# Министерство образование Новосибирской области

# ГБПОУ НСО «Новосибирский авиационный технический колледж имени Б. С. Галущака»

**МДК. 11.01 Технология разработки и защиты баз данных**

**Практическое занятие 5**

Разработал(а):

Еремеева Л.А.

2024 г

## Цель работы

Изучить базовые операции работы с датой и временем. Изучить синтаксис команд. Приобрести навыки работы с датой и временем в PostgreSQL.

## Задание лабораторной работы

На компьютере проделать примеры, получить результат. Вставить результат в качестве скриншота в отчет по лабораторной работе.

## Функции работы с датой и временем

В данном подразделе речь пойдет о функциях работы с датой и временем, а также функциями преобразования типов для даты. Для хранения даты и времени в PostgreSQL предусмотрен специальный тип timestamp. С физической точки зрения это дробное число, целая часть которого хранит количество дней с некоторой базовой даты, а дробная – время. Это позволяет совершать над датами арифметические операции – сложение и вычитание.

***Функция NOW***

Это одна из самых часто употребляемых функций, она возвращает текущую дату и время по часам сервера.

Пример:

SELECT NOW();

NOW

2016-09-05 21:31:10.095806+04

***Функция JUSTIFY\_INTERVAL***

Функция JUSTIFY\_INTERVAL(interval) преобразует интервал (тип interval), указанный в виде строки в соответствующее значение типа timestamp.

Пример:

SELECT NOW() D1,

NOW() + JUSTIFY\_INTERVAL(’30 DAYS 1 HOUR 2 MINUTE’) D2, NOW() - JUSTIFY\_INTERVAL(’30 DAYS 1 HOUR 2 MINUTE’) D3;

|  |  |  |
| --- | --- | --- |
| D1 | D2 | D3 |
| 2016-09-05  21:39:08...+04 | 2016-10-05  22:41:08...+04 | 2016-08-05  20:37:08...+04 |

***Функия DATE\_TRUNC***

Функция DATE\_TRUNC(timestamp) используется для обрезки да- ты или интервала (DATE\_TRUNC(interval)) до определенной точно- сти.

Пример:

SELECT

DATE\_TRUNC(’HOUR’, NOW()) D1, DATE\_TRUNC(’DAY’, NOW()) D2, DATE\_TRUNC(’MONTH’, NOW()) D3;

|  |  |  |
| --- | --- | --- |
| D1 | D2 | D3 |
| 2016-09-05 22:00:00+04 | 2016-09-05 00:00:00+04 | 2016-09-01 00:00:00+04 |

***Получение начала и конца месяца***

Для получения дат соответствующих началу и концу месяца необходимо использовать функции DATE\_TRUNC и JUSTIFY\_INTERVAL.

Пример:

SELECT

DATE\_TRUNC(’MONTH’, NOW()) D1, DATE\_TRUNC(’MONTH’, NOW())

+ JUSTIFY\_INTERVAL(’1 MONTH - 1 DAY’) D2;

|  |  |
| --- | --- |
| D1 | D2 |
| 2016-09-01 00:00:00+04 | 2016-09-30 00:00:00+04 |

Данные функции также могут быть использованы для определения количества дней в заданном месяце.

Например:

SELECT NOW() D1, TO\_CHAR(DATE\_TRUNC(’MONTH’, NOW())

+ JUSTIFY\_INTERVAL(’1 MONTH - 1 DAY’), ’DD’) D2;

|  |  |
| --- | --- |
| D1 | D2 |
| 2016-09-05 21:51:19.727702+04 | 30 |

***Функция AGE***

Функция AGE([end\_date, ]start\_date) возвращает разницу между датами, обозначенными как end\_date и start\_date. Если параметр end\_date опущен, то используется значение глобальной переменной CURRENT\_DATE, которая содержит текущую дату (тип date, дата без времени).

Пример:

SELECT CURRENT\_DATE D1,

AGE(MAKE\_TIMESTAMP(2013, 7, 15, 8, 15, 23.5))

D2, AGE(MAKE\_DATE(2016, 3, 3),

MAKE\_TIMESTAMP(2013, 7, 15, 8, 15, 23.5)) D3;

|  |  |  |
| --- | --- | --- |
| D1 | D2 | D3 |
| 2016-09-05 | 3 years 1 mon 20 days 15:44:36.5 | 2 years 7 mons 18 days 15:44:36.5 |

В данном примере также используются функции MAKE\_TIMESTAMP и MAKE\_DATE, которые возвращают значения типов timestamp и date соответственно.

***Функция EXTRACT***

Функция EXTRACT(field FROM timestamp) извлекает элемент даты field из значения типа timestamp. Также существует функция EXTRACT(field FROM interval) для работы со значениями типа interval.

Пример: SELECT NOW()

D1,

EXTRACT(MONTH FROM NOW()) D2, EXTRACT(YEAR FROM NOW()) D3, EXTRACT(MINUTE FROM NOW()) D4;

|  |  |  |  |
| --- | --- | --- | --- |
| D1 | D2 | D3 | D4 |
| 2016-09-05 22:37:54.6046+04 | 9 | 2016 | 37 |

***Количество месяцев между двумя датами***

Функции EXTRACT, AGE и DATE\_TRUNC могут быть использованы для нахождения количества месяцев между двумя датами.

Пример: SELECT NOW()

D1,

EXTRACT(MONTH FROM AGE(DATE\_TRUNC(’MONTH’, NOW()), DATE\_TRUNC(’MONTH’, MAKE\_DATE(2016, 3, 3)))) D2;

|  |  |
| --- | --- |
| D1 | D2 |
| 2016-09-05 22:41:08.68334+04 | 6 |

***Функция TO\_DATE***

Функция TO\_DATE(str, mask) преобразует строку str в дату. Преобразование ведется по маске mask.

Пример:

SELECT

TO\_DATE(’05 Dec 2000’, ’DD Mon YYY’) D1, TO\_DATE(’15.12.2000’, ’dd.mm.yy’) D2;

|  |  |
| --- | --- |
| D1 | D2 |
| 2000-12-05 | 2000-12-05 |

***Функция TO\_CHAR***

Функция TO\_CHAR(date, mask) преобразует дату date в символь- ную строку в соответствии с заданной маской.

Пример:

SELECT NOW() D1,

TO\_CHAR(NOW(), ’DD.MM.YY HH24:MI’) D2;

Например:

|  |  |
| --- | --- |
| D1 | D2 |
| 2016-09-05 22:49:37.044199+04 | 05.09.16 22:49 |

Некоторые форматные маски представлены в таблице 5.1.

## Маски для форматирования даты и времени

|  |  |
| --- | --- |
| **Маска** | **Назначение** |
| СС | Номер века |
| YYYY,илиYY | Четыре или последние две цифры номера  года |
| Q | Номер квартала |
| MONTH, или MON, или MM | Строковое или числовое представление  номера месяца |
| WW | Номер недели вгоду |
| W | Номер недели вмесяце |
| DDD | Номер дня вгоду |
| DD | Номер дня вмесяце |
| DAY,или DY,илиID | Номер дня недели |
| HH,или HH12, или HH24 | Час |
| MI | Минута |

1. вариант:

***ВАРИАНТЫ ДЛЯ ПРАКТИЧЕСКОЙ ЧАСТИ***

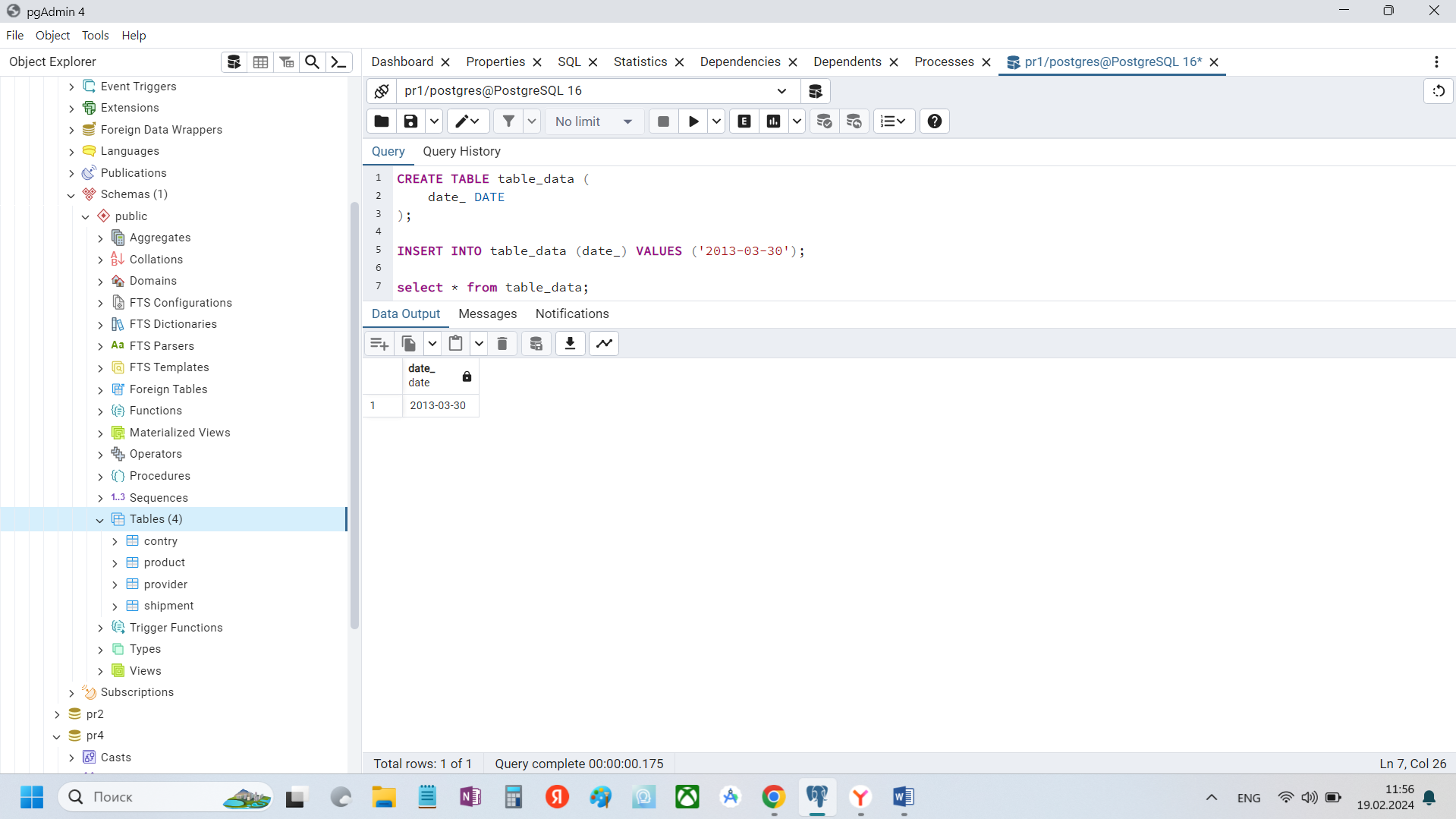
|  |  |
| --- | --- |
| 1 | WHERE date = CURDATE() |

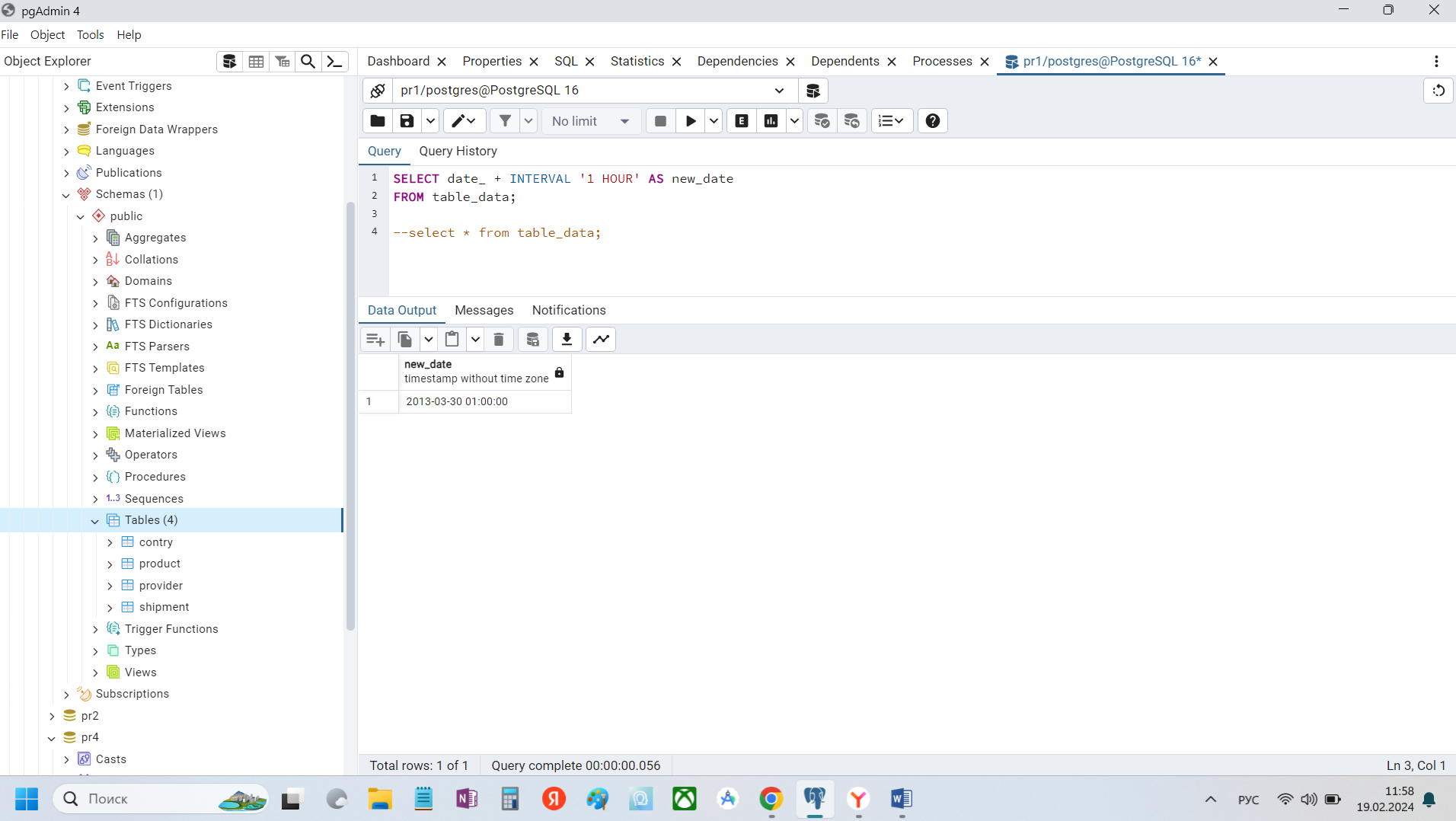
1. вариант:

|  |  |
| --- | --- |
| 1 | WHERE date = STR\_TO\_DATE(now(), '%Y-%m-%d') |

***Прибавить к дате один час в SQL***

|  |  |
| --- | --- |
| 1 | DATE\_ADD('2013-03-30', INTERVAL 1 HOUR) |

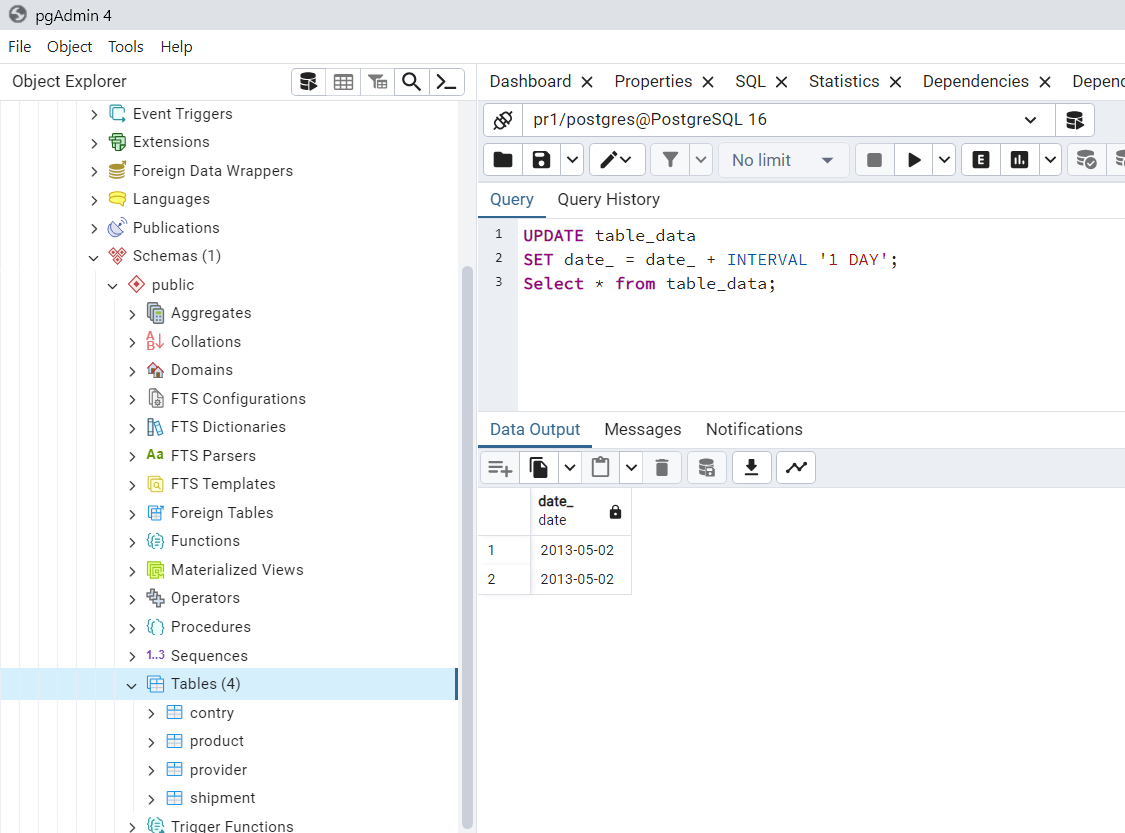


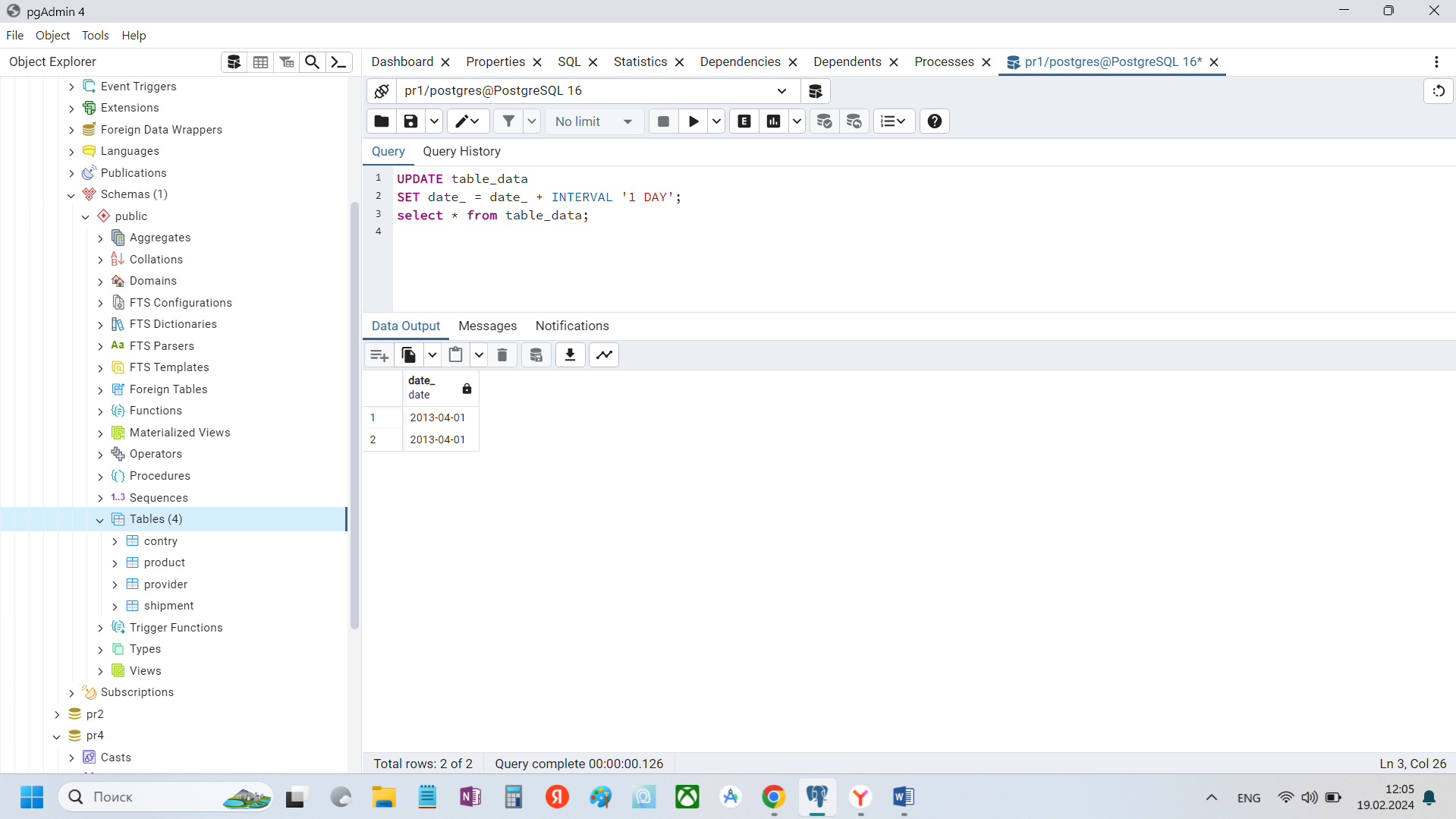


***Прибавить к дате один день в SQL***

|  |  |
| --- | --- |
| 1 | DATE\_ADD('2013-03-30', INTERVAL 1 DAY) |

Аналогично можно прибавлять любое количество дней к текущей дате.



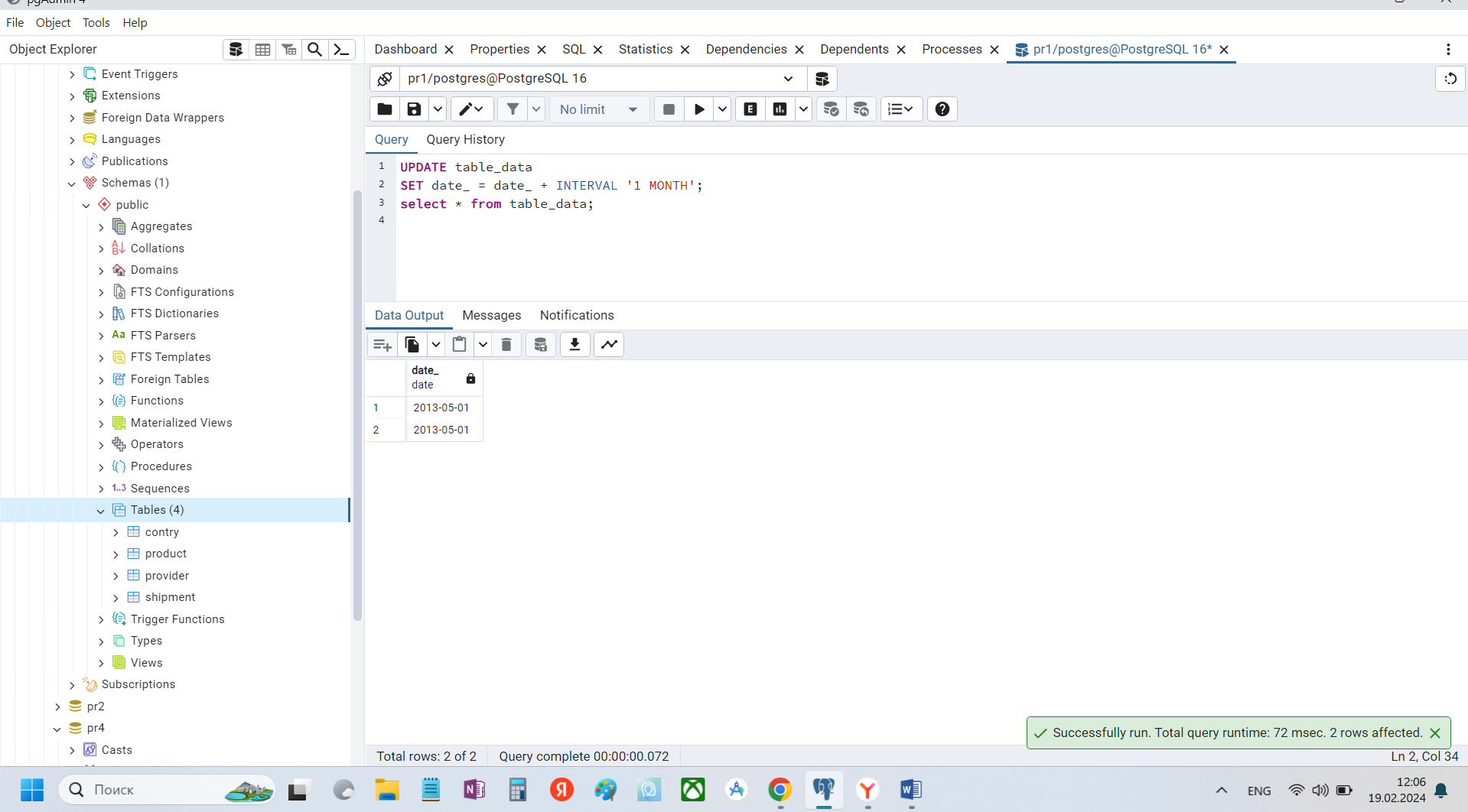


***Прибавить к дате один месяц в SQL***

|  |  |
| --- | --- |
| 1 | DATE\_ADD('2013-03-30', INTERVAL 1 MONTH) |

Аналогично можно прибавлять любое количество месяцев к текущей

дате.



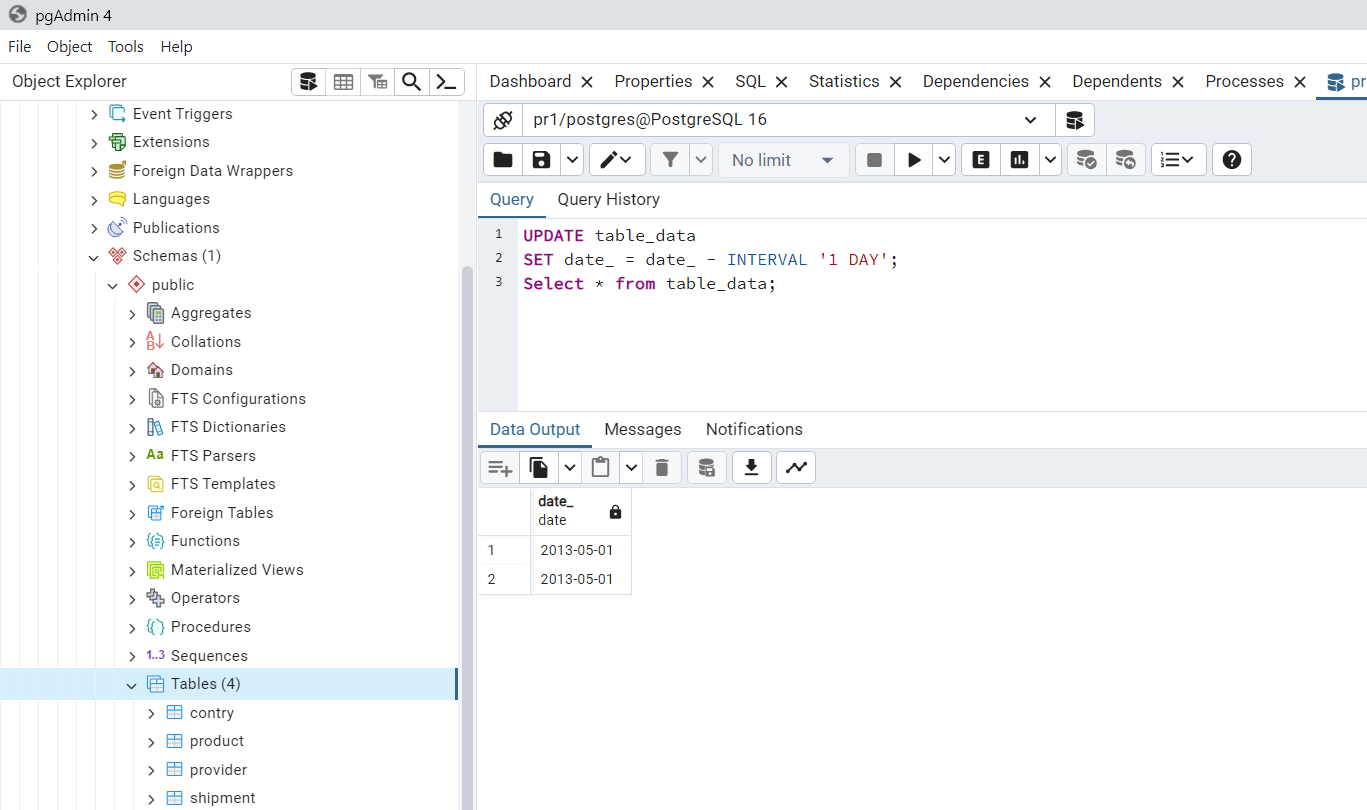
Первый вариант:

***Получить вчерашний день в SQL***

|  |  |
| --- | --- |
| 1 | DATE\_ADD(CURDATE(), INTERVAL -1 DAY) |

Второй вариант:

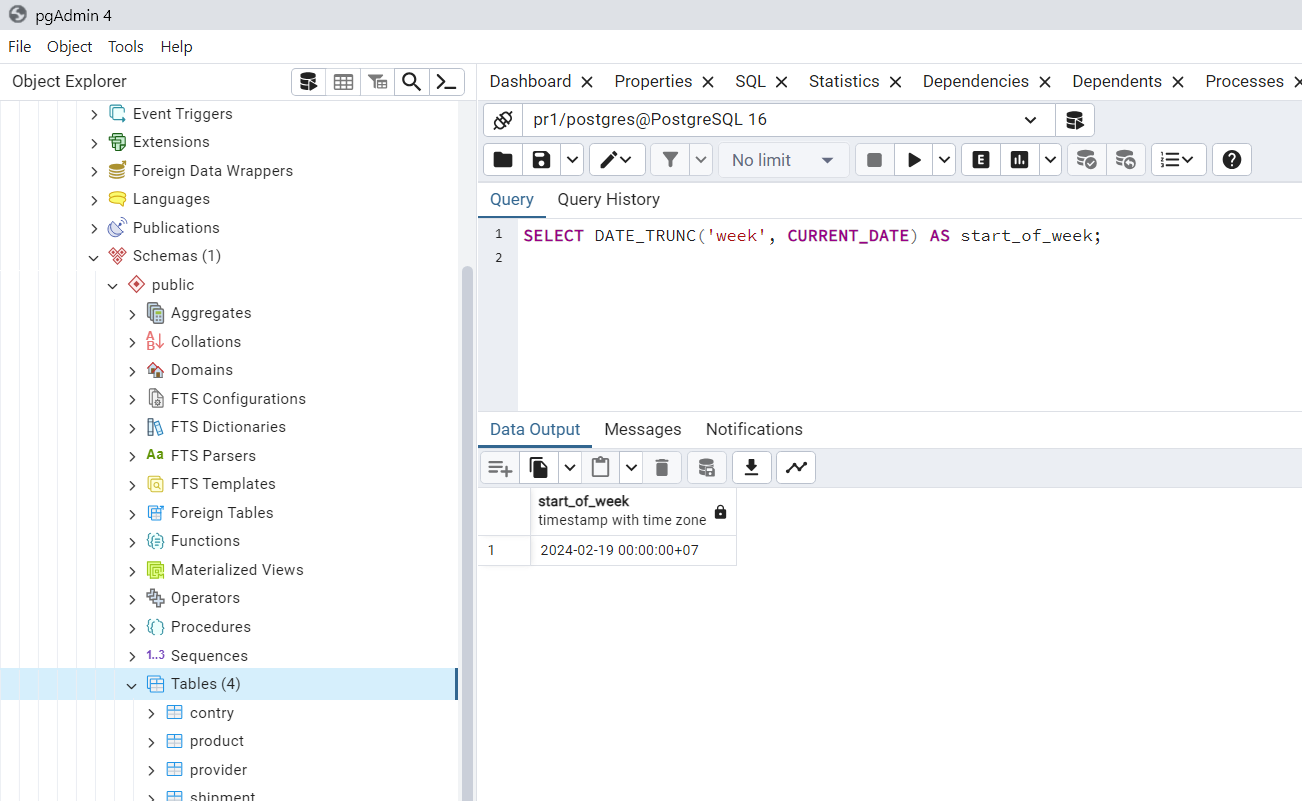
|  |  |
| --- | --- |
| 1 | DATE\_SUB(CURDATE(), INTERVAL 1 DAY) |



***Получить дату начала текущей недели в SQL***

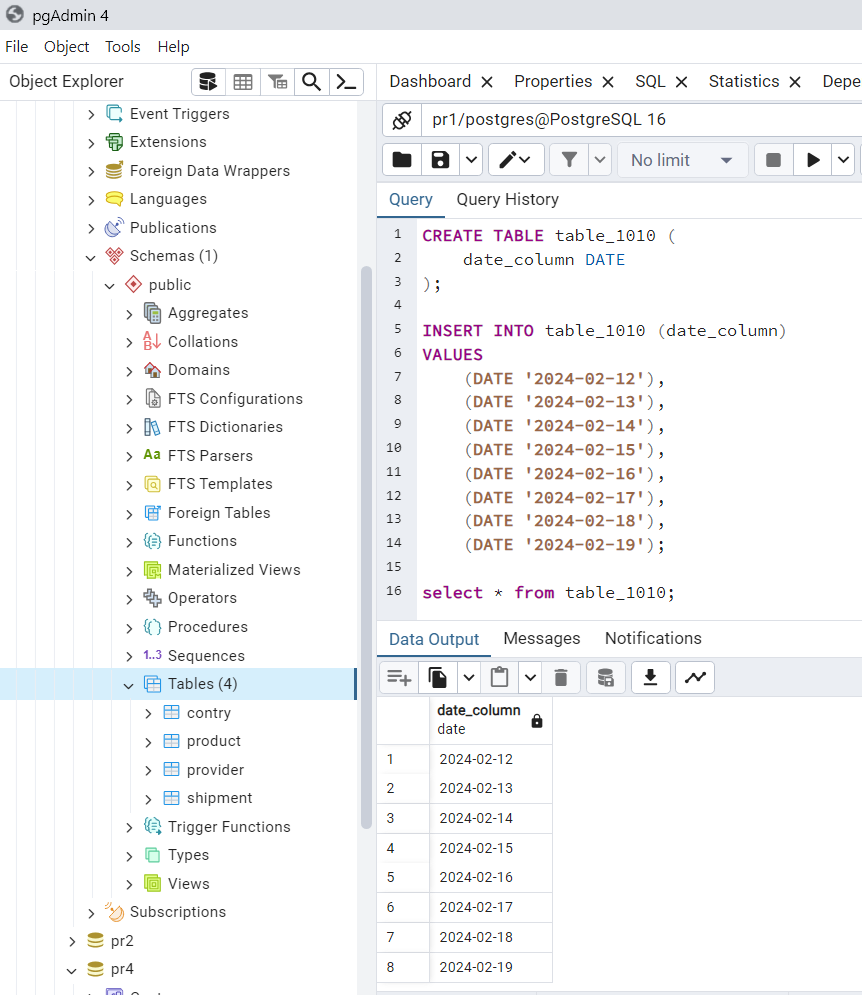
Вот эта одна из самых сложных на первый взгляд задач, но решается очень просто:

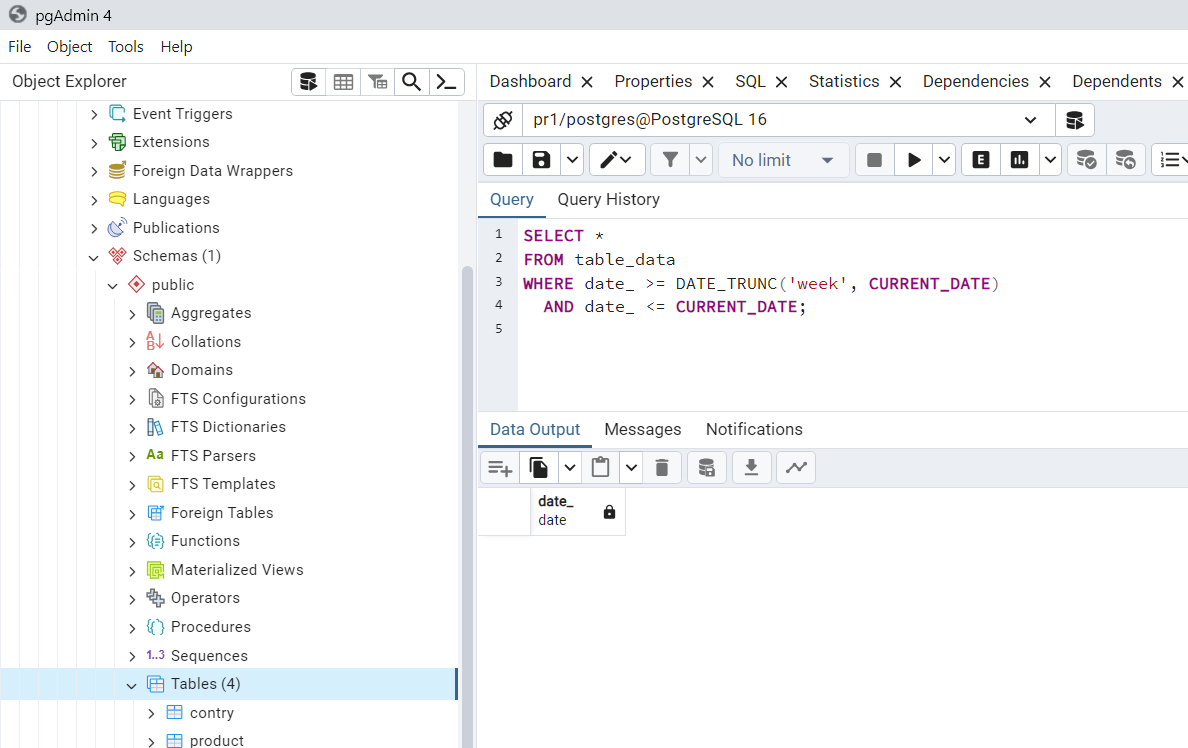
|  |  |
| --- | --- |
| 1 | CURDATE()-WEEKDAY(CURDATE()); |



***Получить выборку с этого понедельника по текущий день недели в SQL***

|  |  |
| --- | --- |
| 1  2  3  4  5 | WHERE (  date BETWEEN  (CURDATE()-WEEKDAY(CURDATE())) AND  CURDATE() |
|  |  |

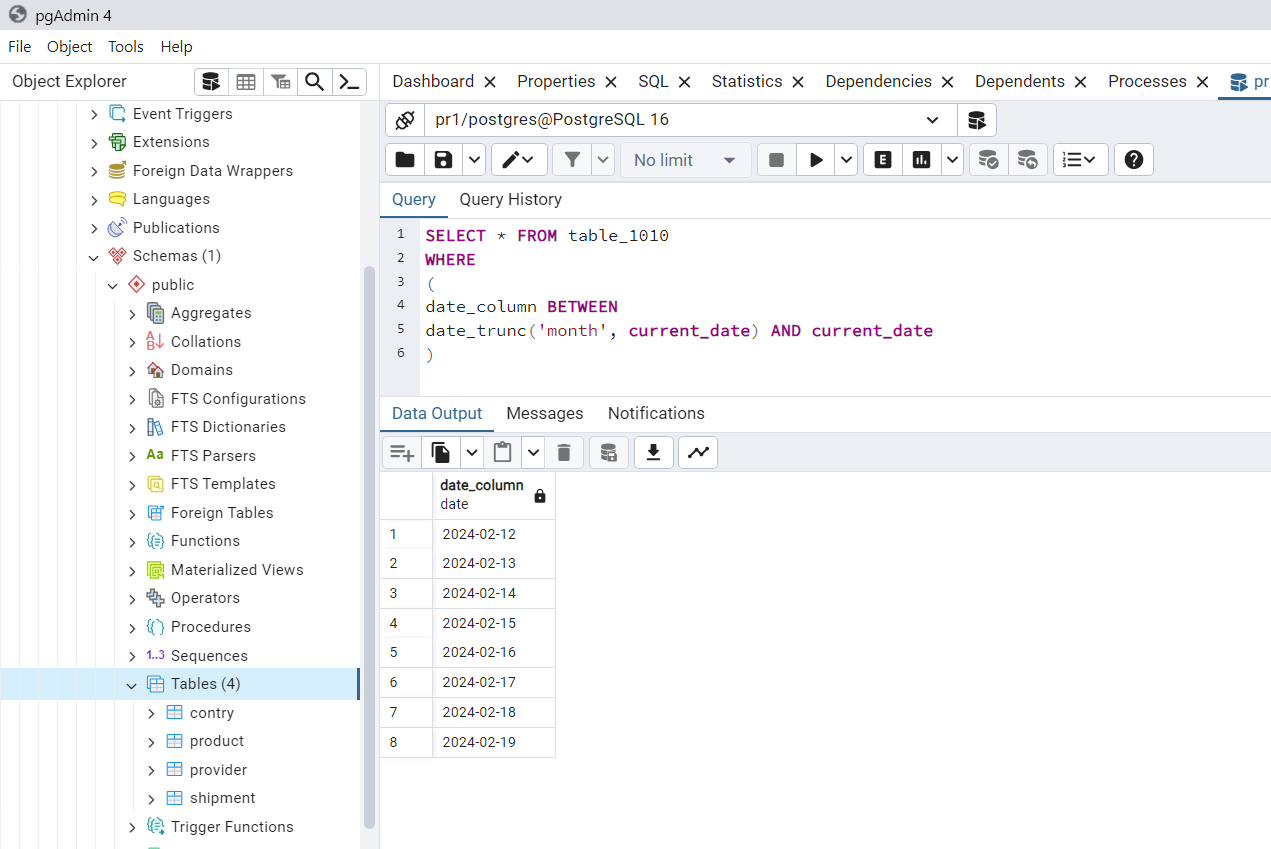




6 )

***Получить выборку с первого числа текущего месяца по текущий день недели в SQL***

|  |  |
| --- | --- |
| 1  2  3  4  5 | WHERE (  date BETWEEN  (CURDATE()-WEEKDAY(CURDATE())) AND  CURDATE()) |



***Как получить дату рождения пользователя в SQL***

|  |  |  |  |
| --- | --- | --- | --- |
| 1  2  3  4  5 | SELECT name, birth, CURRENT\_DATE, (YEAR(CURRENT\_DATE)-YEAR(birth))  - (RIGHT(CURRENT\_DATE,5)<RIGHT(birth,5))  AS age FROM user; |  |  |

***Найти всех пользователей у которых день рождение в следующем месяце в SQL***

|  |  |
| --- | --- |
| 1  2 | SELECT name, birth FROM user  WHERE MONTH(birth) = MONTH(DATE\_ADD(NOW(), INTERVAL 1 MONTH)); |

Или второй вариант:

|  |  |
| --- | --- |
| 1  2 | SELECT name, birth FROM pet  WHERE MONTH(birth) = MOD(MONTH(NOW()), 12) + 1; |

Кроме вышеприведенных кейсов по **работе с датами в SQL**, рекомендую ознакомиться с документацией по следующим операторам :

**NOW()** — Возвращает текущую дату и время.

**CURDATE()** — Возвращает текущую дату. **CURTIME()** — Возвращаем текущее время. **DATE()** — Состоит из двух частей даты и времени.

**EXTRACT()** — Возвращает одно значения даты/времени.

**DATE\_ADD()** — Добавляет до выборки указанное число дней/мину/часов и т.д.

**DATE\_SUB()** — Вычитываем указанный интервал от даты.

**DATEDIFF()** — Возвращает значение времени между двумя датами.

DATE\_FORMAT() — Функция для различного вывода информации о времени.

**Работа с датами в SQl**, как оказывается не такая сложная, и теперь вместо того чтобы вычислять периоды средствами PHP можно делать это еще на этапе выполнения SQL запроса и получать необходимую выборку данных.