

САНКТ- ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

**ФАКУЛЬТЕТ ПРИКЛАДНОЙ МАТЕМАТИКИ ПРОЦЕССОВ
УПРАВЛЕНИЯ**

Углов Владислав Сергеевич

РЕЛЯЦИОННЫЕ БАЗЫ ДАННЫХ

курсовая работа
студента 2 курса бакалавриат ПМИ

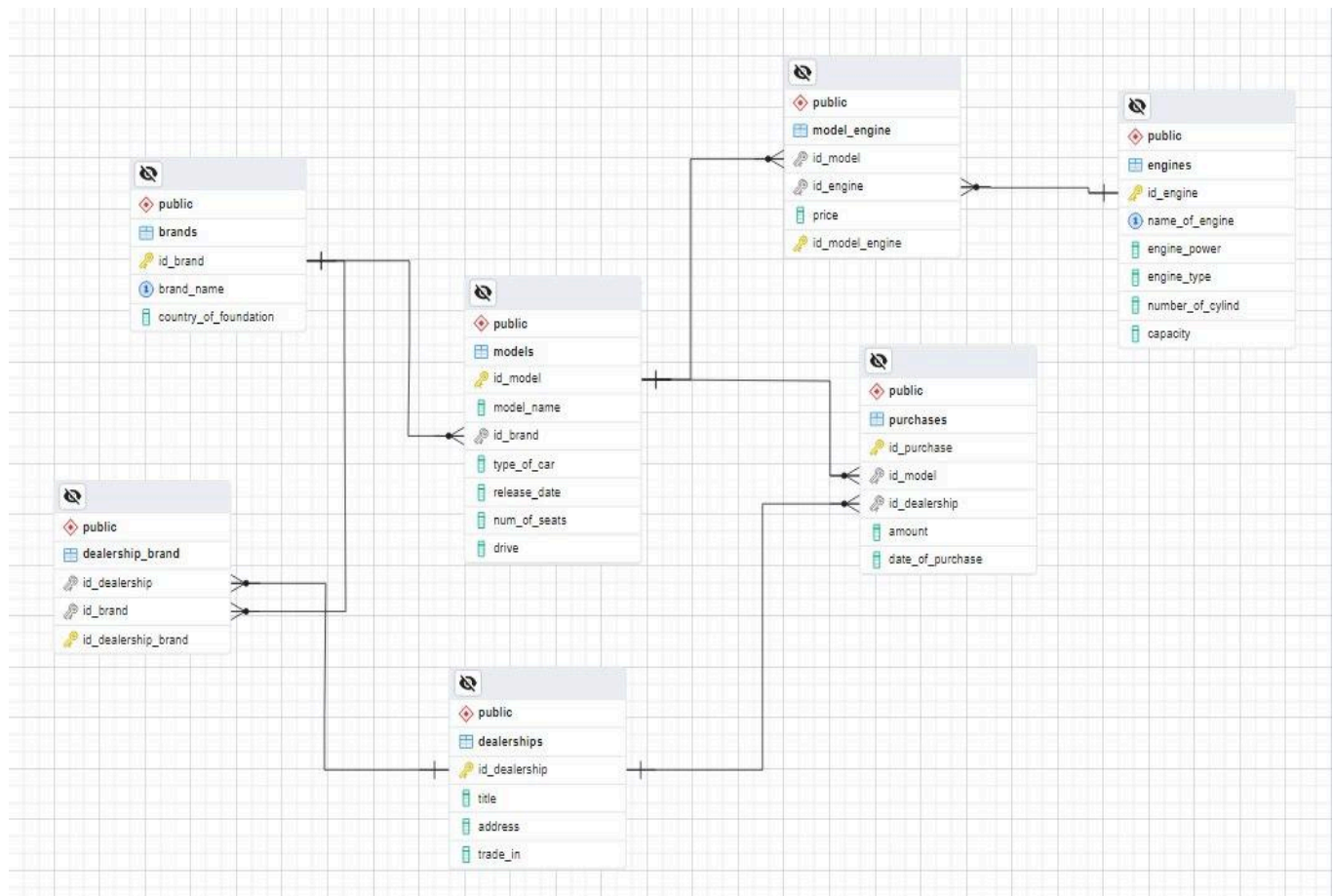


рис. 1

На рис.1 представлена ER-диаграмма сервиса по подбору авто, отражающая содержание базы данных и связи между таблицами. Была реализована небольшая база данных автомобилей и дилерских центров Санкт-Петербурга, содержащая таблицы:

дилерские центры, марки авто, модели, двигатели, покупки авто.

Связи между таблицами: $m:m$, $1:m$. Так, например, у одной модели авто может быть несколько различных двигателей, а один двигателей может устанавливаться на несколько моделей авто($m:m$). У одной марки авто может быть несколько моделей, однако у каждой модели только одна марка($1:m$). База данных приведена к 3 нормальной форме.

Для реализации связей $m:m$ были созданы вспомогательные таблицы:

1. dealership_brand
2. model_engine

SQL-запросы

Легкие

1. Немецкие марки авто
2. Самое старая модель авто
3. Средняя цена всех авто
4. Количество авто каждого типа

Средние

1. Авто, которое купили больше всех в один день
2. Автомобили, которые не покупали в 2023 году
3. Количество моделей авто, проданных в декабре 2023

Сложные

1. Общее количество моделей, купленных в 2023 году, отсортированное в порядке убывания
2. Вывести автомобили с полным приводом, количеством сидений больше 5 и дилерские центры, в которых они продаются.
3. Вывести цену и название авто с мощностью двигателя выше средней

Легкие запросы:

1. Немецкие марки авто

in:

```
1 SELECT brand_name
2 FROM brands
3 WHERE country_of_foundation='Германия';
```

out:

	brand_name character varying (50) 🔒
1	Porsche
2	Volkswagen
3	Mercedes-Benz
4	BMW

2. Самое старая модель авто

in:

```
1 SELECT model_name, release_date
2 FROM models
3 WHERE release_date = (SELECT MIN(release_date) FROM models);
```

out:

	model_name character varying (50) 🔒	release_date smallint 🔒
1	2107	2002

3. Средняя цена всех авто

in:

```
1 SELECT round(AVG(price),1)
2 FROM model_engine;
```

out:

	round numeric 🔒
1	6337383.9

4. Количество авто каждого типа

in:

```
1 SELECT type_of_car, COUNT(*) AS model_count
2 FROM models
3 GROUP BY type_of_car;
4 |
```

out:

	type_of_car character varying (30) 🔒	model_count bigint 🔒
1	Внедорожник	7
2	Универсал	1
3	Купе	1
4	Минивэн	1
5	Кроссовер	6
6	Седан	10
7	Хэтчбек	1

Средние запросы:

1. Авто, которое купили больше всех в один день

in:

```
1 SELECT brand_name, model_name, amount
2 FROM
3     models
4     INNER JOIN brands ON brands.id_brand=models.id_brand
5     INNER JOIN purchases ON models.id_model=purchases.id_model
6     WHERE amount = (SELECT MAX(amount) from purchases);
7
```

out:

	brand_name character varying (50) 🔒	model_name character varying (50) 🔒	amount integer 🔒
1	Chery	Tiggo 7 pro	23

2. Автомобили, которые не покупали в 2023 году

in:

```
1 SELECT brand_name, model_name, amount
2 FROM
3     models
4     LEFT JOIN brands ON brands.id_brand=models.id_brand
5     LEFT JOIN purchases ON purchases.id_model=models.id_model
6 WHERE amount IS NULL;
```

out:

	brand_name character varying (50) 🔒	model_name character varying (50) 🔒	amount integer 🔒
1	Mercedes-Benz	GLS	[null]
2	Chery	Arrizo 8	[null]
3	Volkswagen	Tiguan	[null]
4	Toyota	Alphard	[null]
5	Kia	Sportage	[null]
6	Porsche	911 Turbo S	[null]

3. Количество моделей авто, проданных в декабре 2023
in:

```
1 SELECT
2     model_name, SUM(amount) AS count_sold
3 FROM
4     models INNER JOIN purchases
5     ON models.id_model = purchases.id_model
6 WHERE
7     purchases.date_of_purchase > '30-11-2023'
8 GROUP BY
9     models.model_name;
```

out:

	model_name character varying (50) 🔒	count_sold bigint 🔒
1	Camry	17
2	Cayenne Turbo S	20
3	Duster	22
4	K5	4
5	Largus	20
6	RAV4	1

Сложные запросы:

1. Общее количество моделей, купленных в 2023 году, отсортированное в порядке убывания

in:

```
1 SELECT brand_name, model_name, total_purchases
2 FROM brands
3 INNER JOIN models ON brands.id_brand = models.id_brand
4 INNER JOIN (
5     SELECT id_model, SUM(amount) AS total_purchases
6     FROM purchases
7     GROUP BY id_model
8 ) AS purchase_counts ON models.id_model = purchase_counts.id_model
9 ORDER BY total_purchases DESC;
```

out:

	brand_name character varying (50)	model_name character varying (50)	total_purchases bigint
1	Chery	Tiggo 7 pro	138
2	Porsche	Cayenne Turbo S	40
3	Kia	K5	37
4	LADA	Largus	36
5	Toyota	Camry	35
6	Toyota	RAV4	28
7	Chery	Tiggo 8 Pro Max	27
8	Kia	Sorento	25
9	Tesla	model s	24
10	Renault	Duster	24
11	Kia	Rio	21
12	Mercedes-Benz	S-class	21
13	Volvo	S90	19
14	Skoda	Superb	19
15	Nissan	Qashqai	18
16	LADA	2107	12
17	Volkswagen	Golf R	11
18	LADA	Vesta	8
19	BMW	X7	7
20	Land Rover	Velar	4
21	Volvo	XC90	2

2. Вывести автомобили с полным приводом, количеством мест больше 5 и дилерские центры, в которых они продаются
in:

```
1 SELECT dealership_name, brand_name, model_name, num_of_seats
2 FROM
3     dealerships
4     LEFT JOIN dealership_brand ON dealerships.id_dealership=dealership_brand.id_dealership
5     LEFT JOIN brands ON dealership_brand.id_brand=brands.id_brand
6     LEFT JOIN models ON models.id_brand=brands.id_brand
7 WHERE
8     (drive='полный' and num_of_seats>5)
9 ORDER BY num_of_seats DESC;
```

out:

	dealership_name character varying (30) 🔒	brand_name character varying (50) 🔒	model_name character varying (50) 🔒	num_of_seats smallint 🔒
1	рольф обводный	Chery	Tiggo 8 Pro Max	7
2	авилон юг	Mercedes-Benz	GLS	7
3	автополе на неве	Mercedes-Benz	GLS	7
4	авилон юг	BMW	X7	7
5	рольф обводный	Volvo	XC90	7

3. Вывести цену и название авто с мощностью двигателя выше средней)

in:

```
1 SELECT brand_name, model_name, engine_power, price
2 FROM
3     models
4     LEFT JOIN brands ON brands.id_brand=models.id_brand
5     LEFT JOIN model_engine ON models.id_model=model_engine.id_model
6     LEFT JOIN engines ON engines.id_engine=model_engine.id_engine
7 WHERE engine_power>(SELECT AVG(engine_power) FROM engines)
8 ORDER BY price;
```

out:

	brand_name character varying (50) 🔒	model_name character varying (50) 🔒	engine_power smallint 🔒	price bigint 🔒
1	Volkswagen	Golf R	493	4950000
2	Land Rover	Velar	432	5750000
3	BMW	X7	530	8600000
4	Mercedes-Benz	GLS	432	9890000
5	Porsche	Cayenne Turbo S	390	10500000
6	Mercedes-Benz	GLS	371	10590000
7	BMW	X7	432	12590000
8	BMW	X7	390	14200000
9	Porsche	911 Turbo S	493	18000001
10	Mercedes-Benz	S-class	492	19990000

Легкие запросы

1. Немецкие марки авто

```
SELECT brand_name  
FROM brands  
WHERE country_of_foundation='Германия'
```

2. Самое старое авто

```
SELECT model_name  
FROM models  
WHERE release_date =  
      (SELECT MIN(release_date)  
       FROM models);
```

3. Средняя цена авто

```
SELECT round(AVG(price),1)  
FROM model_engine;
```

4. Количество авто каждого типа

```
SELECT type_of_car, COUNT(*) AS model_count  
FROM models  
GROUP BY type_of_car;
```

Средние запросы

1. Авто, которое купили больше всех в один день

```
SELECT brand_name, model_name, amount  
FROM  
      models  
      INNER JOIN brands ON brands.id_brand=models.id_brand  
      INNER JOIN purchases ON models.id_model=purchases.id_model  
      WHERE amount = (SELECT MAX(amount) from purchases);
```

2. Автомобили, которые не покупали в 2023 году.

```
SELECT brand_name, model_name, amount  
FROM  
      models  
      INNER JOIN brands ON brands.id\_brand=models.id\_brand  
      LEFT JOIN purchases ON purchases.id\_model=models.id\_model  
      WHERE amount IS NULL;
```

3. Количество моделей авто, проданных в декабре 2023

```
SELECT
    models.model_name,
    SUM(purchases.amount) AS count_sold
FROM
    models
    INNER JOIN purchases ON models.id_model = purchases.id_model
WHERE
    purchases.date_of_purchase > '30-11-2023'
GROUP BY
    models.model_name;
```

Сложные запросы

1. Общее количество моделей, купленных в 2023 году, отсортированное в порядке убывания

```
SELECT brand_name, model_name, total_purchases
FROM brands
INNER JOIN models ON brands.id_brand = models.id_brand
INNER JOIN (
    SELECT id_model, SUM(amount) AS total_purchases
    FROM purchases
    GROUP BY id_model
) AS purchase_counts ON models.id_model = purchase_counts.id_model
ORDER BY total_purchases DESC;
```

2. Вывести автомобили с полным приводом, количеством сидений больше 5 и дилерские центры, в которых они продаются.

```
SELECT dealership_name, brand_name, model_name, num_of_seats
FROM
    dealerships
    LEFT JOIN dealership_brand ON
dealerships.id_dealership=dealership_brand.id_dealership
    LEFT JOIN brands ON dealership_brand.id_brand=brands.id_brand
    LEFT JOIN models ON models.id_brand=brands.id_brand
WHERE
    (drive='полный' and num_of_seats>5)
ORDER BY num_of_seats DESC;
```

3. Вывести цену и название авто с мощностью двигателя выше средней)

```
SELECT brand_name, model_name, engine_power, price
FROM
    models
    LEFT JOIN brands ON brands.id_brand=models.id_brand
    LEFT JOIN model_engine ON models.id_model=model_engine.id_model
    LEFT JOIN engines ON engines.id_engine=model_engine.id_engine
WHERE engine_power>(SELECT AVG(engine_power) FROM engines)
ORDER BY price;
```