ГУАП

КАФЕДРА № 43

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

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

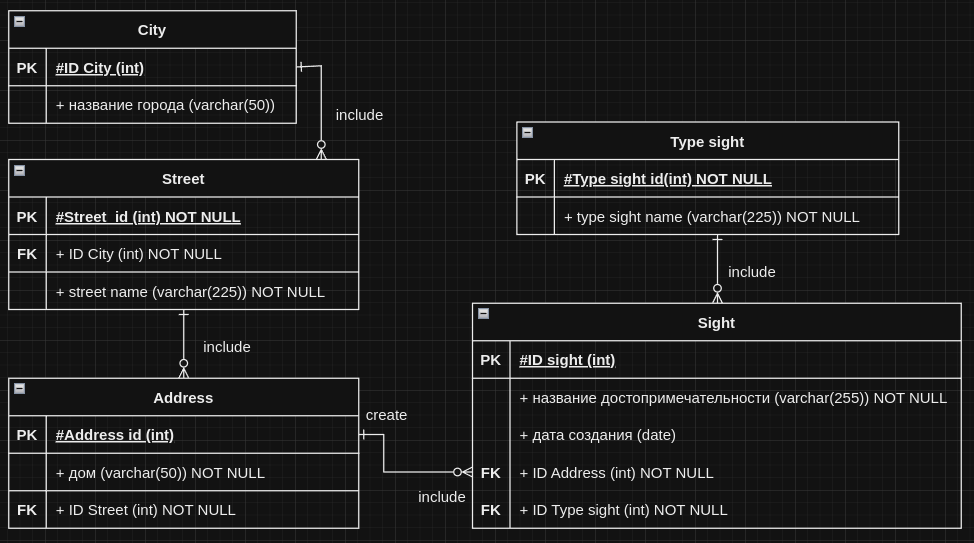
|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №7  Хранимые процедуры |
| **по дисциплине: Проектирование баз данных** |
|  |
|  |

РАБОТУ ВЫПОЛНИЛ

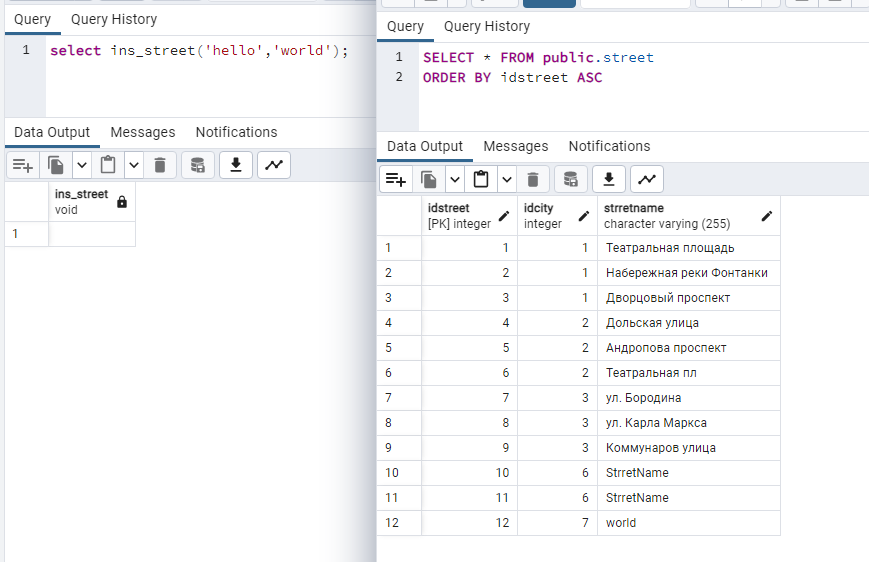
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ ГР. | 4134к |  |  |  | Столяров Н.С. |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург

2023

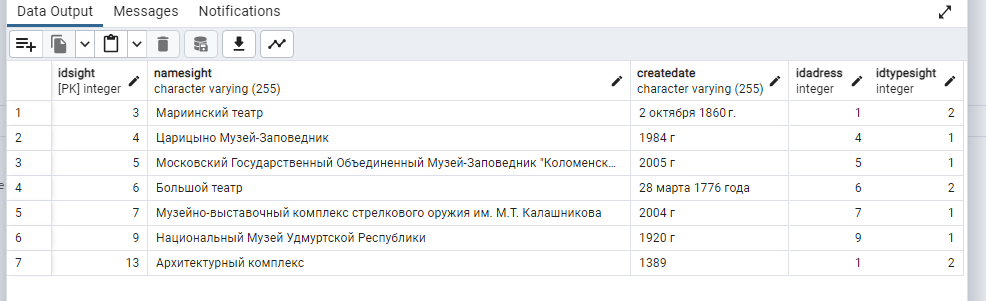
  
  
**вставка улицы по городу если нет города вставка города**

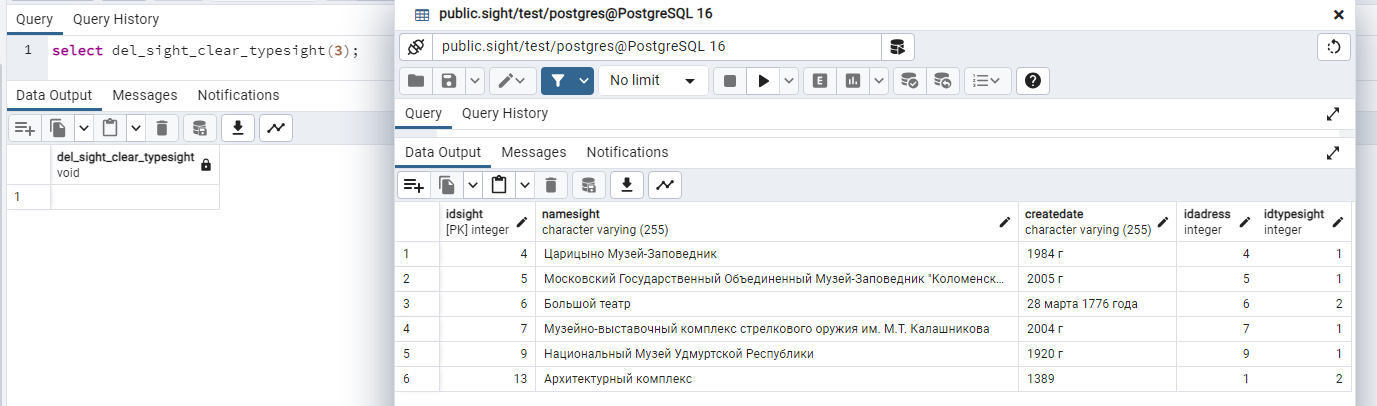
|  |
| --- |
| -- вставка улицы по городу если нет города вставка города  CREATE OR REPLACE FUNCTION ins\_street (\_NameCity varchar(50),\_StrretName varchar(225))  RETURNS VOID AS $$  declare  new\_id\_city int;  new\_id\_street int;  BEGIN  if not exists(select \* from city where NameCity = \_NameCity) then  begin  select COALESCE(max(IDCity) + 1,0) into new\_id\_city from city;  insert into city (IDCity,NameCity) values (new\_id\_city,\_NameCity);  end;  else  select IDCity into new\_id\_city from city where NameCity = \_NameCity;  end if;  select COALESCE(max(IDStreet) + 1,0) into new\_id\_street from street;  insert into street (IDStreet,IDCity,StrretName) values (new\_id\_street,new\_id\_city,\_StrretName);  END;  $$ LANGUAGE plpgsql;  DROP FUNCTION ins\_street;  select ins\_street('123','StrretName'); |



**удаление sight и type sight если на него больше не чего не указывает**

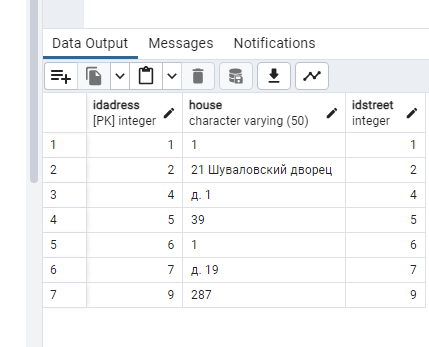
|  |
| --- |
| CREATE OR REPLACE FUNCTION del\_sight\_clear\_typesight(IDSight\_del int)  RETURNS VOID AS $$  declare  IDTypeSight\_del int;  begin  select IDTypeSight into IDTypeSight\_del from Sight where IDSight = IDSight\_del;  delete from Sight where IDSight = IDSight\_del;  if not exists(select \* from Sight where IDTypeSight = IDTypeSight\_del)  then delete from TypeSight where IDTypeSight = IDTypeSight\_del;  end if;  END;  $$ LANGUAGE plpgsql;  select del\_sight\_clear\_typesight(8);  select del\_sight\_clear\_typesight(11); |

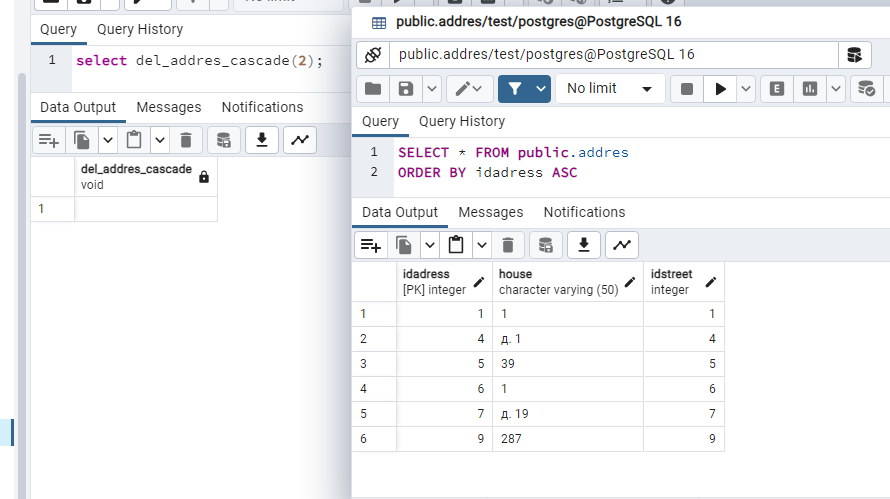




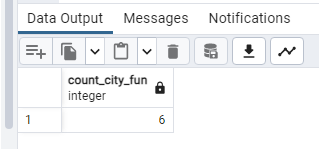
**удаление адреса и всех sight которые на него указывют (каскадное удаление)**

|  |
| --- |
| CREATE OR REPLACE FUNCTION del\_addres\_cascade(IDAdress\_del int)  RETURNS VOID AS $$  begin  delete from Sight where IDAdress = IDAdress\_del;  delete from addres where IDAdress = IDAdress\_del;  end;  $$ LANGUAGE plpgsql;  select del\_addres\_cascade(8); |



**возврат количества городов**

|  |
| --- |
| -- возврат количество городов  CREATE OR REPLACE FUNCTION count\_city\_fun()  RETURNS INT AS $$  declare  val int default 0;  begin  select COALESCE(count(IDCity),0) into val from city;  return val;  end;  $$ LANGUAGE plpgsql; |



**формирование статистики во временной таблице**

|  |
| --- |
| -- формирование статистики во временной таблице  CREATE OR REPLACE FUNCTION city\_statistics ()  RETURNS TABLE (  -- id\_stat int,  IDCity int,  count\_street int,  count\_addres int  ) as $$  begin  create temporary table if not exists city\_stat (  -- id\_stat serial primary key,  IDCity int,  count\_street int,  count\_addres int  );  insert into city\_stat(IDCity,count\_addres,count\_street)  select City.IDCity, count(IDAdress) as count\_addres,count(street.IDStreet) as count\_street from City  left join street on street.IDCity = City.IDCity  left join addres on addres.IDStreet = street.IDStreet  group by City.IDCity;  RETURN QUERY SELECT \* FROM city\_stat;  drop table city\_stat;  end;  $$ LANGUAGE plpgsql;  DROP FUNCTION city\_statistics;  select \* from city\_statistics(); |

