0 OID 0)
33. -- Dependencies: 4
34. -- Name: SCHEMA public; Type: COMMENT; Schema: -; Owner:
    postgres
35. --
36.
37. COMMENT ON SCHEMA public IS 'standard public schema';
38.
39.
40. SET default_tablespace = '';
41.
42. SET default_table_access_method = heap;
43.
44. --
45. -- TOC entry 203 (class 1259 OID 16405)
46. -- Name: car; Type: TABLE; Schema: public; Owner: postgres
47. --
48. /* Создание таблицы «Car» */
49. CREATE TABLE public.car (
50.     id_car INTEGER NOT NULL,
51.     car_mileage INTEGER NOT NULL,
52.     "date_of_TI" DATE NOT NULL,
53.     car_year INTEGER NOT NULL,
54.     gos_number CHARACTER(6) NOT NULL,
55.     id_model INTEGER NOT NULL
56. );
57.
58.
59. ALTER TABLE public.car OWNER TO postgres;
60.
61. --
62. -- TOC entry 201 (class 1259 OID 16395)
63. -- Name: driver; Type: TABLE; Schema: public; Owner: postgres

```

```

64.  --
65.
66.  /* Создание таблицы «Driver» */
67.  CREATE TABLE public.driver (
68.      id_driver INTEGER NOT NULL,
69.      driver_name CHARACTER(100) NOT NULL,
70.      driver_address CHARACTER(100) NOT NULL,
71.      driver_phone_number CHARACTER(11) NOT NULL,
72.      driver_passport CHARACTER(11) NOT NULL,
73.      category CHARACTER(3) NOT NULL
74.  );
75.
76.
77.  ALTER TABLE public.driver OWNER TO postgres;
78.
79.  --
80.  -- TOC entry 208 (class 1259 OID 16430)
81.  -- Name: model; Type: TABLE; Schema: public; Owner: postgres
82.  --
83.
84.  /* Создание таблицы «Model» */
85.  CREATE TABLE public.model (
86.      id_model INTEGER NOT NULL,
87.      name_brand CHARACTER(20) NOT NULL,
88.      name_model CHARACTER(50) NOT NULL,
89.      specifications CHARACTER(100) NOT NULL
90.  );
91.
92.
93.  ALTER TABLE public.model OWNER TO postgres;
94.
95.  --
96.  -- TOC entry 205 (class 1259 OID 16415)
97.  -- Name: passenger; Type: TABLE; Schema: public; Owner:
    postgres
98.  --
99.
100. /* Создание таблицы «Passenger» */
101. CREATE TABLE public.passenger (
102.     id_passenger INTEGER NOT NULL,
103.     pass_phone_number CHARACTER(11) NOT NULL
104. );
105.
106.
107. ALTER TABLE public.passenger OWNER TO postgres;
108.
109. --
110. -- TOC entry 207 (class 1259 OID 16425)
111. -- Name: period; Type: TABLE; Schema: public; Owner: postgres
112. --

```

```

113.
114.  /* Создание таблицы «Period» */
115.  CREATE TABLE public.period (
116.      id_period INTEGER NOT NULL,
117.      start_date DATE NOT NULL,
118.      end_date DATE NOT NULL,
119.      id_car INTEGER NOT NULL,
120.      id_driver INTEGER NOT NULL
121.  );
122.
123.
124.  ALTER TABLE public.period OWNER TO postgres;
125.
126.  --
127.  -- TOC entry 206 (class 1259 OID 16420)
128.  -- Name: tariff; Type: TABLE; Schema: public; Owner: postgres
129.  --
130.
131.  /* Создание таблицы «Tariff» */
132.  CREATE TABLE public.tariff (
133.      id_tariff INTEGER NOT NULL,
134.      price INTEGER NOT NULL,
135.      tariff_name CHARACTER(50) NOT NULL
136.  );
137.
138.
139.  ALTER TABLE public.tariff OWNER TO postgres;
140.
141.  --
142.  -- TOC entry 204 (class 1259 OID 16410)
143.  -- Name: taxi_call; Type: TABLE; Schema: public; Owner:
    postgres
144.  --
145.
146.  /* Создание таблицы «Taxi call» */
147.  CREATE TABLE public.taxi_call (
148.      end_time TIME WITH TIME zone NOT NULL,
149.      call_date DATE NOT NULL,
150.      start_time TIME WITH TIME zone NOT NULL,
151.      distance INTEGER NOT NULL,
152.      "from" CHARACTER(100) NOT NULL,
153.      where_to CHARACTER(100) NOT NULL,
154.      id_taxi_call INTEGER NOT NULL,
155.      id_worker INTEGER NOT NULL,
156.      id_car INTEGER NOT NULL,
157.      id_passenger INTEGER NOT NULL,
158.      id_tariff INTEGER NOT NULL
159.  );
160.
161.

```

```

162. ALTER TABLE public.taxi_call OWNER TO postgres;
163.
164. --
165. -- TOC entry 202 (class 1259 OID 16400)
166. -- Name: worker; Type: TABLE; Schema: public; Owner: postgres
167. --
168.
169. /* Создание таблицы «Worker» */
170. CREATE TABLE public.worker (
171.     id_worker INTEGER NOT NULL,
172.     worker_address CHARACTER(100) NOT NULL,
173.     worker_passport CHARACTER(11) NOT NULL,
174.     worker_name CHARACTER(100) NOT NULL,
175.     worker_phone_number CHARACTER(11) NOT NULL
176. );
177.
178.
179. ALTER TABLE public.worker OWNER TO postgres;
180.
181. --
182. -- TOC entry 3049 (class 0 OID 16405)
183. -- Dependencies: 203
184. -- Data for Name: car; Type: TABLE DATA; Schema: public;
    Owner: postgres
185. --
186.
187. /* Заполнение данными таблицы «Car» */
188. COPY public.car (id_car, car_mileage, "date_of_TI", car_year,
    gos_number, id_model) FROM stdin;
189. 1 4567 2021-01-30      2018 a576kc      3
190. 2 579 2021-01-15      2018 c125cc      1
191. 3 5938 2021-03-27      2020 k983at      2
192. 4 736 2021-02-01      2018 a676ka      4
193. 5 7353 2021-02-01      2018 a783ka      4
194. 6 3782 2021-02-07      2019 p283ee      5
195. \.
196.
197.
198. --
199. -- TOC entry 3047 (class 0 OID 16395)
200. -- Dependencies: 201
201. -- Data for Name: driver; Type: TABLE DATA; Schema: public;
    Owner: postgres
202. --
203.
204. /* Заполнение данными таблицы «Driver» */
205. COPY public.driver (id_driver, driver_name, driver_address,
    driver_phone_number, driver_passport, category) FROM stdin;

```

```

206. 1 Ivanov Ivan
    Lenina 42
    79997376490 9814 574859 B
207. 2 Mikhailov Yuriy
    Moskovskaya 72/2
    79243652332 9813 749387 C
208. 3 Popova Kseniya
    Lomonosova 22
    79142387847 9820 374028 B
209. 4 Kozlov Dmitriy
    Stachek 17
    78653427646 9808 378273 BE
210. 5 Egorova Mariya
    Lermontova 67/4
    79654657967 9810 897652 D
211. \.
212.
213.
214. --
215. -- TOC entry 3054 (class 0 OID 16430)
216. -- Dependencies: 208
217. -- Data for Name: model; Type: TABLE DATA; Schema: public;
    Owner: postgres
218. --
219.
220. /* Заполнение данными таблицы «Model» */
221. COPY public.model (id_model, name_brand, name_model,
    specifications) FROM stdin;
222. 1 Toyota Land Cruiser Prado
    automat
223. 2 Honda Horizon
    automat
224. 3 Renault Logan
    mechanics
225. 4 Skoda Octavia
    mechanics
226. 5 Lexus IS
    automat
227. \.
228.
229.
230. --
231. -- TOC entry 3051 (class 0 OID 16415)
232. -- Dependencies: 205
233. -- Data for Name: passenger; Type: TABLE DATA; Schema: public;
    Owner: postgres
234. --
235.
236. /* Заполнение данными таблицы «Passenger» */

```



```

237. COPY public.passenger (id_passenger, pass_phone_number) FROM
    stdin;
238. 1 79256783498
239. 2 79874763425
240. 3 79453764578
241. 4 89135627856
242. 5 98725386789
243. \.
244.
245.
246. --
247. -- TOC entry 3053 (class 0 OID 16425)
248. -- Dependencies: 207
249. -- Data for Name: period; Type: TABLE DATA; Schema: public;
    Owner: postgres
250. --
251.
252. /* Заполнение данными таблицы «Period» */
253. COPY public.period (id_period, start_date, end_date, id_car,
    id_driver) FROM stdin;
254. 1 2021-01-08      2021-01-31      6      1
255. 2 2021-01-08      2021-01-31      1      3
256. 3 2021-02-01      2021-02-15      2      4
257. 4 2021-02-01      2021-02-15      3      2
258. 5 2021-02-01      2021-02-15      4      5
259. \.
260.
261.
262. --
263. -- TOC entry 3052 (class 0 OID 16420)
264. -- Dependencies: 206
265. -- Data for Name: tariff; Type: TABLE DATA; Schema: public;
    Owner: postgres
266. --
267.
268. /* Заполнение данными таблицы «Tariff» */
269. COPY public.tariff (id_tariff, price, tariff_name) FROM stdin;
270. 1 50 DAY
271. 2 70 night
272. 3 80 morning
273. 4 100 rush HOUR
274. \.
275.
276.
277. --
278. -- TOC entry 3050 (class 0 OID 16410)
279. -- Dependencies: 204
280. -- Data for Name: taxi_call; Type: TABLE DATA; Schema: public;
    Owner: postgres
281. --

```

```

282.
283. /* Заполнение данными таблицы «Taxi call» */
284. COPY public.taxi_call (end_time, call_date, start_time,
    distance, "from", where_to, id_taxi_call, id_worker, id_car,
    id_passenger, id_tariff) FROM stdin;
285. 13:25:00+03 2021-01-15      13:10:00+03      1200 Liteinaya 45/3
    Lomonosova 9
    1 1      3      1      4
286. 09:01:00+03 2021-01-20      08:16:00+03      10470
    Aptekarskiy per 17
    Moskovskaya 73a
    2 2      1      2      3
287. 23:06:00+03 2021-01-20      22:34:00+03      5400 Pravdy 12
    Stachek 29
    3 1      4      3      2
288. \.
289.
290.
291. --
292. -- TOC entry 3048 (class 0 OID 16400)
293. -- Dependencies: 202
294. -- Data for Name: worker; Type: TABLE DATA; Schema: public;
    Owner: postgres
295. --
296.
297. /* Заполнение данными таблицы «Worker» */
298. COPY public.worker (id_worker, worker_address,
    worker_passport, worker_name, worker_phone_number) FROM stdin;
299. 1 Larina 69
    9725 736478 Petrova Larisa
    79253775634
300. 2 Ammosova 4/2
    7837 637687 Soldatova Alena
    79635672395
301. \.
302.
303.
304. --
305. -- TOC entry 2895 (class 2606 OID 16409)
306. -- Name: car car_pkey; Type: CONSTRAINT; Schema: public;
    Owner: postgres
307. --
308.
309. /* Добавление ограничения первичного ключа для столбца id_car
    таблицы «Car» */
310. ALTER TABLE ONLY public.car
311.     ADD CONSTRAINT car_pkey PRIMARY KEY (id_car);
312.
313.
314. --

```

```

315. -- TOC entry 2897 (class 2606 OID 24659)
316. -- Name: car car_unique; Type: CONSTRAINT; Schema: public;
    Owner: postgres
317. --
318. /* Добавление ограничения Unique для столбца gos_number
    таблицы «Car» */
319. ALTER TABLE ONLY public.car
320.     ADD CONSTRAINT car_unique UNIQUE (gos_number);
321.
322.
323. --
324. -- TOC entry 2878 (class 2606 OID 16490)
325. -- Name: car chk_car_mileage; Type: CHECK CONSTRAINT; Schema:
    public; Owner: postgres
326. --
327.
328. /* Добавление ограничения Check для столбца car_mileage
    таблицы «Car» */
329. ALTER TABLE public.car
330.     ADD CONSTRAINT chk_car_mileage CHECK ((car_mileage > 0))
    NOT VALID;
331.
332.
333. --
334. -- TOC entry 2879 (class 2606 OID 16487)
335. -- Name: car chk_car_year; Type: CHECK CONSTRAINT; Schema:
    public; Owner: postgres
336. --
337.
338. /* Добавление ограничения Check для столбца car_year таблицы
    «Car» */
339. ALTER TABLE public.car
340.     ADD CONSTRAINT chk_car_year CHECK ((car_year >= 2015)) NOT
    VALID;
341.
342.
343. --
344. -- TOC entry 2877 (class 2606 OID 24657)
345. -- Name: driver chk_cat ; Type: CHECK CONSTRAINT; Schema:
    public; Owner: postgres
346. --
347.
348. /* Добавление ограничения Check для столбца category таблицы
    «Driver» */
349. ALTER TABLE public.driver
350.     ADD CONSTRAINT "chk_cat " CHECK ((category = ANY
    (ARRAY['B'::bpchar, 'BE'::bpchar, 'C'::bpchar, 'D'::bpchar])))
    NOT VALID;
351.
352.

```

```

353. --
354. -- TOC entry 2880 (class 2606 OID 16491)
355. -- Name: car chk_date_of_TI; Type: CHECK CONSTRAINT; Schema:
    public; Owner: postgres
356. --
357.
358. /* Добавление ограничения Check для столбца date_of_TI таблицы
    «Car» */
359. ALTER TABLE public.car
360.     ADD CONSTRAINT "chk_date_of_TI" CHECK (("date_of_TI" >
    '2021-01-01'::DATE)) NOT VALID;
361.
362.
363. --
364. -- TOC entry 2881 (class 2606 OID 16483)
365. -- Name: taxi_call chk_distance; Type: CHECK CONSTRAINT;
    Schema: public; Owner: postgres
366. --
367.
368. /* Добавление ограничения Check для столбца distance таблицы
    «Taxi call» */
369. ALTER TABLE public.taxi_call
370.     ADD CONSTRAINT chk_distance CHECK ((distance > 100)) NOT
    VALID;
371.
372.
373. --
374. -- TOC entry 2885 (class 2606 OID 16482)
375. -- Name: period chk_end_date; Type: CHECK CONSTRAINT; Schema:
    public; Owner: postgres
376. --
377.
378. /* Добавление ограничения Check для столбца end_date таблицы
    «Period» */
379. ALTER TABLE public.period
380.     ADD CONSTRAINT chk_end_date CHECK ((end_date >=
    start_date)) NOT VALID;
381.
382.
383. --
384. -- TOC entry 2882 (class 2606 OID 16481)
385. -- Name: taxi_call chk_end_time; Type: CHECK CONSTRAINT;
    Schema: public; Owner: postgres
386. --
387.
388. /* Добавление ограничения Check для столбца end_time таблицы
    «Taxi call» */
389. ALTER TABLE public.taxi_call
390.     ADD CONSTRAINT chk_end_time CHECK ((end_time >
    start_time)) NOT VALID;

```

```

391.
392.
393. --
394. -- TOC entry 2884 (class 2606 OID 16489)
395. -- Name: tariff chk_price; Type: CHECK CONSTRAINT; Schema:
    public; Owner: postgres
396. --
397.
398. /* Добавление ограничения Check для столбца price таблицы
    «Tariff» */
399. ALTER TABLE public.tariff
400.     ADD CONSTRAINT chk_price CHECK ((price > 0)) NOT VALID;
401.
402.
403. --
404. -- TOC entry 2883 (class 2606 OID 16435)
405. -- Name: taxi_call chk_taxi_calls; Type: CHECK CONSTRAINT;
    Schema: public; Owner: postgres
406. --
407.
408. ALTER TABLE public.taxi_call
409.     ADD CONSTRAINT chk_taxi_calls CHECK ((id_taxi_call > 0))
    NOT VALID;
410.
411.
412. --
413. -- TOC entry 2887 (class 2606 OID 16399)
414. -- Name: driver driver_pkey; Type: CONSTRAINT; Schema: public;
    Owner: postgres
415. --
416.
417. /* Добавление ограничения первичного ключа для столбца
    id_driver таблицы «Driver» */
418. ALTER TABLE ONLY public.driver
419.     ADD CONSTRAINT driver_pkey PRIMARY KEY (id_driver);
420.
421.
422. --
423. -- TOC entry 2889 (class 2606 OID 24645)
424. -- Name: driver driver_unique; Type: CONSTRAINT; Schema:
    public; Owner: postgres
425. --
426.
427. /* Добавление ограничения Unique для столбца driver_passport
    таблицы «Driver» */
428. ALTER TABLE ONLY public.driver
429.     ADD CONSTRAINT driver_unique UNIQUE (driver_passport,
    driver_phone_number);
430.
431.

```

```

432. --
433. -- TOC entry 2909 (class 2606 OID 16434)
434. -- Name: model_model_pkey; Type: CONSTRAINT; Schema: public;
    Owner: postgres
435. --
436.
437. /* Добавление ограничения первичного ключа для столбца
    id_model таблицы «Model» */
438. ALTER TABLE ONLY public.model
439.     ADD CONSTRAINT model_pkey PRIMARY KEY (id_model);
440.
441. --
442. -- TOC entry 2901 (class 2606 OID 16419)
443. -- Name: passenger_passenger_pkey; Type: CONSTRAINT; Schema:
    public; Owner: postgres
444. --
445.
446. /* Добавление ограничения первичного ключа для столбца
    id_passenger таблицы «Passenger» */
447. ALTER TABLE ONLY public.passenger
448.     ADD CONSTRAINT passenger_pkey PRIMARY KEY (id_passenger);
449.
450.
451. --
452. -- TOC entry 2903 (class 2606 OID 24678)
453. -- Name: passenger_passenger_unique; Type: CONSTRAINT; Schema:
    public; Owner: postgres
454. --
455.
456. /* Добавление ограничения Unique для столбца pass_phone_number
    таблицы «Passenger» */
457. ALTER TABLE ONLY public.passenger
458.     ADD CONSTRAINT passenger_unique UNIQUE
    (pass_phone_number);
459.
460.
461. --
462. -- TOC entry 2907 (class 2606 OID 16429)
463. -- Name: period_period_pkey; Type: CONSTRAINT; Schema: public;
    Owner: postgres
464. --
465.
466. /* Добавление ограничения первичного ключа для столбца
    id_period таблицы «Period» */
467. ALTER TABLE ONLY public.period
468.     ADD CONSTRAINT period_pkey PRIMARY KEY (id_period);
469.
470.
471. --
472. -- TOC entry 2905 (class 2606 OID 16424)

```

```
473. -- Name: tariff_pkey; Type: CONSTRAINT; Schema: public;
    Owner: postgres
474. --
475.
476. /* Добавление ограничения первичного ключа для столбца
    id_tariff таблицы «Tariff» */
477. ALTER TABLE ONLY public.tariff
478.     ADD CONSTRAINT tariff_pkey PRIMARY KEY (id_tariff);
479.
480.
481. --
482. -- TOC entry 2899 (class 2606 OID 16414)
483. -- Name: taxi_call taxi_call_pkey; Type: CONSTRAINT; Schema:
    public; Owner: postgres
484. --
485.
486. /* Добавление ограничения первичного ключа для столбца
    id_taxi_call таблицы «Taxi call» */
487. ALTER TABLE ONLY public.taxi_call
488.     ADD CONSTRAINT taxi_call_pkey PRIMARY KEY (id_taxi_call);
489.
490.
491. --
492. -- TOC entry 2891 (class 2606 OID 16404)
493. -- Name: worker_pkey; Type: CONSTRAINT; Schema: public;
    Owner: postgres
494. --
495.
496. /* Добавление ограничения первичного ключа для столбца
    id_worker таблицы «Worker» */
497. ALTER TABLE ONLY public.worker
498.     ADD CONSTRAINT worker_pkey PRIMARY KEY (id_worker);
499.
500.
501. --
502. -- TOC entry 2893 (class 2606 OID 24714)
503. -- Name: worker_unique; Type: CONSTRAINT; Schema:
    public; Owner: postgres
504. --
505.
506. /* Добавление ограничения Unique для столбца worker_passport
    таблицы «Worker» */
507. ALTER TABLE ONLY public.worker
508.     ADD CONSTRAINT worker_unique UNIQUE (worker_passport,
    worker_phone_number);
509.
510.
511. --
512. -- TOC entry 2912 (class 2606 OID 16456)
```

```

513. -- Name: taxi_call id_car; Type: FK CONSTRAINT; Schema:
    public; Owner: postgres
514. --
515.
516. /* Добавление ограничения внешнего ключа для столбца id_car
    таблицы «Taxi call» */
517. ALTER TABLE ONLY public.taxi_call
518.     ADD CONSTRAINT id_car FOREIGN KEY (id_car) REFERENCES
    public.car(id_car) NOT VALID;
519.
520.
521. --
522. -- TOC entry 2915 (class 2606 OID 16471)
523. -- Name: period id_car; Type: FK CONSTRAINT; Schema: public;
    Owner: postgres
524. --
525.
526. /* Добавление ограничения внешнего ключа для столбца id_car
    таблицы «Period» */
527. ALTER TABLE ONLY public.period
528.     ADD CONSTRAINT id_car FOREIGN KEY (id_car) REFERENCES
    public.car(id_car) NOT VALID;
529.
530.
531. --
532. -- TOC entry 2916 (class 2606 OID 16476)
533. -- Name: period id_driver; Type: FK CONSTRAINT; Schema:
    public; Owner: postgres
534. --
535.
536. /* Добавление ограничения внешнего ключа для столбца id_driver
    таблицы «Period» */
537. ALTER TABLE ONLY public.period
538.     ADD CONSTRAINT id_driver FOREIGN KEY (id_driver)
    REFERENCES public.driver(id_driver) NOT VALID;
539.
540.
541. --
542. -- TOC entry 2910 (class 2606 OID 16505)
543. -- Name: car id_model; Type: FK CONSTRAINT; Schema: public;
    Owner: postgres
544. --
545.
546. /* Добавление ограничения внешнего ключа для столбца id_model
    таблицы «Car» */
547. ALTER TABLE ONLY public.car
548.     ADD CONSTRAINT id_model FOREIGN KEY (id_model) REFERENCES
    public.model(id_model) NOT VALID;
549.
550.

```



```

551. --
552. -- TOC entry 2913 (class 2606 OID 16461)
553. -- Name: taxi_call id_passenger; Type: FK CONSTRAINT; Schema:
    public; Owner: postgres
554. --
555.
556. /* Добавление ограничения внешнего ключа для столбца
    id_passenger таблицы «Taxi call» */
557. ALTER TABLE ONLY public.taxi_call
558.     ADD CONSTRAINT id_passenger FOREIGN KEY (id_passenger)
    REFERENCES public.passenger(id_passenger) NOT VALID;
559.
560.
561. --
562. -- TOC entry 2914 (class 2606 OID 16466)
563. -- Name: taxi_call id_tariff; Type: FK CONSTRAINT; Schema:
    public; Owner: postgres
564. --
565.
566. /* Добавление ограничения внешнего ключа для столбца id_tariff
    таблицы «Taxi call» */
567. ALTER TABLE ONLY public.taxi_call
568.     ADD CONSTRAINT id_tariff FOREIGN KEY (id_tariff)
    REFERENCES public.tariff(id_tariff) NOT VALID;
569.
570.
571. --
572. -- TOC entry 2911 (class 2606 OID 16451)
573. -- Name: taxi_call id_worker; Type: FK CONSTRAINT; Schema:
    public; Owner: postgres
574. --
575.
576. /* Добавление ограничения внешнего ключа для столбца id_worker
    таблицы «Taxi call» */
577. ALTER TABLE ONLY public.taxi_call
578.     ADD CONSTRAINT id_worker FOREIGN KEY (id_worker)
    REFERENCES public.worker(id_worker) NOT VALID;
579.
580.
581. -- Completed on 2021-06-01 00:59:21
582.
583. --
584. -- PostgreSQL database dump complete

```

4. Выводы

В результате выполненной работы:

- Была построена схема и таблицы базы данных согласно индивидуальному заданию – Таксопарк
- Были установлены ограничения на данные Primary Key, Unique, Check, Foreign Key
- Были заполнены таблицы базы данных рабочими данными
- Была создана резервная копия БД в двух расширениях: CUSTOM и PLAIN