Министерство науки и высшего образования Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Факультет инфокоммуникационных технологий

**ОТЧЕТ**

**О ЛАБОРАТОРНОЙ РАБОТЕ № 4**

по теме: Запросы на выборку данных к БД PostgreSQL.

по дисциплине: Проектирование и реализация баз данных

**Специальность**:

09.03.03 Мобильные и сетевые технологии

|  |  |
| --- | --- |
| **Проверил**:  Говорова М.М. \_\_\_\_\_\_\_\_\_\_  **Дата**: «\_\_» \_\_\_\_ 2021 г.  Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_ | **Выполнил**:  студент группы K3240  Рейнгеверц В.А. |

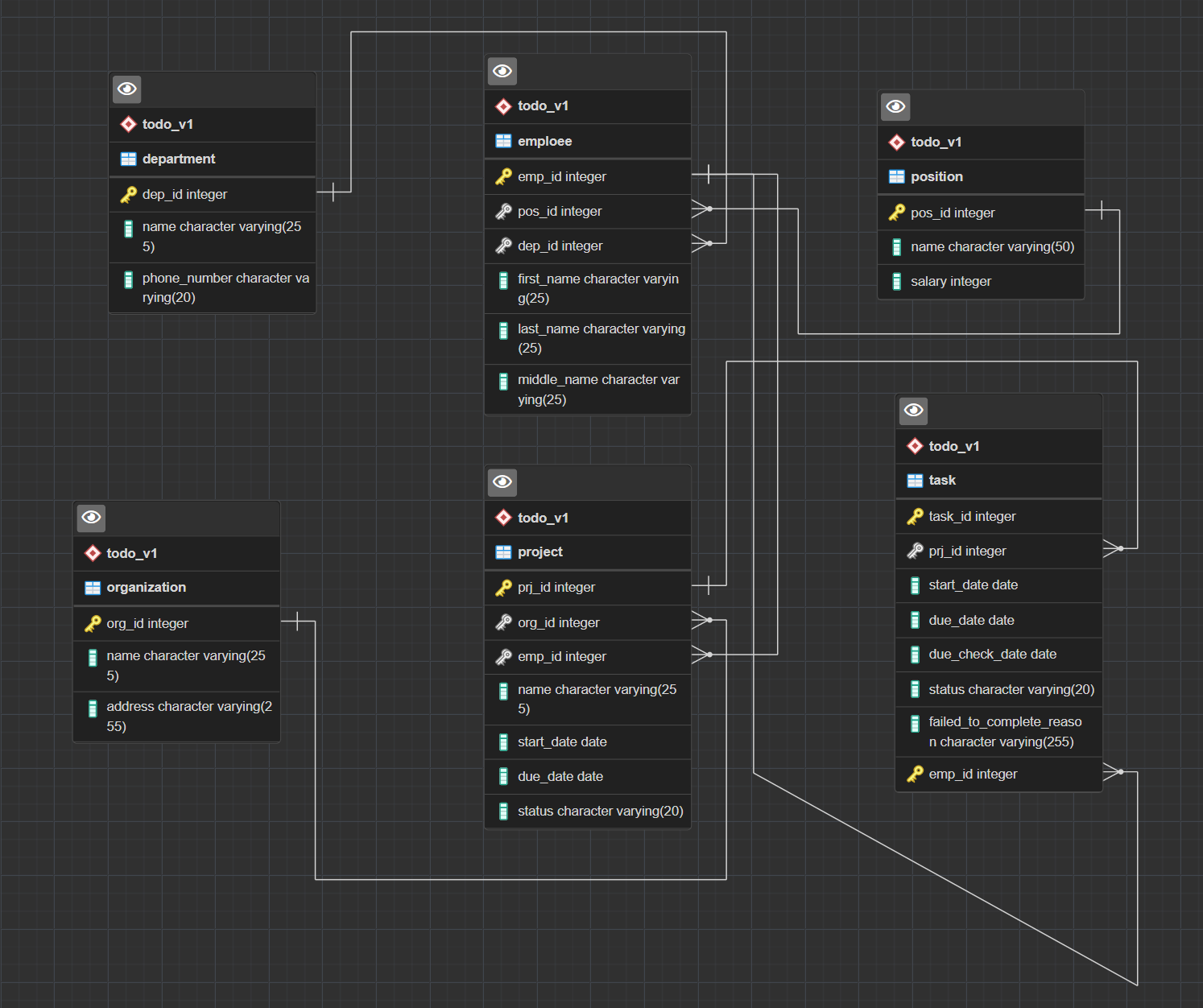
Санкт-Петербург 2021 г.

**ЦЕЛЬ РАБОТЫ**

Овладеть практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQL и использования подзапросов при модификации данных.

**ПРАКТИЧЕСКОЕ ЗАДАНИЕ**

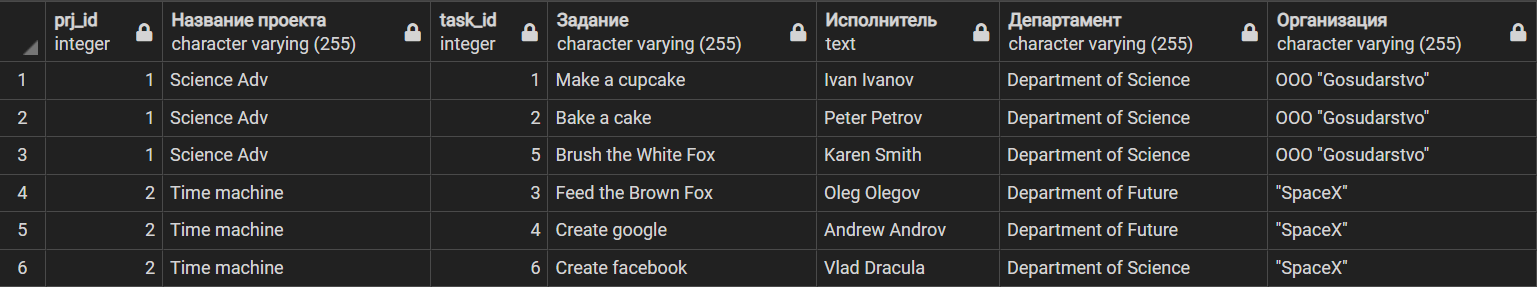
1. Создать запросы и представления на выборку данных к базе данных PostgreSQL (согласно индивидуальному заданию, часть 2 и 3).
2. Составить 3 запроса на модификацию данных (INSERT, UPDATE, DELETE) с использованием подзапросов.
3. Изучить графическое представление запросов.
4. Посмотреть историю запросов.

**СХЕМА БАЗЫ ДАННЫХ**

**ВЫПОЛНЕНИЕ: запросы**

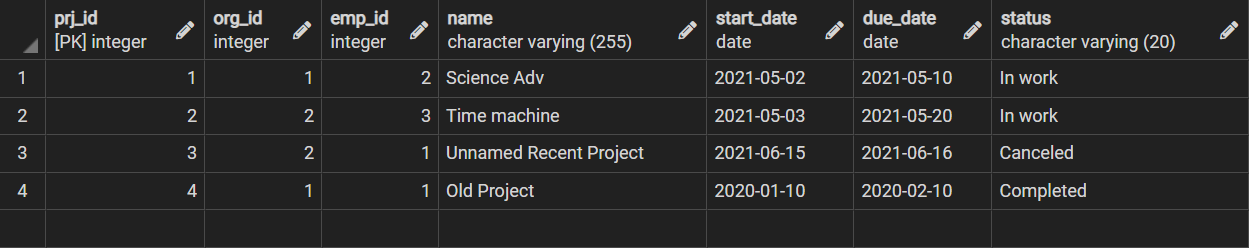
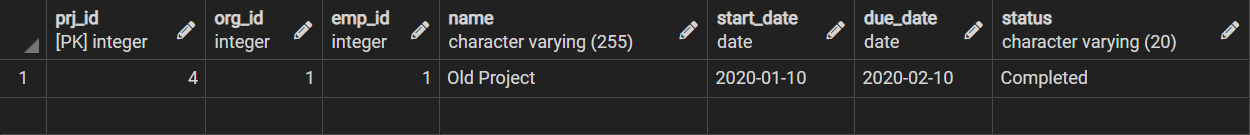
1. Составить список всех заданий каждого проекта с указанием организаций, отделов и исполнителей, занятых в его выполнении.

|  |
| --- |
| SELECT   PROJECT.prj\_id,  PROJECT.name as "Название проекта",   TASK.task\_id,  TASK.description as "Задание",  CONCAT(EMPLOEE.first\_name, ' ', EMPLOEE.last\_name) as "Исполнитель",  DEP.name as "Департамент",  ORG.name as "Организация"   FROM todo\_v1.task TASK  INNER JOIN todo\_v1.project PROJECT ON TASK.prj\_id = PROJECT.prj\_id INNER JOIN todo\_v1.emploee EMPLOEE ON TASK.emp\_id = EMPLOEE.emp\_id INNER JOIN todo\_v1.organization ORG ON PROJECT.org\_id = ORG.org\_id INNER JOIN todo\_v1.department DEP ON EMPLOEE.dep\_id = DEP.dep\_id ORDER BY PROJECT.prj\_id, TASK.task\_id; |



1. Составить список проектов, работа над которыми была начата больше месяца назад.

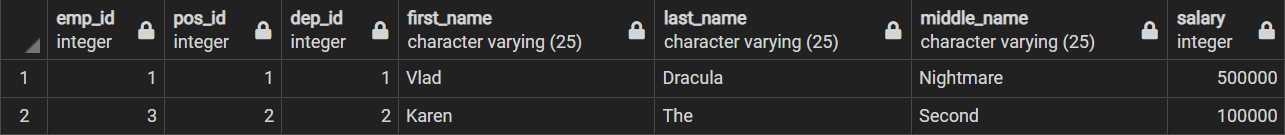
|  |
| --- |
| SELECT \* FROM todo\_v1.project WHERE start\_date < (NOW() - INTERVAL '1 month'); |

До:  
После:  


1. Вывести список сотрудников, оклад которых превышает средний оклад сотрудников своего отдела

|  |
| --- |
| SELECT emp\_id, EMP.pos\_id, dep\_id, first\_name, last\_name, middle\_name, POS.salary FROM todo\_v1.emploee EMP JOIN todo\_v1.position POS ON EMP.pos\_id = POS.pos\_id WHERE POS.salary > (  SELECT AVG(salary)   FROM todo\_v1.position POS  JOIN todo\_v1.emploee EMP\_1 ON EMP\_1.pos\_id = POS.pos\_id   WHERE EMP\_1.dep\_id = EMP.dep\_id ) |

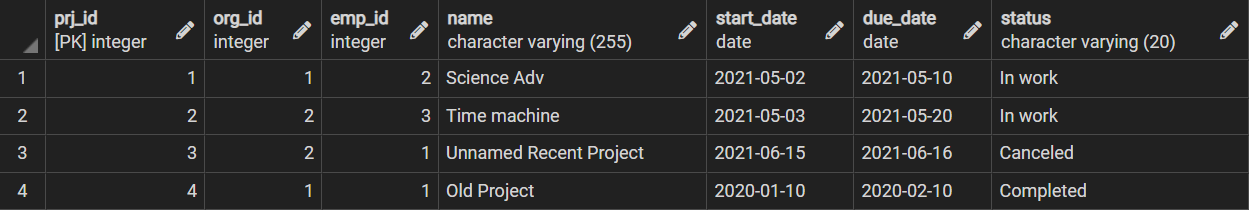
До:  
  
После:

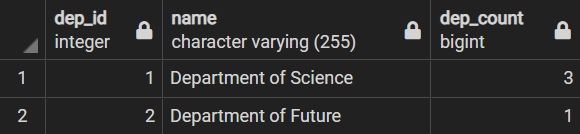


1. Найти отдел, работающий над максимальным количеством проектов.

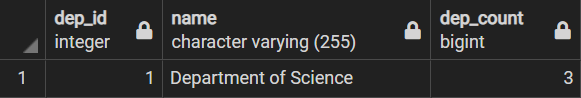
|  |
| --- |
| SELECT dep\_prj\_count.dep\_id, dep\_prj\_count.name, dep\_prj\_count.dep\_count FROM (  SELECT EMPLOEE.dep\_id, DEPARTMENT.name, COUNT(EMPLOEE.dep\_id) as dep\_count  FROM todo\_v1.project PROJECT  JOIN todo\_v1.emploee EMPLOEE ON EMPLOEE.emp\_id = PROJECT.emp\_id  JOIN todo\_v1.department DEPARTMENT ON DEPARTMENT.dep\_id = EMPLOEE.dep\_id   GROUP BY EMPLOEE.dep\_id, DEPARTMENT.name ) dep\_prj\_count  WHERE dep\_prj\_count.dep\_count IN (  SELECT MAX(count\_table.count) FROM (  SELECT COUNT(EMPLOEE.dep\_id)  FROM todo\_v1.project PROJECT  JOIN todo\_v1.emploee EMPLOEE ON EMPLOEE.emp\_id = PROJECT.emp\_id  GROUP BY EMPLOEE.dep\_id  ) count\_table ) |

До:



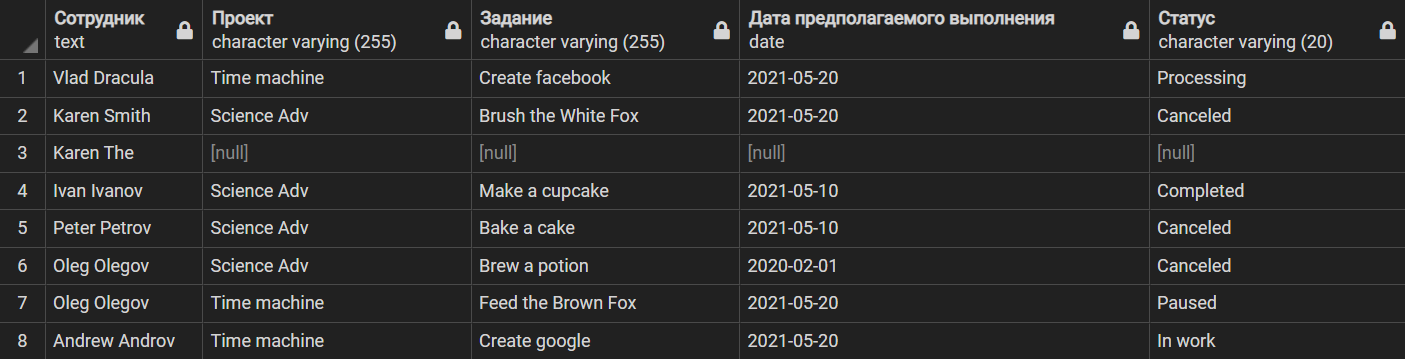


После:



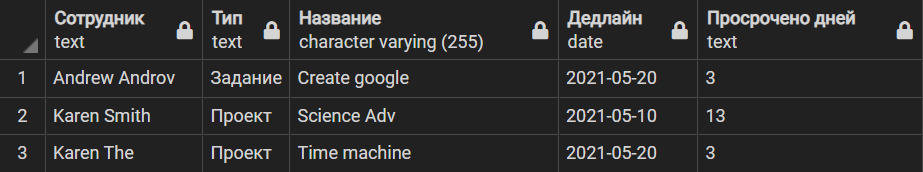
1. Составить список сотрудников, проектов, заданий, в выполнении которых они участвуют и дат предполагаемого выполнения ими заданий. Учесть сотрудников, не участвующих в проектах.

|  |
| --- |
| SELECT   CONCAT(EMP.first\_name, ' ', EMP.last\_name) as "Сотрудник",  PRJ.name as "Проект",  description as "Задание",  TASK.due\_date as "Дата предполагаемого выполнения",  TASK.status as "Статус" FROM todo\_v1.task TASK JOIN todo\_v1.project PRJ ON TASK.prj\_id = PRJ.prj\_id RIGHT JOIN todo\_v1.emploee EMP ON TASK.emp\_id = EMP.emp\_id |



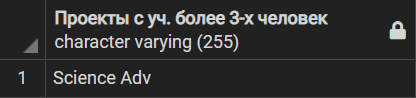
1. Составить список сотрудников, не выполнивших задания в срок с указанием проектов и заданий, которые они должны были выполнить и количества дней просрочки выполнения заданий

|  |
| --- |
| SELECT   CONCAT(EMP.first\_name, ' ', EMP.last\_name) as "Сотрудник",  CONCAT('Задание') as "Тип",  TASK.description as "Название",  TASK.due\_date as "Дедлайн",  CONCAT(NOW()::date - TASK.due\_date) as "Просрочено дней" FROM todo\_v1.task TASK JOIN todo\_v1.emploee EMP ON TASK.emp\_id = EMP.emp\_id WHERE   TASK.status = 'In work' AND TASK.due\_date < NOW()   UNION  SELECT   CONCAT(EMP.first\_name, ' ', EMP.last\_name) as "Сотрудник",  CONCAT('Проект') as "Тип",  PRJ.name as "Название",  PRJ.due\_date as "Дедлайн",  CONCAT(NOW()::date - PRJ.due\_date) as "Просрочено дней" FROM todo\_v1.task TASK JOIN todo\_v1.project PRJ ON TASK.prj\_id = PRJ.prj\_id JOIN todo\_v1.emploee EMP ON PRJ.emp\_id = EMP.emp\_id WHERE   PRJ.status = 'In work' AND PRJ.due\_date < NOW() |



1. Составить список проектов, в выполнении которого участвует более трех человек.

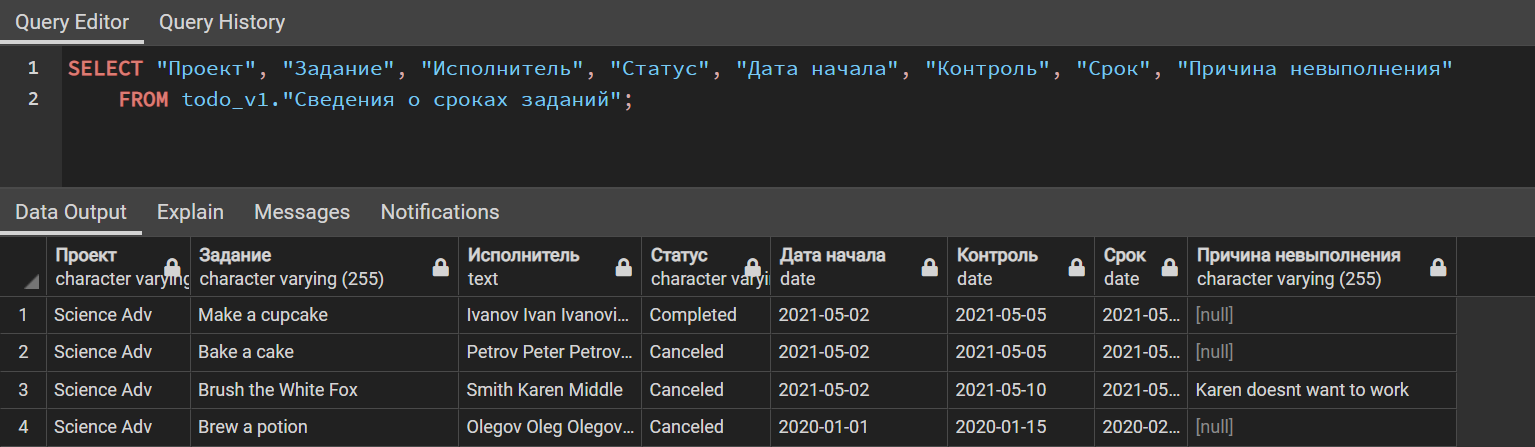
|  |
| --- |
| SELECT DISTINCT PRJ.name as "Проекты с уч. более 3-х человек" FROM todo\_v1.task TASK JOIN todo\_v1.project PRJ ON TASK.prj\_id = PRJ.prj\_id JOIN todo\_v1.emploee EMP ON TASK.emp\_id = EMP.emp\_id WHERE TASK.prj\_id IN (  SELECT prj\_id  FROM todo\_v1.task TASK  GROUP BY prj\_id  HAVING COUNT(emp\_id) > 3 ) ORDER BY PRJ.name |



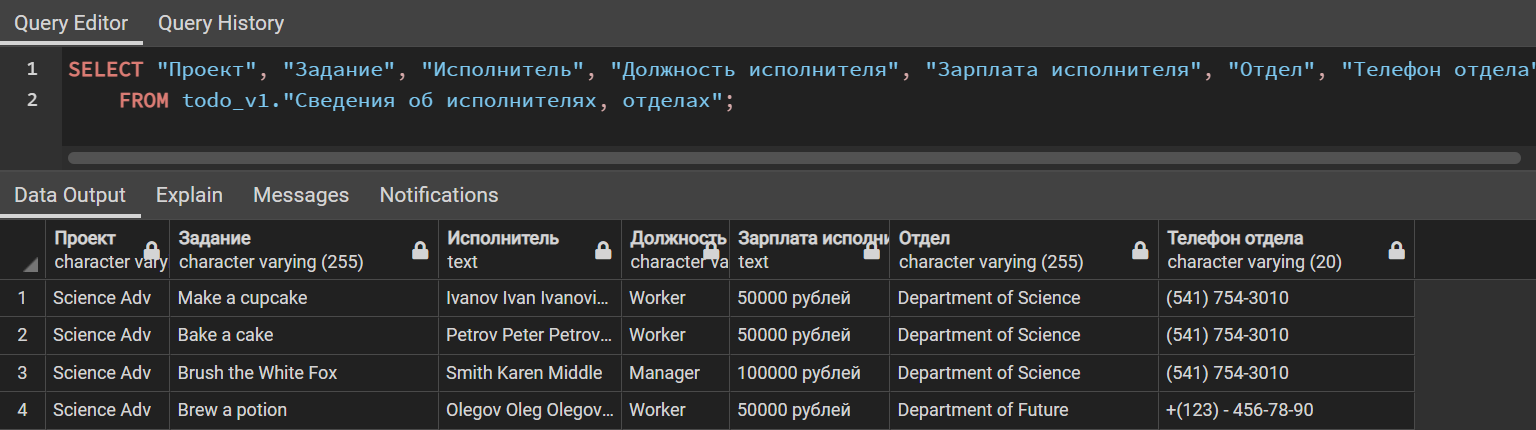
**ВЫПОЛНЕНИЕ: представления**

1. Для руководителей проектов, содержащее сведения об исполнителях, отделах, сроках выполнения заданий, включенных в проект.

|  |
| --- |
| CREATE OR REPLACE VIEW todo\_v1."Сведения о сроках заданий" AS SELECT   PRJ.name AS "Проект",  TASK.description AS "Задание",  CONCAT(EMP.last\_name, ' ', EMP.first\_name, ' ', EMP.middle\_name) AS "Исполнитель",  TASK.status AS "Статус",  TASK.start\_date AS "Дата начала",  TASK.due\_check\_date AS "Контроль",  TASK.due\_date AS "Срок",  TASK.failed\_to\_complete\_reason AS "Причина невыполнения" FROM todo\_v1.task TASK JOIN todo\_v1.project PRJ ON TASK.prj\_id = PRJ.prj\_id JOIN todo\_v1.emploee EMP ON TASK.emp\_id = EMP.emp\_id JOIN todo\_v1.department DEP ON EMP.dep\_id = DEP.dep\_id WHERE PRJ.emp\_id = 2 -- захардкожено, а должно динамически меняться в зависимости от руководителя ORDER BY PRJ.name |



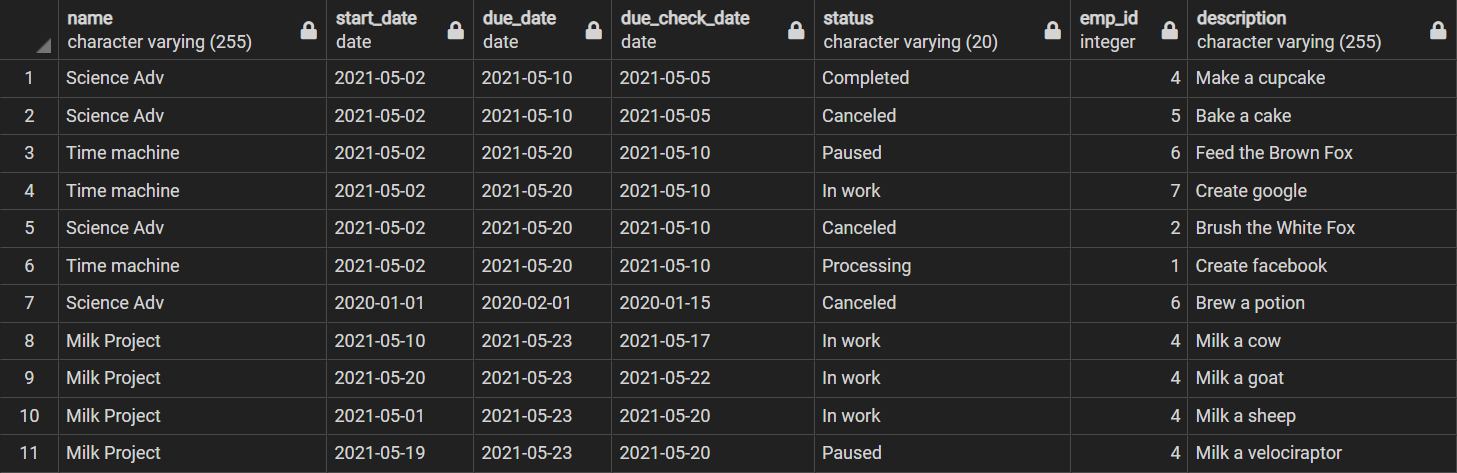
|  |
| --- |
| CREATE OR REPLACE VIEW todo\_v1."Сведения об исполнителях и отделах" AS SELECT   PRJ.name AS "Проект",  TASK.description AS "Задание",  CONCAT(EMP.last\_name, ' ', EMP.first\_name, ' ', EMP.middle\_name) AS "Исполнитель",  POS.name AS "Должность исполнителя",  CONCAT(POS.salary, ' рублей') AS "Зарплата исполнителя",  DEP.name AS "Отдел",  DEP.phone\_number AS "Телефон отдела" FROM todo\_v1.task TASK JOIN todo\_v1.project PRJ ON TASK.prj\_id = PRJ.prj\_id JOIN todo\_v1.emploee EMP ON TASK.emp\_id = EMP.emp\_id JOIN todo\_v1.department DEP ON EMP.dep\_id = DEP.dep\_id JOIN todo\_v1.position POS ON EMP.pos\_id = POS.pos\_id WHERE PRJ.emp\_id = 2 -- захардкожено, а должно динамически меняться в зависимости от руководителя ORDER BY PRJ.name |

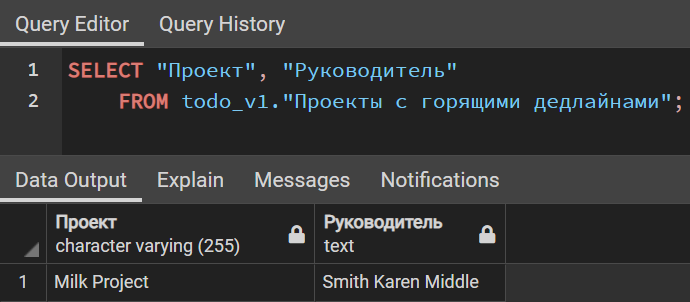


1. Список проектов, срок выполнения которых истекает сегодня, и которые включают больше трех невыполненных заданий.

|  |
| --- |
| CREATE OR REPLACE VIEW todo\_v1."Проекты с горящими дедлайнами" AS SELECT DISTINCT   PRJ.name AS "Проект",  CONCAT(EMP.last\_name, ' ', EMP.first\_name, ' ', EMP.middle\_name) AS "Руководитель" FROM todo\_v1.task TASK JOIN todo\_v1.project PRJ ON TASK.prj\_id = PRJ.prj\_id JOIN todo\_v1.emploee EMP ON PRJ.emp\_id = EMP.emp\_id WHERE TASK.prj\_id IN (  SELECT prj\_id  FROM todo\_v1.task TASK  WHERE TASK.status != 'Completed'  GROUP BY prj\_id  HAVING COUNT(task\_id) > 3 ) AND PRJ.due\_date = NOW()::date |

До:



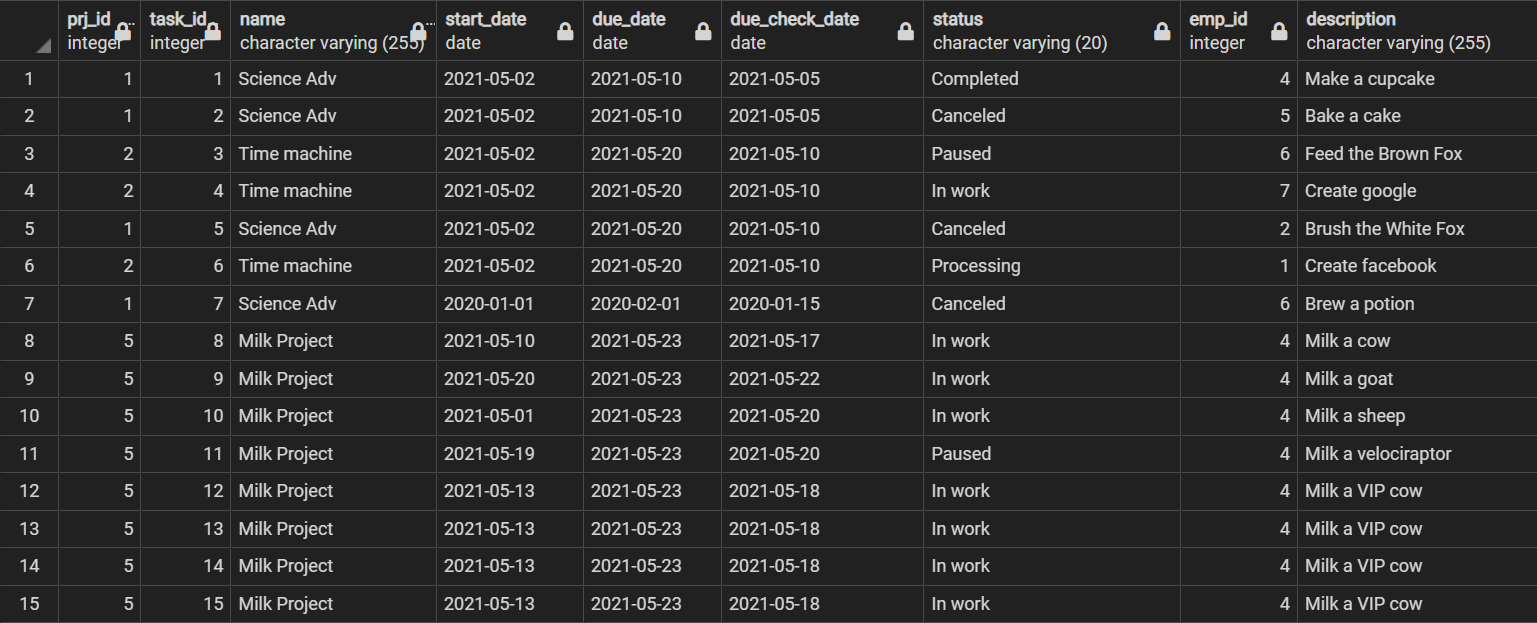
После:  


**ВЫПОЛНЕНИЕ: модификация данных**

1. Запрос с INSERT

Запрос для быстрого добавления заданий в проект “Milk Project”, использующий сроки и исполнителей из уже существующих заданий

|  |
| --- |
| WITH milk\_prj\_default\_date AS (  SELECT TASK.due\_date FROM todo\_v1.task TASK  JOIN todo\_v1.project PRJ ON TASK.prj\_id = PRJ.prj\_id  WHERE PRJ.name = 'Milk Project' LIMIT 1 )  INSERT INTO todo\_v1.task(  task\_id,  prj\_id,  start\_date,  due\_date,  due\_check\_date,  status,  emp\_id,  description ) VALUES (  (SELECT (MAX(task\_id) + 1) FROM todo\_v1.task),  (SELECT prj\_id FROM todo\_v1.project WHERE name = 'Milk Project'),  (SELECT \* FROM milk\_prj\_default\_date) - INTERVAL '10 days',  (SELECT \* FROM milk\_prj\_default\_date),  (SELECT \* FROM milk\_prj\_default\_date) - INTERVAL '5 days',  'In work',  (SELECT emp\_id FROM todo\_v1.task LIMIT 1),  'Milk a VIP cow' ) |

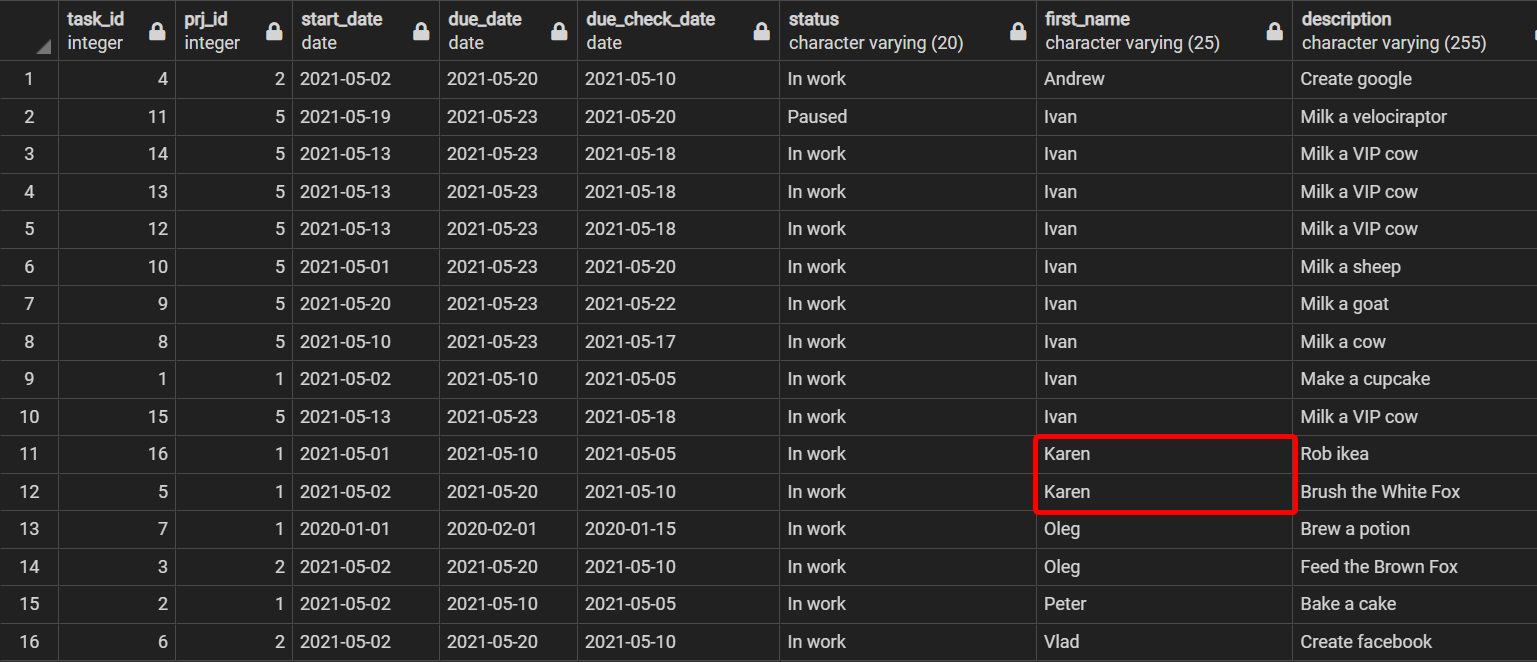
****

1. Запрос с UPDATE

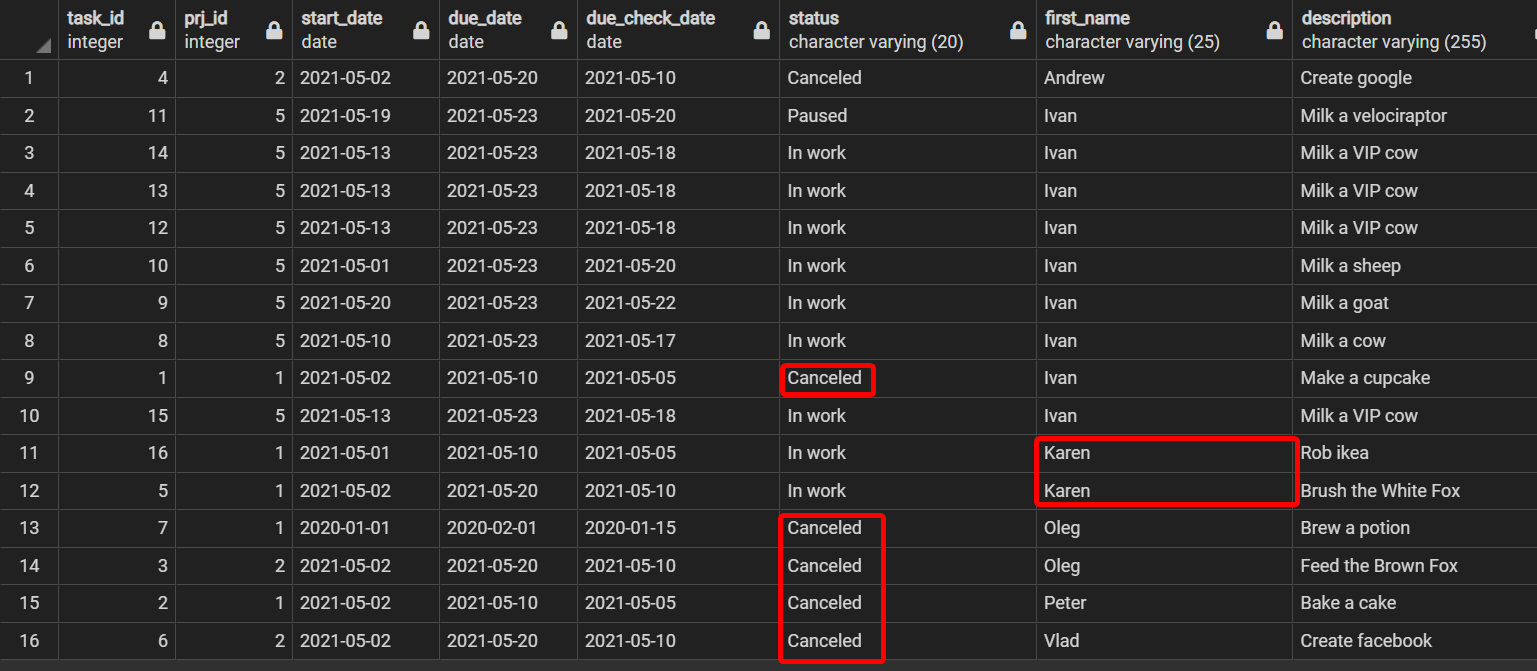
Изменение статуса заданий с истекшим сроком выполнения cо статуса “In work” на “Canceled”, кроме тех, над которыми работает исполнитель с именем “Karen”

|  |
| --- |
| UPDATE todo\_v1.task SET status='Canceled' WHERE NOW()::date > due\_date AND status = 'In work' AND task\_id NOT IN (  SELECT TASK.task\_id FROM todo\_v1.task TASK  JOIN todo\_v1.emploee EMP ON TASK.emp\_id = EMP.emp\_id  WHERE first\_name = 'Karen' ) |

До:



После:

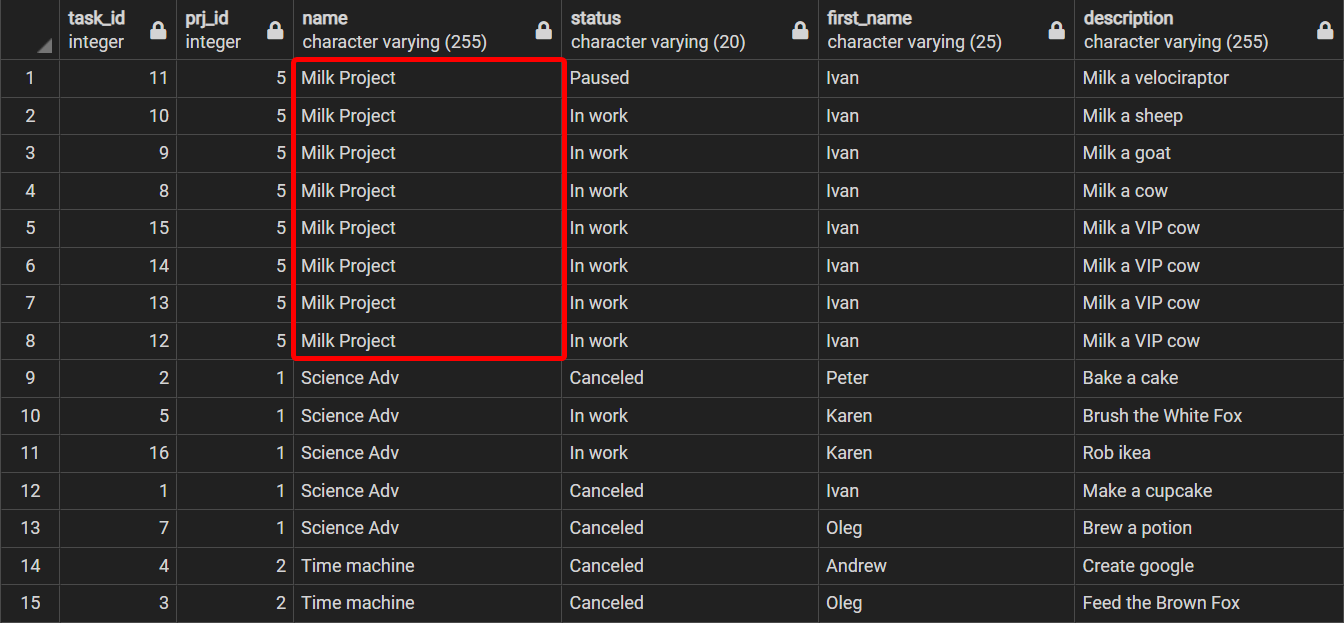
****

1. Запрос с DELETE

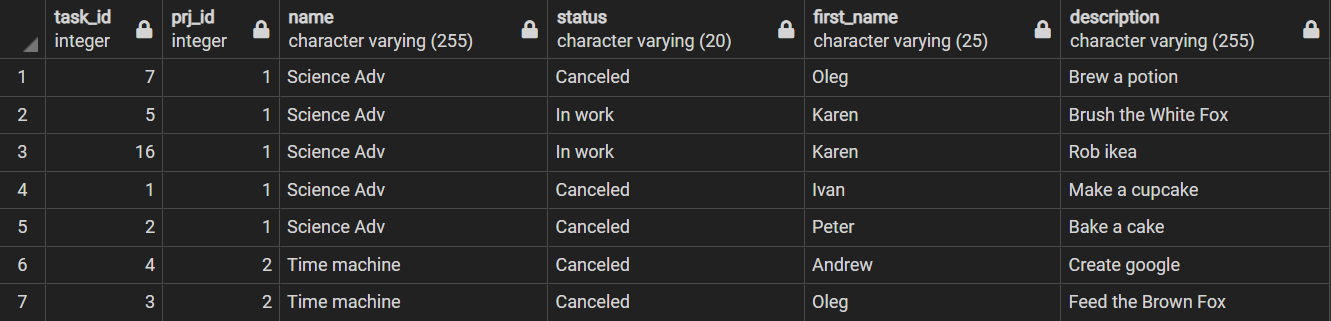
Удалить все задания связанные с проектом “Milk Project”

|  |
| --- |
| DELETE FROM todo\_v1.task TASK WHERE TASK.prj\_id = (SELECT prj\_id FROM todo\_v1.project WHERE name = 'Milk Project') |

До:



После:



**ВЫВОДЫ**

Понимание механизма выборки данных на низком уровне даст более глубокое понимание устройства реляционных БД, что, в свою очередь, позволит использовать уже высокоуровневые концепты более гибко.

Благодаря тому, что из этих концептов исчезает “магия”, то необычная ошибка или прецедент использования с меньшей вероятностью застанут программиста врасплох.