МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное образовательное учреждение высшего образования

«САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ   
АЭРОКОСМИЧЕСКОГО ПРИБОРОСТРОЕНИЯ»

КАФЕДРА КОМПЬЮТЕРНЫХ ТЕХНОЛОГИЙ И ПРОГРАММНОЙ ИНЖЕНЕРИИ

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Старший преподаватель |  |  |  | Н.В. Путилова |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №10 |
| Объектно-реляционные базы данных.  Манипуляция данными и пользовательские операторы |
| по курсу: Проектирование баз данных |
|  |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. № | 4932 |  |  |  | Н.С. Иванов |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2022

#### Текст задания

1)Выполнить вставку тестовых данных в таблицы, созданные в ходе выполнения лабораторной работы 9.

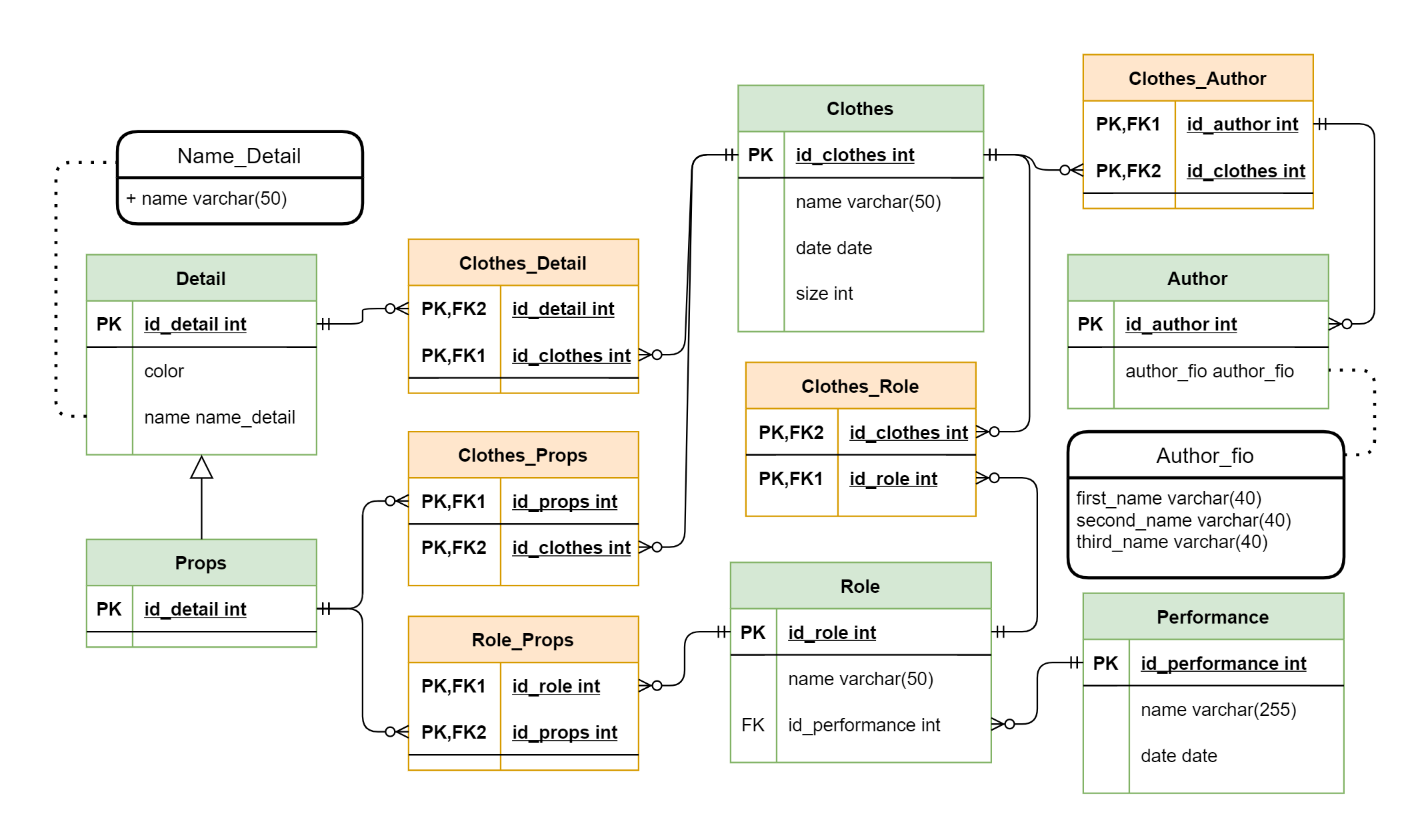
2)Придумать (задание) и реализовать запрос выборки с условием к таблицам и предку и потомку.

Придумать и реализовать запросы выборки с условием к таблице только к предку и только к потомку.

3)Придумать и создать пользовательский оператор для своей предметной области

4)Придумать и создать пользовательскую агрегатную функцию для своей предметной области

#### Физическая модель



#### Вставка тестовых данных в таблицы

SET search\_path TO theatre;  
  
insert into performance (*date*, *name*)  
 values  
 ('2021-10-11', 'Представление с шутом' ),  
 ('2021-10-12', 'Представление с королем' ),  
 ('2021-10-13', 'Представление (без ролей и одежды)' ),  
 ('2021-10-18', 'Золушка' ),  
 ('2021-10-19', 'Мастер и Маргарита' ),  
 ('2021-10-20', 'Представление (с ролями и без одежды)' );  
  
  
insert into author (*author\_fio*)  
 values  
 ('(Имя 1, Фамилия 1, Отчество 1)'),  
 ('(Имя 2, Фамилия 2, Отчество 2)'),  
 ('(Имя 3, Фамилия 3, Отчество 3)'),  
 ('(Имя 4, Фамилия 4, Отчество 4)');  
  
  
insert into clothes (*name*, *date*, *size*)  
 values  
 ('Шут', '2020-12-11', 30),  
 ('Маленький король', '2020-12-14', 30),  
 ('Платье Золушки', '2020-12-10', 25),  
 ('Костюм Мастера', '2020-12-13', 30),  
 ('Костюм Маргариты', '2020-12-09', 25);  
  
  
insert into role (*id\_performance*, *name*)  
 values  
 ((select *id\_performance* from performance where *name* like 'Представление с шутом'),  
 'Роль шута'),  
  
 ((select *id\_performance* from performance where *name* like 'Представление с королем'),  
 'Роль короля'),  
  
 ((select *id\_performance* from performance where *name* like 'Золушка'),  
 'Золушка'),  
  
 ((select *id\_performance* from performance where *name* like 'Мастер и Маргарита'),  
 'Мастер'),  
  
 ((select *id\_performance* from performance where *name* like 'Мастер и Маргарита'),  
 'Маргарита'),  
  
 ((select *id\_performance* from performance where *name* like 'Представление (с ролями и без одежды)'),  
 'Роль для пустого представления');  
  
  
insert into detail (*name*, *color*)  
 values  
 ('(Куртка)', 'Черный' ),  
 ('(Куртка)', 'Белый' ),  
 ('(Куртка)', 'Серый' ),  
 ('(Штаны)', 'Черный' ),  
 ('(Штаны)', 'Белый' ),  
 ('(Штаны)', 'Серый' ),  
 ('(Шляпа)', 'Черный' ),  
 ('(Шляпа)', 'Красный' ),  
 ('(Шляпа)', 'Желтый' ),  
 ('(Шляпа)', 'Белый' ),  
 ('(Шляпа)', 'Серый' ),  
 ('(Брюки)', 'Черный' ),  
 ('(Брюки)', 'Белый' ),  
 ('(Брюки)', 'Серый' ),  
 ('(Цилиндр)', 'Черный' ),  
 ('(Цилиндр)', 'Белый' ),  
 ('(Цилиндр)', 'Серый' ),  
 ('(Ботинки)', 'Черный' ),  
 ('(Ботинки)', 'Белый' ),  
 ('(Ботинки)', 'Серый' );  
  
  
insert into props (*name*, *color*)  
 values  
 ('(Жезл)', 'Золотой' ),  
 ('(Жезл)', 'Серебрянный' ),  
 ('(Посох)', 'Золотой' ),  
 ('(Посох)', 'Серебрянный' ),  
 ('(Трон)', 'Золотой' ),  
 ('(Трон)', 'Серебрянный' ),  
 ('(Туфли)', 'Хрусталь' );  
  
  
insert into clothes\_author (*id\_clothes*, *id\_author*)  
 values  
 ((select *id\_clothes* from clothes where *name* like 'Шут'),  
 (select *id\_author* from author where (*author\_fio*).first\_name like 'Имя 1')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Маленький король'),  
 (select *id\_author* from author where (*author\_fio*).first\_name like 'Имя 2')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Платье Золушки'),  
 (select *id\_author* from author where (*author\_fio*).first\_name like 'Имя 2')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Костюм Мастера'),  
 (select *id\_author* from author where (*author\_fio*).first\_name like 'Имя 2')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Костюм Мастера'),  
 (select *id\_author* from author where (*author\_fio*).first\_name like 'Имя 3')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Костюм Маргариты'),  
 (select *id\_author* from author where (*author\_fio*).first\_name like 'Имя 2'));  
  
  
insert into clothes\_detail (*id\_clothes*, *id\_detail*)  
values  
 ((select *id\_clothes* from clothes where *name* like 'Шут'),  
 (select *id\_detail* from detail where *name* = *cast*('Куртка' as varchar)  
 and *color* = 'Черный')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Шут'),  
 (select *id\_detail* from detail where *name* = *cast*('Штаны' as varchar)  
 and *color* = 'Черный')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Шут'),  
 (select *id\_detail* from detail where *name* = *cast*('Шляпа' as varchar)  
 and *color* = 'Черный')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Шут'),  
 (select *id\_detail* from detail where *name* = *cast*('Брюки' as varchar)  
 and *color* = 'Черный')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Шут'),  
 (select *id\_detail* from detail where *name* = *cast*('Цилиндр' as varchar)  
 and *color* = 'Черный')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Шут'),  
 (select *id\_detail* from detail where *name* = *cast*('Ботинки' as varchar)  
 and *color* = 'Черный')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Маленький король'),  
 (select *id\_detail* from detail where *name* = *cast*('Куртка' as varchar)  
 and *color* = 'Белый')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Маленький король'),  
 (select *id\_detail* from detail where *name* = *cast*('Шляпа' as varchar)  
 and *color* = 'Белый')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Маленький король'),  
 (select *id\_detail* from detail where *name* = *cast*('Ботинки' as varchar)  
 and *color* = 'Белый')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Маленький король'),  
 (select *id\_detail* from detail where *name* = *cast*('Штаны' as varchar)  
 and *color* = 'Белый')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Платье Золушки'),  
 (select *id\_detail* from detail where *name* = *cast*('Куртка' as varchar)  
 and *color* = 'Серый')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Костюм Мастера'),  
 (select *id\_detail* from detail where *name* = *cast*('Штаны' as varchar)  
 and *color* = 'Черный')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Костюм Маргариты'),  
 (select *id\_detail* from detail where *name* = *cast*('Штаны' as varchar)  
 and *color* = 'Серый'));  
  
  
*-- Добавить props для role*insert into role\_props (*id\_role*, *id\_props*)  
values  
 ((select *id\_role* from role where *name* like 'Золушка'),  
 (select *id\_detail* from props p where *name* = *cast*('Туфли' as varchar)  
 and *color* = 'Хрусталь')),  
  
 ((select *id\_role* from role where *name* like 'Мастер'),  
 (select *id\_detail* from props p where *name* = *cast*('Жезл' as varchar)  
 and *color* = 'Золотой')),  
  
 ((select *id\_role* from role where *name* like 'Роль шута'),  
 (select *id\_detail* from props p where *name* = *cast*('Жезл' as varchar)  
 and *color* = 'Серебрянный'));  
  
  
*-- Добавить props для clothes*insert into clothes\_props (*id\_clothes*, *id\_props*)  
values  
 ((select *id\_clothes* from clothes where *name* like 'Платье Золушки'),  
 (select *id\_detail* from props where *name* = *cast*('Туфли' as varchar)  
 and *color* = 'Хрусталь')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Костюм Мастера'),  
 (select *id\_detail* from props where *name* = *cast*('Жезл' as varchar)  
 and *color* = 'Золотой')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Шут'),  
 (select *id\_detail* from props where *name* = *cast*('Жезл' as varchar)  
 and *color* = 'Серебрянный'));  
  
  
insert into clothes\_role (*id\_clothes*, *id\_role*)  
 values ((select *id\_clothes* from clothes where *name* like 'Шут'),  
 (select *id\_role* from role where *name* like 'Роль шута')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Маленький король'),  
 (select *id\_role* from role where *name* like 'Роль короля')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Платье Золушки'),  
 (select *id\_role* from role where *name* like 'Золушка')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Костюм Мастера'),  
 (select *id\_role* from role where *name* like 'Мастер')),  
  
 ((select *id\_clothes* from clothes where *name* like 'Костюм Маргариты'),  
 (select *id\_role* from role where *name* like 'Маргарита'));

#### Запрос выборки с условием к предку

*-- К предку*select d.id\_detail, name, color, p.relname  
from only detail d, pg\_class p  
where d.*TABLEOID* = p.oid;

#### Запрос выборки с условием к потомку

*-- К потомку*select pr.id\_detail, name, color, id\_detail, p.relname  
from props pr, pg\_class p  
where pr.*TABLEOID* = p.oid;

#### Запрос выборки с условием к предку и потомку

*-- К предку и потомку*select d.id\_detail, name, color, p.relname  
from detail d, pg\_class p  
where d.*TABLEOID* = p.oid;

#### Код операторов

create or replace function  
 *is\_equal\_name\_detail\_with\_varchar*(td name\_detail, str varchar)  
 returns boolean as  
 $$ select *cast*($1 as varchar(100))  
 = *concat*('(', str, ')'); $$ language SQL;  
  
create or replace function  
 *is\_equal\_name\_details*(td1 name\_detail, td2 name\_detail)  
 returns boolean as  
 $$ select *cast*($1 as varchar(100))  
 = *cast*($2 as varchar(100)); $$ language SQL;  
  
create operator = (  
 leftarg = name\_detail,  
 rightarg = varchar,  
 procedure = *is\_equal\_name\_detail\_with\_varchar*);  
  
create operator = (  
 leftarg = name\_detail,  
 rightarg = name\_detail,  
 procedure = *is\_equal\_name\_details*,  
 commutator = =  
);

#### Пример выполнения операторов

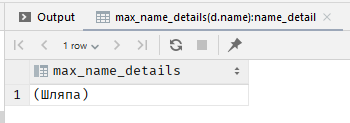
select *is\_equal\_name\_detail\_with\_varchar*('(test)','test');  
select not *is\_equal\_name\_detail\_with\_varchar*('(test)','(test)');  
select *is\_equal\_name\_details*('(test)', '(test)');  
  
select *cast*('(test)' as name\_detail)  
 =  
 *cast*('test' as varchar);  
  
select *cast*('(test)' as name\_detail)  
 =  
 *cast*('(test)' as name\_detail);

#### Пользовательская агрегатная функция

create or replace function  
 *count\_name\_details*(name\_detail)  
 returns int language sql as  
 $$  
 select *count*(d.*name*)  
 from detail d  
 where d.*name* = $1  
 group by d.*name*;  
 $$;  
  
create or replace function  
 *is\_left\_larger*(td1 name\_detail, td2 name\_detail)  
 returns boolean language sql as  
 $$  
 select *count\_name\_details*($1) > *count\_name\_details*($2);  
 $$;  
  
  
create operator > (  
 leftarg = name\_detail,  
 rightarg = name\_detail,  
 procedure = *is\_left\_larger*,  
 commutator = >  
);  
  
  
create or replace function *max\_name\_detail*(nd1 name\_detail, nd2 name\_detail)  
 returns name\_detail language sql as  
 $$  
 select case  
 when nd1 > nd2  
 then nd1  
 else nd2  
 end;  
 $$;  
  
  
create aggregate *max\_name\_details* (name\_detail) (  
 sfunc = *max\_name\_detail*,  
 stype = name\_detail  
);

#### Пример выполнения пользовательской агрегатной функции

select *max\_name\_details*(d.*name*) from detail d;



Вспомогательные запросы

select *\** from detail;  
  
select d.*name*, *count*(d.*name*) as c  
 from detail d  
 group by d.*name* order by c desc;