

После того как таблица определена, ее структуру, свойства и ограничения можно изменить, используя оператор ALTER TABLE.

Для присвоения столбцу обязательного значения (NOT NULL) используется конструкция ALTER COLUMN:

```
photo=# \d client

        Таблица "public.client"
  Столбец |      Тип      | Правило сортировки | Допустимость NULL | По умолчанию
-----+-----+-----+-----+-----
 id_client | integer       |                     | not null           |
 name      | character(35) |                     |                    |
Индексы:
 "client_pkey" PRIMARY KEY, btree (id_client)
Ссылки извне:
 TABLE "journal" CONSTRAINT "fk_client" FOREIGN KEY (id_client) REFERENCES client(id_client)

photo=# alter table client alter column name set not null;
ALTER TABLE
photo=# \d client

        Таблица "public.client"
  Столбец |      Тип      | Правило сортировки | Допустимость NULL | По умолчанию
-----+-----+-----+-----+-----
 id_client | integer       |                     | not null           |
 name      | character(35) |                     | not null           |
Индексы:
 "client_pkey" PRIMARY KEY, btree (id_client)
Ссылки извне:
 TABLE "journal" CONSTRAINT "fk_client" FOREIGN KEY (id_client) REFERENCES client(id_client)
```

```
photo=# \d photograph

        Таблица "public.photograph"
  Столбец |      Тип      | Правило сортировки | Допустимость NULL | По умолчанию
-----+-----+-----+-----+-----
 id_photograph | integer       |                     | not null           |
 name          | character(35) |                     |                    |
Индексы:
 "photograph_pkey" PRIMARY KEY, btree (id_photograph)
Ссылки извне:
 TABLE "journal" CONSTRAINT "fk_photograph" FOREIGN KEY (id_photograph) REFERENCES photograph(id_photograph)

photo=# alter table photograph alter column name set not null;
ALTER TABLE
photo=# \d photograph

        Таблица "public.photograph"
  Столбец |      Тип      | Правило сортировки | Допустимость NULL | По умолчанию
-----+-----+-----+-----+-----
 id_photograph | integer       |                     | not null           |
 name          | character(35) |                     | not null           |
Индексы:
 "photograph_pkey" PRIMARY KEY, btree (id_photograph)
Ссылки извне:
 TABLE "journal" CONSTRAINT "fk_photograph" FOREIGN KEY (id_photograph) REFERENCES photograph(id_photograph)
```

```
photo=# \d zakaz
```

Таблица "public.zakaz"				
Столбец	Тип	Правило сортировки	Допустимость NULL	По умолчанию
id_num	integer			
id_title	integer			
sum	integer			
total	integer			

Ограничения внешнего ключа:

```
"fk_journal" FOREIGN KEY (id_num) REFERENCES journal(id_num)
```

```
"fk_pricelist" FOREIGN KEY (id_title) REFERENCES pricelist(id_title)
```

```
photo=# alter table zakaz alter column id_num set not null;
```

```
ALTER TABLE
```

```
photo=# alter table zakaz alter column id_title set not null;
```

```
ALTER TABLE
```

```
photo=# alter table zakaz alter column sum set not null;
```

```
ALTER TABLE
```

```
photo=# alter table zakaz alter column total set not null;
```

```
ALTER TABLE
```

```
photo=# \d zakaz
```

Таблица "public.zakaz"				
Столбец	Тип	Правило сортировки	Допустимость NULL	По умолчанию
id_num	integer		not null	
id_title	integer		not null	
sum	integer		not null	
total	integer		not null	

Ограничения внешнего ключа:

```
"fk_journal" FOREIGN KEY (id_num) REFERENCES journal(id_num)
```

```
"fk_pricelist" FOREIGN KEY (id_title) REFERENCES pricelist(id_title)
```

```
photo=# \d journal
```

Таблица "public.journal"				
Столбец	Тип	Правило сортировки	Допустимость NULL	По умолчанию
id_num	integer		not null	
id_client	integer			
id_photograph	integer			
date	date			

Индексы:

```
"journal_pkey" PRIMARY KEY, btree (id_num)
```

Ограничения внешнего ключа:

```
"fk_client" FOREIGN KEY (id_client) REFERENCES client(id_client)
```

```
"fk_photograph" FOREIGN KEY (id_photograph) REFERENCES photograph(id_photograph)
```

Ссылки извне:

```
TABLE "zakaz" CONSTRAINT "fk_journal" FOREIGN KEY (id_num) REFERENCES journal(id_num)
```

```
photo=# alter table journal alter column id_client set not null;
```

```
ALTER TABLE
```

```
photo=# alter table journal alter column id_photograph set not null;
```

```
ALTER TABLE
```

```
photo=# alter table journal alter column date set not null;
```

```
ALTER TABLE
```

```
photo=# \d journal
```

Таблица "public.journal"				
Столбец	Тип	Правило сортировки	Допустимость NULL	По умолчанию
id_num	integer		not null	
id_client	integer		not null	
id_photograph	integer		not null	
date	date		not null	

Индексы:

```
"journal_pkey" PRIMARY KEY, btree (id_num)
```

Ограничения внешнего ключа:

```
"fk_client" FOREIGN KEY (id_client) REFERENCES client(id_client)
```

```
"fk_photograph" FOREIGN KEY (id_photograph) REFERENCES photograph(id_photograph)
```

Ссылки извне:

```
TABLE "zakaz" CONSTRAINT "fk_journal" FOREIGN KEY (id_num) REFERENCES journal(id_num)
```

```

photo=# \d pricelist
        Таблица "public.pricelist"
  Столбец |      Тип      | Правило сортировки | Допустимость NULL | По умолчанию
-----+-----+-----+-----+-----
 id_title | integer       |                     | not null           |
 title    | character(40)  |                     |                     |
 price    | integer       |                     |                     |
Индексы:
    "pricelist_pkey" PRIMARY KEY, btree (id_title)
Ссылки извне:
    TABLE "zakaz" CONSTRAINT "fk_pricelist" FOREIGN KEY (id_title) REFERENCES pricelist(id_title)

photo=# alter table pricelist alter column title set not null;
ALTER TABLE
photo=# alter table pricelist alter column price set not null;
ALTER TABLE
photo=# \d pricelist
        Таблица "public.pricelist"
  Столбец |      Тип      | Правило сортировки | Допустимость NULL | По умолчанию
-----+-----+-----+-----+-----
 id_title | integer       |                     | not null           |
 title    | character(40)  |                     | not null           |
 price    | integer       |                     | not null           |
Индексы:
    "pricelist_pkey" PRIMARY KEY, btree (id_title)
Ссылки извне:
    TABLE "zakaz" CONSTRAINT "fk_pricelist" FOREIGN KEY (id_title) REFERENCES pricelist(id_title)

```

Также, при помощи данного оператора можно добавлять (удалять) новые столбцы:

```

photo=# \d client
        Таблица "public.client"
  Столбец |      Тип      | Правило сортировки | Допустимость NULL | По умолчанию
-----+-----+-----+-----+-----
 id_client | integer       |                     | not null           |
 name      | character(35)  |                     | not null           |
Индексы:
    "client_pkey" PRIMARY KEY, btree (id_client)
Ссылки извне:
    TABLE "journal" CONSTRAINT "fk_client" FOREIGN KEY (id_client) REFERENCES client(id_client)

photo=# alter table client add column phone varchar;
ALTER TABLE
photo=# \d client
        Таблица "public.client"
  Столбец |      Тип      | Правило сортировки | Допустимость NULL | По умолчанию
-----+-----+-----+-----+-----
 id_client | integer       |                     | not null           |
 name      | character(35)  |                     | not null           |
 phone     | character varying |                     |                     |
Индексы:
    "client_pkey" PRIMARY KEY, btree (id_client)
Ссылки извне:
    TABLE "journal" CONSTRAINT "fk_client" FOREIGN KEY (id_client) REFERENCES client(id_client)

```

```

photo=# alter table client drop column phone;
ALTER TABLE
photo=# \d client
        Таблица "public.client"
  Столбец |      Тип      | Правило сортировки | Допустимость NULL | По умолчанию
-----+-----+-----+-----+-----
 id_client | integer       |                     | not null           |
 name      | character(35)  |                     | not null           |
Индексы:
    "client_pkey" PRIMARY KEY, btree (id_client)
Ссылки извне:
    TABLE "journal" CONSTRAINT "fk_client" FOREIGN KEY (id_client) REFERENCES client(id_client)

```

Для того, чтобы выполнить запрос выборки используется оператор SELECT.

Чтение заданных столбцов из одиночной таблицы

```
photo=# select id_num, date from journal;
 id_num |      date
-----+-----
    2003 | 2023-03-17
(1 строка)
```

Чтение заданных строк из одиночной таблицы

```
photo=# select title, price from pricelist where price > '300';
          title          | price
-----+-----
    Фото 9x12           |   340
    Фото 3x4            |   340
(2 строки)
```

Чтение заданных строк и столбцов из одиночной таблицы

Форма предложения WHERE предполагает задания списка значений, которые может иметь столбец. Это делается с помощью ключевого слова IN:

```
photo=# select * from client where id_client in ('102', '104');
 id_client |              name
-----+-----
        102 | Проскурина Анна Владимировна
        104 | Молякова Светлана Михайловна
(2 строки)
```

Чтобы выбрать строки, у которых столбец не равен ни одному из заданных значений, используется ключевое слово NOT IN:

```
photo=# select * from client where id_client not in ('102', '104');
 id_client |              name
-----+-----
        101 | Каркаров Игорь Петрович
        103 | Костяшкин Филипп Сергеевич
(2 строки)
```

Чтение запроса в заданном диапазоне

В предложениях WHERE могут также указываться диапазоны. Для задания диапазонов используется ключевое слово BETWEEN:

```
photo=# select * from pricelist where price between '20' and '340';
 id_title |          title          | price
-----+-----+-----
     1102 | Печать цветная         |    25
     1104 | Фото 9x12               |   340
     1105 | Фото 3x4                |   340
(3 строки)
```

Сортировка результатов

Используя ORDER BY 'Название колонки' ASC (DESC), можно вывести результаты в порядке возрастания (убывания):

```
photo=# select * from client order by name asc;
 id_client |          name
-----+-----
      101 | Каркаров Игорь Петрович
      103 | Костяшкин Филипп Сергеевич
      104 | Молякова Светлана Михайловна
      102 | Проскурина Анна Владимировна
(4 строки)
```

```
photo=# select id_title, title from pricelist order by title desc;
 id_title |          title
-----+-----
     1104 | Фото 9x12
     1105 | Фото 3x4
     1101 | Печать ч/б
     1102 | Печать цветная
     1103 | Ксерокопия
(5 строк)
```

Чтение запроса по условию символа

Для задания шаблонов поиска в SQL используется ключевое слово LIKE. Символ процента ('%') представляет последовательность из одного или нескольких произвольных символов.

```
photo=# select * from client where name like 'K%';
 id_client |          name
-----+-----
      101 | Каркаров Игорь Петрович
      103 | Костяшкин Филипп Сергеевич
(2 строки)
```

Агрегатные функции SQL

Используя функцию COUNT, можно вывести количество строк из определённой таблицы, а функции MIN и MAX выводят минимальные и максимальные значения из таблицы.

```
photo=# select count(*) from client;
 count
-----
      4
(1 строка)
```

```
photo=# select min(price), max(price) from pricelist;
 min | max
-----+-----
   10 | 340
(1 строка)
```

Используя UPDATE можно изменить данные не только одного столбца, но и сразу нескольких:

```
photo=# select * from client;
 id_client |          name
-----+-----
      101 | Каркаров Игорь Петрович
      102 | Проскурина Анна Владимировна
      103 | Костяшкин Филипп Сергеевич
      104 | Молякова Светлана Михайловна
(4 строки)

photo=# update client set name = 'Костяшкин Кирилл Филиппович' where id_client = 103;
UPDATE 1
photo=# select * from client;
 id_client |          name
-----+-----
      101 | Каркаров Игорь Петрович
      102 | Проскурина Анна Владимировна
      104 | Молякова Светлана Михайловна
      103 | Костяшкин Кирилл Филиппович
(4 строки)
```