Министерство науки и высшего образования Российской Федерации
федеральное государственное автономное образовательное учреждение высшего образования
«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Отчет

по лабораторной работе №4 «ЗАПРОСЫ НА ВЫБОРКУ И МОДИФИКАЦИЮ ДАННЫХ. ПРЕДСТАВЛЕНИЯ. РАБОТА С ИНДЕКСАМИ»

по дисциплине **«Проектирование и реализация баз данных»**

Автор: Федорин К.В.

Факультет: ИКТ

Группа: К3240

Преподаватель: Говорова М.М.

Санкт-Петербург 2023

ЛАБОРАТОРНАЯ РАБОТА №4

Цель работы: овладеть практическими навыками создания представлений и запросов на выборку данных к базе данных PostgreSQL, использования подзапросов при модификации данных и индексов.

Оборудование: компьютерный класс.

Программное обеспечение: СУБД PostgreSQL, pgadmin 4.

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 и просмотреть историю запросов.
4. Создать простой и составной индексы для двух произвольных запросов и сравнить время выполнения запросов без индексов и с индексами. Для получения плана запроса использовать команду EXPLAIN.

Вариант 4. Управление задачами

Создание запросов

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

```
with contract_company as (select "ContractID", "CompanyName" from "CompanyContract" join "CustomerCompany"
                    ON "CustomerCompany"."CustomerCompanyID" = "CompanyContract"."CustomerCompanyID"),

project_company as (select "ProjectName", "ProjectID", "CompanyName" from "Project" as p join
                    contract_company as cc on cc."ContractID" = p."ContractID"),

workerc_department as (select "ContractID", w."DepartmentID", "DepartmentName", "WorkerID"
                      from "WorkerOnContract" as w join "Department" as d on w."DepartmentID" = d."DepartmentID"),

worker_contract as (select w."WorkerID", "ContractID", "FIO", "DepartmentID", "DepartmentName"
                    from workerc_department wd join "Worker" w on wd."WorkerID" = w."WorkerID"),

worker_ass as (select wa."ContractID", "AssignmentId", "WorkerID", "FIO", "DepartmentID", "DepartmentName"
               from  "WorkerAssignment" wa join worker_contract wc on wc."ContractID" = wa."ContractID"),

task_ass as (select t."TaskID", "AssignmentID", "TaskName", "ProjectID", "Description"
             from "Task" t join "TaskAssignment" ON "TaskAssignment"."TaskID" = t."TaskID"),

worker_task as (select * from task_ass ta join worker_ass wa on ta."AssignmentID" = wa."AssignmentId"),

worker_task_project_company as (select "TaskID", "TaskName", "Description", "ContractID",
                                "WorkerID", "FIO", "DepartmentID", "DepartmentName",
                                pc."ProjectID", "CompanyName"
                                from worker_task wa join project_company pc on wa."ProjectID" = pc."ProjectID")

select * from worker_task_project_company;
```

Data Output | Messages | Notifications

	TaskID integer	TaskName character varying	Description character varying	ContractID integer	WorkerID integer	FIO character varying	DepartmentID integer	DepartmentName character varying	ProjectID integer	CompanyName character varying (100)
1	2	Develope Backend	Do smth2	14	14	Popov Ivan Holmovich	123121	Amazon	1232	Google
2	2	Develope Backend	Do something	14	14	Popov Ivan Holmovich	123121	Amazon	1232	Google
3	1	Develope Frontend	Do smth	13	13	Vasiliy Ivanov Ivanovich	123123	Yandex	1231	Google

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

```
select * from "Project" where "StartDate" < CURRENT_DATE - interval '1 month';
```

Data Output | Messages | Notifications

	ProjectID [PK] integer	ProjectName character varying	CompletionStatus character varying	StartDate date	EndDate date	PaymentStatus character varying	TeamleaderID integer	ContractID integer
1	1231	Yandex Map	в работе	2003-12-31	2040-12-31	оплачен	13	1
2	1232	Amazon Web	не начат	2003-12-31	2060-12-31	частично оплачен	14	1

3) Вывести список сотрудников, оклад которых превышает средний оклад сотрудников своего отдела.

```
with worker as (select w."WorkerID", "FIO", "SalaryPercentrage", "PositionID", "DepartmentID"
                from "Worker" w join "WorkerOnContract" woc on woc."WorkerID" = w."WorkerID"),
worker_job as (select *, "SalaryPercentrage" * "Salary" / 100 as income from worker join "Position" p on p."PositionID" = worker."PositionID"),
avg_dep as (select "DepartmentID", AVG(income) as average_income  from worker_job group by "DepartmentID")

select "WorkerID", "FIO", income, wj."DepartmentID", "PositionName" from worker_job wj join avg_dep ad on
wj."DepartmentID" = ad."DepartmentID" where income >= average_income;
```

Data Output | Messages | Notifications

	WorkerID integer	FIO character varying	income double precision	DepartmentID integer	PositionName character varying
1	13	Vasiliy Ivanov Ivanovich	30000	123123	Python Backend
2	14	Popov Ivan Holmovich	300000	123121	DevOPS

4) Найти отдел, работающий над максимальным количеством проектов

```
with worker as (select w."WorkerID", "FIO", "PositionID", "DepartmentID" , "ContractID"
                from "Worker" w join "WorkerOnContract" woc on woc."WorkerID" = w."WorkerID"),
worker_ass as (select * from worker w join "WorkerAssignment" wa on w."ContractID" = wa."ContractID"),
dep_proj as (select "DepartmentID", COUNT(DISTINCT "ProjectID") as project_count from worker_ass group by "DepartmentID")

select * from dep_proj where project_count = (select max(project_count) from dep_proj);
```

Data Output | Messages | Notifications

	DepartmentID integer	project_count bigint
1	123123	2

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

```
with worker as (select w."WorkerID", "FIO", "SalaryPercentrage", "PositionID", "DepartmentID", "ContractID"
                from "Worker" w left join "WorkerOnContract" woc on woc."WorkerID" = w."WorkerID"),
ded_w as (select "WorkerID", "FIO", w."ContractID", "DeadlineDate", "TaskID" from worker w
          left join "TaskCompletionControl" tcc on w."ContractID" = tcc."ContractID"),
task_proj as (select "TaskID", "TaskName", "TaskDescription", t."ProjectID", "ProjectName"
              from "Task" t join "Project" p on t."ProjectID" = p."ProjectID"
              where t."Status"='в работе'),
cur_projects as (select "WorkerID", "FIO", "ContractID", "DeadlineDate",
                 d."TaskID", "TaskName", "TaskDescription", "ProjectID", "ProjectName"
                 from ded_w d join task_proj tp on d."TaskID" = tp."TaskID")


select w."WorkerID", w."FIO", "DeadlineDate", cp."TaskID", "TaskName", "TaskDescription", "ProjectID", "ProjectName"
from "Worker" w left join cur_projects cp on w."WorkerID" = cp."WorkerID";

```

Data Output | Messages | Notifications

	WorkerID integer	FIO character varying	DeadlineDate date	TaskID integer	TaskName character varying	TaskDescription character varying	ProjectID integer	ProjectName character varying
1	14	Popov Ivan Holmovich	2022-12-31	2	Develope Backend	Use django	1232	Amazon Web
2	13	Vasiliy Ivanov Ivanovich	[null]	[null]	[null]	[null]	[null]	[null]
3	12	Rorik Pogov Lolovich	[null]	[null]	[null]	[null]	[null]	[null]

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

```
with worker as (select w."WorkerID", "FIO", "SalaryPercentrage", "PositionID", "DepartmentID", "ContractID"
                from "Worker" w join "WorkerOnContract" woc on woc."WorkerID" = w."WorkerID"),
ded_w as (select "WorkerID", "FIO", w."ContractID", "DeadlineDate", "TaskID" from worker w
          join "TaskCompletionControl" tcc on w."ContractID" = tcc."ContractID"),
task_proj as (select "TaskID", "TaskName", "TaskDescription", t."ProjectID", "ProjectName", "Status"
              from "Task" t join "Project" p on t."ProjectID" = p."ProjectID")


select "WorkerID", "FIO", "ContractID", "DeadlineDate", tp."TaskID",
"TaskName", "TaskDescription", "Status", "ProjectID", "ProjectName",
CURRENT_DATE - "DeadlineDate" as prosrok

from ded_w d join task_proj tp on d."TaskID" = tp."TaskID"
where not "Status" = 'закончен' and current_date > "DeadlineDate";

```

Data Output | Messages | Notifications

	WorkerID integer	FIO character varying	ContractID integer	DeadlineDate date	TaskID integer	TaskName character varying	TaskDescription character varying	Status character varying	ProjectID integer	ProjectName character varying	prosrok integer
1	14	Popov Ivan Holmovich	14	2022-12-31	2	Develope Backend	Use django	в работе	1232	Amazon Web	333

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

```sql
select  wa."ProjectID", "ProjectName", count(wa."ContractID") as people
from "WorkerAssignment" wa join "Project" p
on wa."ProjectID" = p."ProjectID"
group by wa."ProjectID", "ProjectName"
having count(wa."ContractID") > 3
order by people desc;
```

Data Output | Messages | Notifications

	ProjectID integer	ProjectName character varying	people bigint
1	1232	Amazon Web	4

Создание представлений (View)

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

```
create view project_task_deadline as
with contract_company as (select "ContractID", "CompanyName" from "CompanyContract" join "CustomerCompany"
                    ON "CustomerCompany"."CustomerCompanyID" = "CompanyContract"."CustomerCompanyID"),

project_company as (select "ProjectName", "ProjectID", "CompanyName" from "Project" as p join
                    contract_company as cc on cc."ContractID" = p."ContractID"),

workerc_department as (select "ContractID", w."DepartmentID", "DepartmentName", "WorkerID"
                       from "WorkerOnContract" as w join "Department" as d on w."DepartmentID" = d."DepartmentID"),

worker_contract as (select w."WorkerID", "ContractID", "FIO", "DepartmentID", "DepartmentName"
                    from workerc_department wd join "Worker" w on wd."WorkerID" = w."WorkerID"),

worker_ass as (select wa."ContractID", "AssignmentId", "WorkerID", "FIO", "DepartmentID", "DepartmentName"
               from  "WorkerAssignment" wa join worker_contract wc on wc."ContractID" = wa."ContractID"),

task_ass as (select t."TaskID", "AssignmentID", "TaskName", "ProjectID", "Description"
             from "Task" t join "TaskAssignment" ON "TaskAssignment"."TaskID" = t."TaskID"),

worker_task as (select "TaskID", wa."AssignmentId", "TaskName", "ProjectID", "Description", "ContractID",
                "WorkerID", "FIO", "DepartmentID", "DepartmentName"
                from task_ass ta join worker_ass wa on ta."AssignmentID" = wa."AssignmentId"),
worker_task_ded as (select wt."TaskID", "TaskName", wt."AssignmentId",  "Description", "ProjectID", wt."ContractID", "WorkerID",
                    "FIO", "DepartmentID", "DepartmentName", "DeadlineDate", "DeadlineCompletion", "DeadlineID"
                    from worker_task wt join "TaskCompletionControl" tcc on tcc."TaskID"=wt."TaskID")

select * from worker_task_ded;
```

```
with contract_company as (select "ContractID", "CompanyName" from "CompanyContract" join "CustomerCompany"
                    ON "CustomerCompany"."CustomerCompanyID" = "CompanyContract"."CustomerCompanyID"),

project_company as (select "ProjectName", "ProjectID", "CompanyName" from "Project" as p join
                    contract_company as cc on cc."ContractID" = p."ContractID"),

workerc_department as (select "ContractID", w."DepartmentID", "DepartmentName", "WorkerID"
                       from "WorkerOnContract" as w join "Department" as d on w."DepartmentID" = d."DepartmentID"),

worker_contract as (select w."WorkerID", "ContractID", "FIO", "DepartmentID", "DepartmentName"
                    from workerc_department wd join "Worker" w on wd."WorkerID" = w."WorkerID"),

worker_ass as (select wa."ContractID", "AssignmentId", "WorkerID", "FIO", "DepartmentID", "DepartmentName"
               from  "WorkerAssignment" wa join worker_contract wc on wc."ContractID" = wa."ContractID"),

task_ass as (select t."TaskID", "AssignmentID", "TaskName", "ProjectID", "Description"
             from "Task" t join "TaskAssignment" ON "TaskAssignment"."TaskID" = t."TaskID"),

worker_task as (select "TaskID", wa."AssignmentId", "TaskName", "ProjectID", "Description", "ContractID",
                "WorkerID", "FIO", "DepartmentID", "DepartmentName"
                from task_ass ta join worker_ass wa on ta."AssignmentID" = wa."AssignmentId"),
worker_task_ded as (select wt."TaskID", "TaskName", wt."AssignmentId",  "Description", "ProjectID", wt."ContractID", "WorkerID",
                    "FIO", "DepartmentID", "DepartmentName", "DeadlineDate", "DeadlineCompletion", "DeadlineID"
                    from worker_task wt join "TaskCompletionControl" tcc on tcc."TaskID"=wt."TaskID")

select * from worker_task_ded;
```

Data Output | Messages | Notifications

	TaskID integer	TaskName character varying	AssignmentId integer	Description character varying	ProjectID integer	ContractID integer	WorkerID integer	FIO character varying	DepartmentID integer	DepartmentName character varying	DeadlineDate date	DeadlineCompletion integer	DeadlineID integer
1	2	Develope Backend	2	Do smth2	1232	14	14	Popov Ivan Holmovich	123121	Amazon	2022-12-31	50	2
2	2	Develope Backend	2	Do something	1232	14	14	Popov Ivan Holmovich	123121	Amazon	2022-12-31	50	2
3	1	Develope Frontend	1	Do smth	1231	13	13	Vasiliy Ivanov Ivanovich	123123	Yandex	2023-11-29	95	1
4	1	Develope Frontend	1	Do smth	1231	13	13	Vasiliy Ivanov Ivanovich	123123	Yandex	2023-11-29	40	3
5	1	Develope Frontend	1	Do smth	1231	13	13	Vasiliy Ivanov Ivanovich	123123	Yandex	2023-11-29	80	4

2) список проектов, срок исполнения которых истекает сегодня и которые включают больше трех невыполненных заданий.

```sql
with task_proj as (select "TaskID", "TaskName", "TaskDescription", t."ProjectID", "ProjectName"
                from "Task" t join "Project" p on t."ProjectID" = p."ProjectID")


select "ProjectID", "ProjectName" as task_count
from "TaskCompletionControl" tcc join task_proj tp on tcc."TaskID" = tp."TaskID"
where "DeadlineDate" = CURRENT_DATE
group by "ProjectID", "ProjectName"
having count(tp."TaskID")>=3;


```

Data Output | Messages | Notifications

	ProjectID integer	task_count character varying
1	1231	Yandex Map

```sql
create view burning_projects as
select "ProjectID", "ProjectName" as task_count
from "TaskCompletionControl" tcc join (select "TaskID", "TaskName", "TaskDescription", t."ProjectID", "ProjectName"
                from "Task" t join "Project" p on t."ProjectID" = p."ProjectID") tp on tcc."TaskID" = tp."TaskID"
where "DeadlineDate" = CURRENT_DATE
group by "ProjectID", "ProjectName"
having count(tp."TaskID")>=3;


```

Data Output | Messages | Notifications

```
CREATE VIEW

Query returned successfully in 32 msec.
```

Модификация данных

1) Создание нового задания, с привязкой к проекту по его имени
   Запрос:

```
insert into "Task" ("TaskID", "ProjectID", "TaskName", "TaskDescription", "Cost", "Status")
values (30, (select "ProjectID" from "Project" where "ProjectName" like 'Amazon Web'),
        'Develop frontend', 'Use figma and react.js', 15000, 'не начат')
```

До:

	TaskID [PK] integer	TaskName character varying	TaskDescription character varying	Cost integer	Status character varying	ProjectID integer
1	1	Develope Frontend	Develope React Frontend	30000	не начат	1231
2	2	Develope Backend	Use django	3000	в работе	1232

После:

	TaskID [PK] integer	TaskName character varying	TaskDescription character varying	Cost integer	Status character varying	ProjectID integer
1	1	Develope Frontend	Develope React Frontend	30000	не начат	1231
2	2	Develope Backend	Use django	3000	в работе	1232
3	30	Develop frontend	Use figma and react.js	15000	не начат	1232

2) Удаление задания по его названию и имени назначенного на него сотрудника.

Запрос:

```
delete from "TaskAssignment" where
    "TaskID" = (select "TaskID" from "Task" where "TaskName" ='Develope Backend')
    and "AssignmentID" = (select "AssignmentId" from
        (select "AssignmentId" from "WorkerAssignment" wa join
         (select "ContractID" from "WorkerOnContract" wc
           join "Worker" w on w."WorkerID" = wc."WorkerID"
           where "FIO" like 'Popov%') fio_contract
         on fio_contract."ContractID"=wa."ContractID") as assign);

```

Data Output | Messages | Notifications

```
DELETE 2

Query returned successfully in 69 msec.
```

До:

	AssignmentID integer	TaskID integer	Id [PK] integer	Status character varying (100)	StartDate date	EndDate date	Description character varying
1	2	2	2	Не начат	2023-08-11	2023-08-24	do smth2
2	1	1	1	В работе	2023-08-11	2023-08-24	Do smth

После:

	AssignmentID integer	TaskID integer	Id [PK] integer	Status character varying (100)	StartDate date	EndDate date	Description character varying
1	1	1	1	В работе	2023-08-11	2023-08-24	Do smth

3) Повышение зарплаты сотруднику по его ФИО

Запрос:

```
update "WorkerOnContract" set "SalaryPercentrage" = 120
where "WorkerID" = (select "WorkerID" from "Worker" where "FIO" like 'Popov%')
```

До:

	ContractID [PK] integer	SalaryPercentrage double precision	WorkerID integer	PositionID integer	DepartmentID integer
1	1	50	13	12331	123123
2	13	100	13	12331	123123
3	14	100	14	12333	123121

После:

	ContractID [PK] integer	SalaryPercentrage double precision	WorkerID integer	PositionID integer	DepartmentID integer
1	1	50	13	12331	123123
2	13	100	13	12331	123123
3	14	120	14	12333	123121

Индекс

С индексом:

Query Query History

```
1  --create index idx_task on "Task" ("Cost");
2  select * from "Task" where "Cost" between 10000 and 30000;
```

Data Output Messages Explain × Notifications

```
Successfully run. Total query runtime: 55 msec.
1 rows affected.
```

Без:

```
1  --drop index idx_task;
2  select * from "Task" where "Cost" between 10000 and 30000;
```

Data Output Messages Explain × Notifications

```
Successfully run. Total query runtime: 44 msec.
1 rows affected.
```

Data Output Messages Explain × Notifications

Graphical Analysis Statistics

#	Node
1.	→ Seq Scan on Task as Task **Filter**: (("Cost" >= 10000) AND ("Cost" <= 30000))

Graphical Analysis Statistics

Statistics per Node Type

Node type	Count
Seq Scan	1

Statistics per Relation

Relation name	Scan count
Node type	Count
Task	1
Seq Scan	1

Индекс на несколько ключей:

```
1  create index idx_task on "Task" ("TaskName", "TaskDescription");
2  --select * from "Task" where "Cost" between 10000 and 30000;
```

Data Output | Messages | Explain × | Notifications

```
CREATE INDEX

Query returned successfully in 58 msec.
```

С индексом:

```
1  --create index idx_task on "Task" ("TaskName", "TaskDescription");
2  select * from "Task" where "TaskName" like 'Develop%' and "TaskDescription" like 'Use%';
```

Data Output | Messages | Explain × | Notifications

```
Successfully run. Total query runtime: 53 msec.
2 rows affected.
```

Без:

```
1  --drop index idx_task
2  select * from "Task" where "TaskName" like 'Develop%' and "TaskDescription" like 'Use%';
```

Data Output | Messages | Explain × | Notifications

```
Successfully run. Total query runtime: 56 msec.
2 rows affected.
```

Data Output | Messages | Explain × | Notifications

Graphical | Analysis | Statistics

#	Node
1.	→ Seq Scan on Task as Task **Filter**: ((("TaskName")::text ~~ 'Develop%'::text) AND (("TaskDescription")::text ~~ 'Use%'::text))

Вывод:

В ходе лабораторной работы были освоены практические навыки по созданию, запросов к базе данных в PostgreSQL с использованием инструмента управления pgAdmin4. Были созданы запросы на выборку, обновление, вставку и удаление, а также представления и индексы а также проанализировано их влияние на скорость выполнения запросов. Результаты показали что запросы с индексами работают быстрее, чем без них.