ЭТАПЫ ВЫПОЛНЕНИЯ

**Этапы выполнения работы:**

1. Файлы скриптов:
   1. Файл создания хранимых процедур и распределения прав доступа;



* 1. Файл заполнение структуры данными;



* 1. Файл манипулирования над данными;



* 1. Файл создания пересоздания структуры БД;



* 1. Файл удаления структуры.



1. Приложение 1, создание хранимых процедур и распределение прав доступа;
2. Приложение 2, заполнение структуры базы данных;
3. Приложение 3, выборка данных и манипулирование данными;
4. Создание структуры базы данных;

Таблица 1 – Хронология создания структуры БД

| № ПП | Файл | Скрипт |
| --- | --- | --- |
|  | 5 КТ Create.sql | create or replace procedure Structure\_Create ()  language plpgsql  as $$  begin  execute procedure\_sctipt;  create table if not exists Department  (  ID\_Department serial not null constraint PK\_Department primary key,  Name\_Department Varchar(100) not null  );  create index if not exists index\_ID\_Department on Department (ID\_Department);  create index if not exists index\_Name\_Department on Department (Name\_Department);  create table if not exists Post  (  ID\_Post Serial not null constraint PK\_Post primary key,  Name\_Post Varchar(50) not null  );  create index if not exists index\_ID\_Post on Post (ID\_Post);  create index if not exists index\_Name\_Post on Post (Name\_Post);  create table if not exists Discipline  (  ID\_Discipline Serial not null constraint PK\_Discipline primary key,  Prefix\_Discipline Varchar(10) not null,  Name\_Discipline Varchar(100) not null  );  create index if not exists index\_ID\_Discipline on Discipline (ID\_Discipline);  create index if not exists index\_Prefix\_Name\_Discipline on Discipline (Prefix\_Discipline, Name\_Discipline);  create table if not exists User\_Profile  (  UP\_Login Varchar(36) not null constraint PK\_User\_Profile primary key,  UP\_Password Varchar(36) not null,  U\_Surname Varchar(50) not null,  U\_Name Varchar(50) not null,  U\_Patronymic Varchar(50) null  );  create index if not exists index\_UP\_Login\_Password on User\_Profile (UP\_Login, UP\_Password);  create index if not exists index\_U\_Surname\_Name\_Patronymic on User\_Profile (U\_Surname, U\_Name, U\_Patronymic);  create table if not exists Territory  (  ID\_Territory Serial not null constraint PK\_Territory primary key,  Name\_Territory Varchar(50) not null  );  create index if not exists index\_ID\_Territory on Territory (ID\_Territory);  create index if not exists index\_Name\_Territory on Territory (Name\_Territory);  create table if not exists Audience  (  ID\_Audience Serial not null constraint PK\_Audience primary key,  Number\_Audience Varchar(7) not null  );  create index if not exists index\_ID\_Audience on Audience (ID\_Audience);  create index if not exists index\_Number\_Audience on Audience (Number\_Audience);  create table if not exists Bussines\_Role  (  ID\_Bussines\_Role Serial not null constraint PK\_Bussines\_Role primary key,  Name\_BR Varchar(50) not null,  Department\_ID Int not null references Department (ID\_Department)  );  create index if not exists index\_ID\_Bussines\_Role on Bussines\_Role (ID\_Bussines\_Role);  create index if not exists index\_Name\_BR on Bussines\_Role (Name\_BR);  create table if not exists WrkCrr  (  ID\_WrkCrr Serial not null constraint PK\_WrkCrr primary key,  N\_Week\_WrkCrr Int not null,  N\_Hours\_WrkCrr Decimal(5,2) not null,  Bussines\_Role\_ID Int not null references Bussines\_Role (ID\_Bussines\_Role)  );  create index if not exists index\_ID\_WrkCrr on WrkCrr (ID\_WrkCrr);  create table if not exists Study\_Grpoup  (  ID\_Study\_Grpoup Serial not null constraint PK\_Study\_Grpoup primary key,  Name\_St\_Grp Varchar(10) not null,  Date\_Create\_SG Date not null,  WrkCrr\_ID Int not null references WrkCrr (ID\_WrkCrr)  );  create index if not exists index\_ID\_Study\_Grpoup on Study\_Grpoup (ID\_Study\_Grpoup);  create index if not exists index\_Name\_St\_Grp on Study\_Grpoup (Name\_St\_Grp);  create index if not exists index\_Date\_Create\_SG on Study\_Grpoup (Date\_Create\_SG);  create table if not exists Student  (  Login\_Student Varchar(36) not null constraint PK\_Student primary key  references User\_Profile (UP\_Login) on update cascade on delete cascade,  Credit\_Card\_N Varchar(13) not null  );  create index if not exists index\_Login\_Student on Student (Login\_Student);  create index if not exists index\_Credit\_Card\_N on Student (Credit\_Card\_N);  create table if not exists Employee  (  Employee\_Login Varchar(36) not null constraint PK\_Employee primary key  references User\_Profile (UP\_Login) on update cascade on delete cascade,  Personal\_File\_N Varchar(13) not null  );  create index if not exists index\_Employee\_Login on Employee (Employee\_Login);  create index if not exists index\_Personal\_File\_N on Employee (Personal\_File\_N);  create table if not exists Dep\_Discipl  (  ID\_Dep\_Discipl Serial not null constraint PK\_Dep\_Discipl primary key,  Course\_N Int not null,  Semester\_N Int not null,  N\_H\_Dep\_Discipl Int not null,  WrkCrr\_ID Int not null references WrkCrr (ID\_WrkCrr),  Discipline\_ID Int not null references Discipline (ID\_Discipline)  );  create index if not exists index\_ID\_Dep\_Discipl on Dep\_Discipl(ID\_Dep\_Discipl);  create index if not exists index\_Course\_N on Dep\_Discipl (Course\_N);  create index if not exists index\_Semester\_N on Dep\_Discipl (Semester\_N);  create table if not exists Distrib\_Grps  (  ID\_Distrib\_Grps Serial not null constraint PK\_Distrib\_Grps primary key,  Study\_Grpoup\_ID Int not null references Study\_Grpoup (ID\_Study\_Grpoup),  Student\_Login Varchar(36) not null references Student (Login\_Student) on update cascade on delete cascade  );  create index if not exists index\_ID\_Distrib\_Grps on Distrib\_Grps (ID\_Distrib\_Grps);  create table if not exists Audiens\_Territ  (  ID\_Audiens\_Territ Serial not null constraint PK\_Audiens\_Territ primary key,  Territory\_ID Int not null references Territory (ID\_Territory),  Audience\_ID Int not null references Audience (ID\_Audience)  );  create index if not exists index\_ID\_Audiens\_Territ on Audiens\_Territ (ID\_Audiens\_Territ);  create table if not exists Combination  (  ID\_Combination Serial not null constraint PK\_Combination primary key,  Department\_ID Int not null references Department (ID\_Department),  Post\_ID Int not null references Post (ID\_Post),  Login\_Employee Varchar(36) not null references Employee (Employee\_Login) on update cascade on delete cascade  );  create index if not exists index\_ID\_Combination on Combination (ID\_Combination);  create table if not exists Workload  (  ID\_Workload Serial not null constraint PK\_Workload primary key,  N\_Hr\_Workload Decimal(5,1) not null,  Study\_Grpoup\_ID Int references Study\_Grpoup (ID\_Study\_Grpoup),  Dep\_Discipl\_ID Int references Dep\_Discipl (ID\_Dep\_Discipl),  Combination\_ID Int references Combination (ID\_Combination)  );  create index if not exists index\_ID\_Workload on Workload (ID\_Workload);  create table if not exists Schedule  (  ID\_Schedule Serial not null constraint PK\_Schedule primary key,  D\_W\_Schedule Varchar(11) not null,  Pair\_N\_Schedule Int not null,  N\_S\_H\_Schedule Int not null,  Workload\_ID Int not null references Workload (ID\_Workload),  Audiens\_Territ\_ID Int not null references Audiens\_Territ (ID\_Audiens\_Territ)  );  create index if not exists index\_ID\_Schedule on Schedule (ID\_Schedule);  create index if not exists index\_D\_W\_Schedule on Schedule (D\_W\_Schedule);  create index if not exists index\_Pair\_N\_Schedule on Schedule (Pair\_N\_Schedule);  grant select, insert, update on Department to rl\_architect;  grant usage, select on sequence department\_id\_department\_seq to rl\_architect;  grant select, insert, update on Bussines\_Role to rl\_architect;  grant usage, select on sequence bussines\_role\_id\_bussines\_role\_seq to rl\_architect;  grant select on Post to rl\_architect;  grant select on Study\_Grpoup to rl\_architect;  grant select on Student to rl\_architect;  grant select on Distrib\_Grps to rl\_architect;  grant select on Audiens\_Territ to rl\_architect;  grant select on Combination to rl\_architect;  grant select on Schedule to rl\_architect;  grant select, insert, update, delete on Discipline to rl\_architect;  grant usage, select on sequence discipline\_id\_discipline\_seq to rl\_architect;  grant select, insert, update, delete on WrkCrr to rl\_architect;  grant usage, select on sequence wrkcrr\_id\_wrkcrr\_seq to rl\_architect;  grant select, insert, update, delete on Dep\_Discipl to rl\_architect;  grant usage, select on sequence dep\_discipl\_id\_dep\_discipl\_seq to rl\_architect;  grant select, insert, update, delete on Workload to rl\_architect;  grant usage, select on sequence workload\_id\_workload\_seq to rl\_architect;  grant select, update on User\_Profile to rl\_architect;  grant select, update on User\_Profile to rl\_teacher;  grant select on Student to rl\_teacher;  grant select on Schedule to rl\_teacher;  grant select, update on User\_Profile to rl\_student;  grant select, update on Student to rl\_student;  grant select on Schedule to rl\_student;  grant select, update on User\_Profile to rl\_managersd;  grant select, insert, update, delete on Study\_Grpoup to rl\_managersd;  grant usage, select on sequence study\_grpoup\_id\_study\_grpoup\_seq to rl\_managersd;  grant select, insert, update, delete on Student to rl\_managersd;  grant select, insert, update, delete on Distrib\_Grps to rl\_managersd;  grant usage, select on sequence distrib\_grps\_id\_distrib\_grps\_seq to rl\_managersd;  grant select, insert, update, delete on Schedule to rl\_managersd;  grant usage, select on sequence schedule\_id\_schedule\_seq to rl\_managersd;  grant select, insert, update, delete on Department to rl\_administrator;  grant usage, select on sequence department\_id\_department\_seq to rl\_administrator;  grant select, insert, update, delete on Post to rl\_administrator;  grant usage, select on sequence post\_id\_post\_seq to rl\_administrator;  grant select, insert, update, delete on Territory to rl\_administrator;  grant usage, select on sequence territory\_id\_territory\_seq to rl\_administrator;  grant select, insert, update, delete on Audience to rl\_administrator;  grant usage, select on sequence audience\_id\_audience\_seq to rl\_administrator;  grant select, insert, update, delete on Bussines\_Role to rl\_administrator;  grant usage, select on sequence bussines\_role\_id\_bussines\_role\_seq to rl\_administrator;  grant select, insert, update, delete on Student to rl\_administrator;  grant select, insert, update, delete on Employee to rl\_administrator;  grant select, insert, update, delete on Audiens\_Territ to rl\_administrator;  grant usage, select on sequence audiens\_territ\_id\_audiens\_territ\_seq to rl\_administrator;  grant select, insert, update, delete on Combination to rl\_administrator;  grant usage, select on sequence combination\_id\_combination\_seq to rl\_administrator;  grant select, insert, update, delete on Schedule to rl\_administrator;  grant usage, select on sequence schedule\_id\_schedule\_seq to rl\_administrator;  grant select, insert, delete on User\_Profile to rl\_administrator;  end;  $$; |
|  | 5.1 КТ Proceudre.sql | create or replace procedure Department\_Insert (p\_name\_department Varchar(100))  language plpgsql  as $$  begin  insert into Department (name\_department)  values (p\_name\_department);  end;  $$;  create or replace procedure Department\_Update (p\_id\_department int, p\_name\_department Varchar(100))  language plpgsql  as $$  begin  update Department set  name\_department = p\_name\_department  where  id\_department = p\_id\_department;  end;  $$;  create or replace procedure Department\_Delete (p\_id\_department int)  language plpgsql  as $$  begin  delete from Department  where  id\_department = p\_id\_department;  end;  $$;  create or replace procedure Post\_Insert (p\_Name\_Post varchar (50))  language plpgsql  as $$  begin  insert into Post (Name\_Post)  values (p\_Name\_Post);  end;  $$;  create or replace procedure Post\_Update (p\_ID\_Post int, p\_Name\_Post varchar (50))  language plpgsql  as $$  begin  update Post set  Name\_Post = p\_Name\_Post  where  ID\_Post = p\_ID\_Post;  end;  $$;  create or replace procedure Post\_Delete (p\_ID\_Post int)  language plpgsql  as $$  begin  delete from Post  where  ID\_Post = p\_ID\_Post;  end;  $$;  create or replace procedure Discipline\_Insert (p\_Prefix\_Discipline varchar (10), p\_Name\_Discipline varchar (100))  language plpgsql  as $$  begin  insert into Discipline (Prefix\_Discipline, Name\_Discipline)  values (p\_Prefix\_Discipline, p\_Name\_Discipline);  end;  $$;  create or replace procedure Discipline\_Update (p\_ID\_Discipline int, p\_Prefix\_Discipline varchar (10), p\_Name\_Discipline varchar (100))  language plpgsql  as $$  begin  update Discipline set  Prefix\_Discipline = p\_Prefix\_Discipline,  Name\_Discipline = p\_Name\_Discipline  where  ID\_Discipline = p\_ID\_Discipline;  end;  $$;  create or replace procedure Discipline\_Delete (p\_ID\_Discipline int)  language plpgsql  as $$  begin  delete from Discipline  where  ID\_Discipline = p\_ID\_Discipline;  end;  $$;  create or replace procedure User\_Profile\_Insert (p\_UP\_Login Varchar(36), p\_UP\_Password Varchar(36), p\_U\_Surname Varchar(50),  p\_U\_Name Varchar(50), p\_U\_Patronymic Varchar(50))  language plpgsql  as $$  begin  insert into User\_Profile (UP\_Login, UP\_Password, U\_Surname, U\_Name, U\_Patronymic)  values (p\_UP\_Login, p\_UP\_Password, p\_U\_Surname, p\_U\_Name, p\_U\_Patronymic);  end;  $$;  create or replace procedure User\_Profile\_Update (p\_UP\_Login\_Old Varchar(36), p\_UP\_Login\_New Varchar(36), p\_UP\_Password Varchar(36),  p\_U\_Surname Varchar(50), p\_U\_Name Varchar(50), p\_U\_Patronymic Varchar(50))  language plpgsql  as $$  begin  update User\_Profile set  UP\_Login = p\_UP\_Login\_New,  UP\_Password = p\_UP\_Password,  U\_Surname = p\_U\_Surname,  U\_Name = p\_U\_Name,  U\_Patronymic = p\_U\_Patronymic  where  UP\_Login = p\_UP\_Login\_Old;  end;  $$;  create or replace procedure User\_Profile\_Delete (p\_UP\_Login Varchar(36))  language plpgsql  as $$  begin  delete from User\_Profile  where  UP\_Login = p\_UP\_Login;  end;  $$;  create or replace procedure Territory\_Insert (p\_Name\_Territory varchar (50))  language plpgsql  as $$  begin  insert into Territory (Name\_Territory)  values (p\_Name\_Territory);  end;  $$;  create or replace procedure Territory\_Update (p\_ID\_Territory int, p\_Name\_Territory varchar (50))  language plpgsql  as $$  begin  update Territory set  Name\_Territory = p\_Name\_Territory  where  ID\_Territory = p\_ID\_Territory;  end;  $$;  create or replace procedure Territory\_Delete (p\_ID\_Territory int)  language plpgsql  as $$  begin  delete from Territory  where  ID\_Territory = p\_ID\_Territory;  end;  $$;  create or replace procedure Territory\_Delete (p\_ID\_Territory int)  language plpgsql  as $$  begin  delete from Territory  where  ID\_Territory = p\_ID\_Territory;  end;  $$;  create or replace procedure Audience\_Insert (p\_Number\_Audience varchar (7))  language plpgsql  as $$  begin  insert into Audience (Number\_Audience)  values (p\_Number\_Audience);  end;  $$;  create or replace procedure Audience\_Update (p\_ID\_Audience int, p\_Number\_Audience varchar (7))  language plpgsql  as $$  begin  update Audience set  Number\_Audience = p\_Number\_Audience  where  ID\_Audience = p\_ID\_Audience;  end;  $$;  create or replace procedure Audience\_Delete (p\_ID\_Audience int)  language plpgsql  as $$  begin  delete from Audience  where  ID\_Audience = p\_ID\_Audience;  end;  $$;  create or replace procedure Bussines\_Role\_Insert (p\_Name\_BR varchar (50), p\_Department\_ID int)  language plpgsql  as $$  begin  insert into Bussines\_Role (Name\_BR, Department\_ID)  values (p\_Name\_BR, p\_Department\_ID);  end;  $$;  create or replace procedure Bussines\_Role\_Update (p\_ID\_Bussines\_Role int, p\_Name\_BR varchar (50), p\_Department\_ID int)  language plpgsql  as $$  begin  update Bussines\_Role set  Name\_BR = p\_Name\_BR,  Department\_ID = p\_Department\_ID  where  ID\_Bussines\_Role = p\_ID\_Bussines\_Role;  end;  $$;  create or replace procedure Bussines\_Role\_Delete (p\_ID\_Bussines\_Role int)  language plpgsql  as $$  begin  delete from Bussines\_Role  where  ID\_Bussines\_Role = p\_ID\_Bussines\_Role;  end;  $$;  create or replace procedure WrkCrr\_Insert (p\_N\_Week\_WrkCrr int, p\_N\_Hours\_WrkCrr decimal(5,2), p\_Bussines\_Role\_ID int)  language plpgsql  as $$  begin  insert into WrkCrr (N\_Week\_WrkCrr, N\_Hours\_WrkCrr, Bussines\_Role\_ID)  values (p\_N\_Week\_WrkCrr, p\_N\_Hours\_WrkCrr, p\_Bussines\_Role\_ID);  end;  $$;  create or replace procedure WrkCrr\_Update (p\_ID\_WrkCrr int, p\_N\_Week\_WrkCrr int, p\_N\_Hours\_WrkCrr decimal(5,2), p\_Bussines\_Role\_ID int)  language plpgsql  as $$  begin  update WrkCrr set  N\_Week\_WrkCrr = p\_N\_Week\_WrkCrr,  N\_Hours\_WrkCrr = p\_N\_Hours\_WrkCrr,  Bussines\_Role\_ID = p\_Bussines\_Role\_ID  where  ID\_WrkCrr = p\_ID\_WrkCrr;  end;  $$;  create or replace procedure WrkCrr\_Delete (p\_ID\_WrkCrr int)  language plpgsql  as $$  begin  delete from WrkCrr  where  ID\_WrkCrr = p\_ID\_WrkCrr;  end;  $$;  create or replace procedure Study\_Grpoup\_Insert (p\_Name\_St\_Grp varchar (10), p\_Date\_Create\_SG date, p\_WrkCrr\_ID int)  language plpgsql  as $$  begin  insert into Study\_Grpoup (Name\_St\_Grp, Date\_Create\_SG, WrkCrr\_ID)  values (p\_Name\_St\_Grp, p\_Date\_Create\_SG, p\_WrkCrr\_ID);  end;  $$;  create or replace procedure Study\_Grpoup\_Update (p\_ID\_Study\_Grpoup int, p\_Name\_St\_Grp varchar (10), p\_Date\_Create\_SG date, p\_WrkCrr\_ID int)  language plpgsql  as $$  begin  update Study\_Grpoup set  Name\_St\_Grp = p\_Name\_St\_Grp,  Date\_Create\_SG = p\_Date\_Create\_SG,  WrkCrr\_ID = p\_WrkCrr\_ID  where  ID\_Study\_Grpoup = p\_ID\_Study\_Grpoup;  end;  $$;  create or replace procedure Study\_Grpoup\_Delete (p\_ID\_Study\_Grpoup int)  language plpgsql  as $$  begin  delete from Study\_Grpoup  where  ID\_Study\_Grpoup = p\_ID\_Study\_Grpoup;  end;  $$;  create or replace procedure Student\_Insert (p\_Login\_Student varchar (36), p\_Credit\_Card\_N varchar (13))  language plpgsql  as $$  begin  insert into Student (Login\_Student, Credit\_Card\_N)  values (p\_Login\_Student, p\_Credit\_Card\_N);  end;  $$;  create or replace procedure Student\_Update (p\_Login\_Student\_Old varchar (36), p\_Login\_Student\_New varchar (36), p\_Credit\_Card\_N varchar (13))  language plpgsql  as $$  begin  update Student set  Login\_Student = p\_Login\_Student\_New,  Credit\_Card\_N = p\_Credit\_Card\_N  where  Login\_Student = p\_Login\_Student\_Old;  end;  $$;  create or replace procedure Student\_Delete (p\_Login\_Student varchar (36))  language plpgsql  as $$  begin  delete from Student  where  Login\_Student = p\_Login\_Student;  end;  $$;  create or replace procedure Employee\_Insert (p\_Employee\_Login Varchar(36), p\_Personal\_File\_N Varchar(13))  language plpgsql  as $$  begin  insert into Employee (Employee\_Login, Personal\_File\_N)  values (p\_Employee\_Login, p\_Personal\_File\_N);  end;  $$;  create or replace procedure Employee\_Update (p\_Employee\_Login\_Old Varchar(36), p\_Employee\_Login\_New Varchar(36), p\_Personal\_File\_N Varchar(13))  language plpgsql  as $$  begin  update Employee set  Employee\_Login = p\_Employee\_Login\_New,  Personal\_File\_N = p\_Personal\_File\_N  where  Employee\_Login = p\_Employee\_Login\_Old;  end;  $$;  create or replace procedure Employee\_Delete (p\_Employee\_Login Varchar(36))  language plpgsql  as $$  begin  delete from Employee  where  Employee\_Login = p\_Employee\_Login;  end;  $$;  create or replace procedure Dep\_Discipl\_Insert (p\_Course\_N int, p\_Semester\_N int,  p\_N\_H\_Dep\_Discipl int, p\_WrkCrr\_ID int, p\_Discipline\_ID int)  language plpgsql  as $$  begin  insert into Dep\_Discipl (Course\_N, Semester\_N, N\_H\_Dep\_Discipl, WrkCrr\_ID, Discipline\_ID)  values (p\_Course\_N, p\_Semester\_N, p\_N\_H\_Dep\_Discipl, p\_WrkCrr\_ID, p\_Discipline\_ID);  end;  $$;  create or replace procedure Dep\_Discipl\_Update (p\_ID\_Dep\_Discipl int, p\_Course\_N int, p\_Semester\_N int,  p\_N\_H\_Dep\_Discipl int, p\_WrkCrr\_ID int, p\_Discipline\_ID int)  language plpgsql  as $$  begin  update Dep\_Discipl set  Course\_N = p\_Course\_N,  Semester\_N = p\_Semester\_N,  N\_H\_Dep\_Discipl = p\_N\_H\_Dep\_Discipl,  WrkCrr\_ID = p\_WrkCrr\_ID,  Discipline\_ID = p\_Discipline\_ID  where  ID\_Dep\_Discipl = p\_ID\_Dep\_Discipl;  end;  $$;  create or replace procedure Dep\_Discipl\_Delete (p\_ID\_Dep\_Discipl int)  language plpgsql  as $$  begin  delete from Dep\_Discipl  where  ID\_Dep\_Discipl = p\_ID\_Dep\_Discipl;  end;  $$;  create or replace procedure Distrib\_Grps\_Insert (p\_Student\_Login varchar (36), p\_Study\_Grpoup\_ID int)  language plpgsql  as $$  begin  insert into Distrib\_Grps (Student\_Login, Study\_Grpoup\_ID)  values (p\_Student\_Login, p\_Study\_Grpoup\_ID);  end;  $$;  create or replace procedure Distrib\_Grps\_Update (p\_ID\_Distrib\_Grps int, p\_Student\_Login varchar (36), p\_Study\_Grpoup\_ID int)  language plpgsql  as $$  begin  update Distrib\_Grps set  Student\_Login = p\_Student\_Login,  Study\_Grpoup\_ID = p\_Study\_Grpoup\_ID  where  ID\_Distrib\_Grps = p\_ID\_Distrib\_Grps;  end;  $$;  create or replace procedure Distrib\_Grps\_Delete (p\_ID\_Distrib\_Grps int)  language plpgsql  as $$  begin  delete from Distrib\_Grps  where  ID\_Distrib\_Grps = p\_ID\_Distrib\_Grps;  end;  $$;  create or replace procedure Audiens\_Territ\_Insert (p\_Territory\_ID int, p\_Audience\_ID int)  language plpgsql  as $$  begin  insert into Audiens\_Territ (Territory\_ID, Audience\_ID)  values (p\_Territory\_ID, p\_Audience\_ID);  end;  $$;  create or replace procedure Audiens\_Territ\_Update (p\_Audiens\_Territ int, p\_Territory\_ID int, p\_Audience\_ID int)  language plpgsql  as $$  begin  update Audiens\_Territ set  Territory\_ID = p\_Territory\_ID,  Audience\_ID = p\_Audience\_ID  where  ID\_Audiens\_Territ = p\_Audiens\_Territ;  end;  $$;  create or replace procedure Audiens\_Territ\_Delete (p\_ID\_Audiens\_Territ int)  language plpgsql  as $$  begin  delete from Audiens\_Territ  where  ID\_Audiens\_Territ = p\_ID\_Audiens\_Territ;  end;  $$;  create or replace procedure Combination\_Insert (p\_Login\_Employee varchar (36), p\_Department\_ID int, p\_Post\_ID int)  language plpgsql  as $$  begin  insert into Combination (Login\_Employee, Department\_ID, Post\_ID)  values (p\_Login\_Employee, p\_Department\_ID, p\_Post\_ID);  end;  $$;  create or replace procedure Combination\_Update (p\_ID\_Combination int, p\_Login\_Employee varchar (36), p\_Department\_ID int, p\_Post\_ID int)  language plpgsql  as $$  begin  update Combination set  Login\_Employee = p\_Login\_Employee,  Department\_ID = p\_Department\_ID,  Post\_ID = p\_Post\_ID  where  ID\_Combination = p\_ID\_Combination;  end;  $$;  create or replace procedure Combination\_Delete (p\_ID\_Combination int)  language plpgsql  as $$  begin  delete from Combination  where  ID\_Combination = p\_ID\_Combination;  end;  $$;  create or replace procedure Workload\_Insert (p\_N\_Hr\_Workload decimal(5,1), p\_Study\_Grpoup\_ID int,  p\_Combination\_ID int, p\_Dep\_Discipl\_ID int)  language plpgsql  as $$  begin  insert into Workload (N\_Hr\_Workload, Study\_Grpoup\_ID, Combination\_ID, Dep\_Discipl\_ID)  values (p\_N\_Hr\_Workload, p\_Study\_Grpoup\_ID, p\_Combination\_ID, p\_Dep\_Discipl\_ID);  end;  $$;  create or replace procedure Workload\_Update (p\_ID\_Workload int, p\_N\_Hr\_Workload decimal(5,1), p\_Study\_Grpoup\_ID int,  p\_Combination\_ID int, p\_Dep\_Discipl\_ID int)  language plpgsql  as $$  begin  update Workload set  N\_Hr\_Workload = p\_N\_Hr\_Workload,  Study\_Grpoup\_ID = p\_Study\_Grpoup\_ID,  Combination\_ID = p\_Combination\_ID,  Dep\_Discipl\_ID = p\_Dep\_Discipl\_ID  where  ID\_Workload = p\_ID\_Workload;  end;  $$;  create or replace procedure Workload\_Delete (p\_ID\_Workload int)  language plpgsql  as $$  begin  delete from Workload  where  ID\_Workload = p\_ID\_Workload;  end;  $$;  create or replace procedure Schedule\_Insert (p\_D\_W\_Schedule Varchar(11), p\_Pair\_N\_Schedule int,  p\_N\_S\_H\_Schedule int, p\_Audiens\_Territ\_ID int, p\_Workload\_ID int)  language plpgsql  as $$  begin  insert into Schedule (D\_W\_Schedule, Pair\_N\_Schedule, N\_S\_H\_Schedule, Audiens\_Territ\_ID, Workload\_ID)  values (p\_D\_W\_Schedule, p\_Pair\_N\_Schedule, p\_N\_S\_H\_Schedule, p\_Audiens\_Territ\_ID, p\_Workload\_ID);  end;  $$;  create or replace procedure Schedule\_Update (p\_ID\_Schedule int, p\_D\_W\_Schedule Varchar(11), p\_Pair\_N\_Schedule int,  p\_N\_S\_H\_Schedule int, p\_Audiens\_Territ\_ID int, p\_Workload\_ID int)  language plpgsql  as $$  begin  update Schedule set  D\_W\_Schedule = p\_D\_W\_Schedule,  Pair\_N\_Schedule = p\_Pair\_N\_Schedule,  N\_S\_H\_Schedule = p\_N\_S\_H\_Schedule,  Audiens\_Territ\_ID = p\_Audiens\_Territ\_ID,  Workload\_ID = p\_Workload\_ID  where  ID\_Schedule = p\_ID\_Schedule;  end;  $$;  create or replace procedure Schedule\_Delete (p\_ID\_Schedule int)  language plpgsql  as $$  begin  delete from Schedule  where  ID\_Schedule = p\_ID\_Schedule;  end;  $$;  grant execute on procedure user\_profile\_update to rl\_architect;  grant execute on procedure department\_update to rl\_architect;  grant execute on procedure department\_insert to rl\_architect;  grant execute on procedure bussines\_role\_update to rl\_architect;  grant execute on procedure bussines\_role\_insert to rl\_architect;  grant execute on procedure discipline\_update to rl\_architect;  grant execute on procedure discipline\_insert to rl\_architect;  grant execute on procedure discipline\_delete to rl\_architect;  grant execute on procedure wrkcrr\_update to rl\_architect;  grant execute on procedure wrkcrr\_insert to rl\_architect;  grant execute on procedure wrkcrr\_delete to rl\_architect;  grant execute on procedure dep\_discipl\_delete to rl\_architect;  grant execute on procedure dep\_discipl\_insert to rl\_architect;  grant execute on procedure dep\_discipl\_update to rl\_architect;  grant execute on procedure workload\_update to rl\_architect;  grant execute on procedure workload\_insert to rl\_architect;  grant execute on procedure workload\_delete to rl\_architect;  grant execute on procedure user\_profile\_update to rl\_teacher;  grant execute on procedure user\_profile\_update to rl\_student;  grant execute on procedure student\_update to rl\_student;  grant execute on procedure user\_profile\_update to rl\_managersd;  grant execute on procedure student\_update to rl\_managersd;  grant execute on procedure student\_insert to rl\_managersd;  grant execute on procedure student\_delete to rl\_managersd;  grant execute on procedure study\_grpoup\_update to rl\_managersd;  grant execute on procedure study\_grpoup\_insert to rl\_managersd;  grant execute on procedure study\_grpoup\_delete to rl\_managersd;  grant execute on procedure distrib\_grps\_delete to rl\_managersd;  grant execute on procedure distrib\_grps\_insert to rl\_managersd;  grant execute on procedure distrib\_grps\_update to rl\_managersd;  grant execute on procedure schedule\_update to rl\_managersd;  grant execute on procedure schedule\_insert to rl\_managersd;  grant execute on procedure schedule\_delete to rl\_managersd;  grant execute on procedure user\_profile\_insert to rl\_administrator;  grant execute on procedure user\_profile\_delete to rl\_administrator;  grant execute on procedure student\_update to rl\_administrator;  grant execute on procedure student\_insert to rl\_administrator;  grant execute on procedure student\_delete to rl\_administrator;  grant execute on procedure department\_update to rl\_administrator;  grant execute on procedure department\_insert to rl\_administrator;  grant execute on procedure department\_delete to rl\_administrator;  grant execute on procedure bussines\_role\_update to rl\_administrator;  grant execute on procedure bussines\_role\_insert to rl\_administrator;  grant execute on procedure bussines\_role\_delete to rl\_administrator;  grant execute on procedure post\_update to rl\_administrator;  grant execute on procedure post\_insert to rl\_administrator;  grant execute on procedure post\_delete to rl\_administrator;  grant execute on procedure territory\_delete to rl\_administrator;  grant execute on procedure territory\_insert to rl\_administrator;  grant execute on procedure territory\_update to rl\_administrator;  grant execute on procedure audience\_delete to rl\_administrator;  grant execute on procedure audience\_insert to rl\_administrator;  grant execute on procedure audience\_update to rl\_administrator;  grant execute on procedure audiens\_territ\_delete to rl\_administrator;  grant execute on procedure audiens\_territ\_insert to rl\_administrator;  grant execute on procedure audiens\_territ\_update to rl\_administrator;  grant execute on procedure employee\_delete to rl\_administrator;  grant execute on procedure employee\_insert to rl\_administrator;  grant execute on procedure employee\_update to rl\_administrator;  grant execute on procedure combination\_delete to rl\_administrator;  grant execute on procedure combination\_insert to rl\_administrator;  grant execute on procedure combination\_update to rl\_administrator;  grant execute on procedure schedule\_update to rl\_administrator;  grant execute on procedure schedule\_insert to rl\_administrator;  grant execute on procedure schedule\_delete to rl\_administrator; |
|  | 5.2 КТ Insert.sql | insert into Department (Name\_Department)  values  ('Информационные технологии');  call Department\_Insert('Информационная безопасность');  insert into Bussines\_Role (Name\_BR, Department\_ID)  values  ('Frontend разработка',1),  ('Backend Разработка',1);  call Bussines\_Role\_Insert ('Безопасность корпоративных приложений', 2);  insert into Discipline (Prefix\_Discipline, Name\_Discipline)  values  ('22ОПJS', 'Основы программирования на JS'),  ('22ОБД', 'Основы баз данных');  call Discipline\_Insert ('22ПP','Программирование на Python');  call Discipline\_Insert ('2СБ','Сетевая безопасность');  call Discipline\_Insert ('22ПМБ','Программирование модулей безопасности');  insert into Post (Name\_Post)  values  ('Архитектор кафедры');  call Post\_Insert ('Преподаватели');  insert into Territory (Name\_Territory)  values  ('ВДНХ');  call Territory\_Insert ('Тульская');  insert into Audience (Number\_Audience)  values  ('101'),  ('102'),  ('201');  call Audience\_Insert('202');  call Audience\_Insert('203');  call Audience\_Insert('204');  insert into Audiens\_Territ (Territory\_ID, Audience\_ID)  values  (1, 1),  (1, 2),  (2, 3);  call Audiens\_Territ\_Insert(2, 4);  call Audiens\_Territ\_Insert(2, 5);  call Audiens\_Territ\_Insert(2, 6);  insert into User\_Profile (UP\_Login, UP\_Password, U\_Surname, U\_Name, U\_Patronymic)  values  ('User\_Ivanov', 'Pa$$w0rd', 'Иванов', 'Иван', 'Иванович'),  ('User\_Semenov', 'Pa$$w0rd', 'Семёнов', 'Семён', 'Семёнович'),  ('User\_Borisov', 'Pa$$w0rd', 'Борисов', 'Борис', 'Борисович'),  ('User\_Petrov', 'Pa$$w0rd', 'Петров', 'Пётр', 'Петрович'),  ('User\_Antonov', 'Pa$$w0rd', 'Антонов', 'Антон', 'Антонович'),  ('User\_Kirillov', 'Pa$$w0rd', 'Кириллов', 'Кирилл', 'Кириллович'),  ('St\_User\_Andreev', 'Pa$$w0rd', 'Андреев', 'Андрей', 'Андреевич');  call User\_Profile\_Insert ('St\_User\_Pavlov', 'Pa$$w0rd', 'Павлов', 'Павел', 'Павлович');  call User\_Profile\_Insert ('St\_User\_Vladimirov', 'Pa$$w0rd', 'Владимиров', 'Владимир', '-');  call User\_Profile\_Insert ('St\_User\_Ilin', 'Pa$$w0rd', 'Ильин', 'Илья', 'Ильич');  call User\_Profile\_Insert ('St\_User\_Alekseev', 'Pa$$w0rd', 'Алексеев', 'Алексей', '-');  call User\_Profile\_Insert ('St\_User\_Egorov', 'Pa$$w0rd', 'Егоров', 'Егор', 'Егорович');  call User\_Profile\_Insert ('St\_User\_Romanov', 'Pa$$w0rd', 'Романов', 'Роман', 'Романович');  call User\_Profile\_Insert ('St\_User\_Dmitriev', 'Pa$$w0rd', 'Дмитриев', 'Дмитрий', 'Дмитриевич')  insert into Student (Login\_Student, Credit\_Card\_N)  values  ('St\_User\_Andreev', 'ЗК-0000001-22'),  ('St\_User\_Pavlov', 'ЗК-0000002-22'),  ('St\_User\_Vladimirov', 'ЗК-0000003-22'),  ('St\_User\_Ilin', 'ЗК-0000004-22');  call Student\_Insert ('St\_User\_Alekseev', 'ЗК-0000005-22');  call Student\_Insert ('St\_User\_Egorov', 'ЗК-0000006-22');  call Student\_Insert ('St\_User\_Romanov', 'ЗК-0000001-21');  call Student\_Insert ('St\_User\_Dmitriev', 'ЗК-0000002-21');  insert into Employee (Employee\_Login, Personal\_File\_N)  values  ('User\_Ivanov', 'ЛД-0000000001'),  ('User\_Semenov', 'ЛД-0000000003'),  ('User\_Borisov', 'ЛД-0000000004');  call Employee\_Insert('User\_Petrov', 'ЛД-0000000002');  call Employee\_Insert('User\_Antonov', 'ЛД-0000000005');  call Employee\_Insert('User\_Kirillov', 'ЛД-0000000006');  insert into WrkCrr (N\_Week\_WrkCrr, N\_Hours\_WrkCrr, Bussines\_Role\_ID)  values  (17, 136, 1);  call WrkCrr\_Insert(17, 102, 2);  call WrkCrr\_Insert(16, 160, 3);  insert into Study\_Grpoup (Name\_St\_Grp, Date\_Create\_SG, WrkCrr\_ID)  values  ('2П1.22', '2022-01-09', 1),  ('2П2.22', '2022-01-09', 1);  call Study\_Grpoup\_Insert ('2Р1.22', '2022-08-31', 2);  call Study\_Grpoup\_Insert ('3БК1.21', '2021-08-31', 3);  insert into Distrib\_Grps (Student\_Login, Study\_Grpoup\_ID)  values  ('St\_User\_Andreev', 1),  ('St\_User\_Pavlov', 1),  ('St\_User\_Vladimirov', 2),  ('St\_User\_Ilin', 2);  call Distrib\_Grps\_Insert('St\_User\_Alekseev', 3);  call Distrib\_Grps\_Insert('St\_User\_Egorov', 3);  call Distrib\_Grps\_Insert('St\_User\_Romanov', 4);  call Distrib\_Grps\_Insert('St\_User\_Dmitriev', 4);  insert into Combination (Login\_Employee, Department\_ID, Post\_ID)  values  ('User\_Ivanov', 1, 1),  ('User\_Semenov', 1, 2),  ('User\_Borisov', 1, 2);  call Combination\_Insert('User\_Petrov', 2, 1);  call Combination\_Insert('User\_Antonov', 2, 2);  call Combination\_Insert('User\_Kirillov', 2, 2);  insert into Dep\_Discipl (Course\_N, Semester\_N, N\_H\_Dep\_Discipl, WrkCrr\_ID, Discipline\_ID)  values  (2, 1, 68, 1, 1),  (2, 1, 68, 1, 2),  (2, 1, 68, 2, 2);  call Dep\_Discipl\_Insert(2, 1, 34, 2, 3);  call Dep\_Discipl\_Insert(3, 1, 96, 3, 4);  call Dep\_Discipl\_Insert(3, 1, 64, 3, 5);  insert into Workload (N\_Hr\_Workload, Study\_Grpoup\_ID, Combination\_ID, Dep\_Discipl\_ID)  values  (272, 1, 2, 1),  (272, 2, 2, 1),  (340, 1, 3, 2),  (340, 2, 3, 2),  (340, 3, 3, 3);  call Workload\_Insert (164, 1, 5, 1);  call Workload\_Insert (164, 4, 5, 5);  call Workload\_Insert (98, 4, 6, 6);  call Workload\_Insert (98, 3, 6, 4);  insert into Schedule (D\_W\_Schedule, Pair\_N\_Schedule, N\_S\_H\_Schedule, Audiens\_Territ\_ID, Workload\_ID)  values  ('Понедельник', 1, 12, 1, 1),  ('Понедельник', 2, 12, 2, 3),  ('Понедельник', 2, 4, 1, 2),  ('Понедельник', 3, 4, 2, 4),  ('Вторник', 1, 6, 2, 5),  ('Вторник', 2, 6, 2, 5);  call Schedule\_Insert('Вторник', 1, 12, 3, 6);  call Schedule\_Insert('Вторник', 2, 12, 4, 1);  call Schedule\_Insert('Среда', 1, 4, 3, 7);  call Schedule\_Insert('Среда', 2, 4, 5, 8);  call Schedule\_Insert('Среда', 1, 12, 4, 1);  call Schedule\_Insert('Среда', 2, 12, 6, 3);  call Schedule\_Insert('Среда', 4, 6, 5, 9); |

1. Перезапись структуры;

Таблица 2 – Файл перезаписи структуры БД

|  |  |
| --- | --- |
| Название файла | Скрипт |
| 5 КТ Re-Create.sql | create or replace procedure Structure\_Re\_Create ()  language plpgsql  as $$  begin  revoke execute on procedure user\_profile\_update from rl\_architect;  revoke execute on procedure department\_update from rl\_architect;  revoke execute on procedure department\_insert from rl\_architect;  revoke execute on procedure bussines\_role\_update from rl\_architect;  revoke execute on procedure bussines\_role\_insert from rl\_architect;  revoke execute on procedure discipline\_update from rl\_architect;  revoke execute on procedure discipline\_insert from rl\_architect;  revoke execute on procedure discipline\_delete from rl\_architect;  revoke execute on procedure wrkcrr\_update from rl\_architect;  revoke execute on procedure wrkcrr\_insert from rl\_architect;  revoke execute on procedure wrkcrr\_delete from rl\_architect;  revoke execute on procedure dep\_discipl\_delete from rl\_architect;  revoke execute on procedure dep\_discipl\_insert from rl\_architect;  revoke execute on procedure dep\_discipl\_update from rl\_architect;  revoke execute on procedure workload\_update from rl\_architect;  revoke execute on procedure workload\_insert from rl\_architect;  revoke execute on procedure workload\_delete from rl\_architect;  revoke execute on procedure user\_profile\_update from rl\_teacher;  revoke execute on procedure user\_profile\_update from rl\_student;  revoke execute on procedure student\_update from rl\_student;  revoke execute on procedure user\_profile\_update from rl\_managersd;  revoke execute on procedure student\_update from rl\_managersd;  revoke execute on procedure student\_insert from rl\_managersd;  revoke execute on procedure student\_delete from rl\_managersd;  revoke execute on procedure study\_grpoup\_update from rl\_managersd;  revoke execute on procedure study\_grpoup\_insert from rl\_managersd;  revoke execute on procedure study\_grpoup\_delete from rl\_managersd;  revoke execute on procedure distrib\_grps\_delete from rl\_managersd;  revoke execute on procedure distrib\_grps\_insert from rl\_managersd;  revoke execute on procedure distrib\_grps\_update from rl\_managersd;  revoke execute on procedure schedule\_update from rl\_managersd;  revoke execute on procedure schedule\_insert from rl\_managersd;  revoke execute on procedure schedule\_delete from rl\_managersd;  revoke execute on procedure user\_profile\_insert from rl\_administrator;  revoke execute on procedure user\_profile\_delete from rl\_administrator;  revoke execute on procedure student\_update from rl\_administrator;  revoke execute on procedure student\_insert from rl\_administrator;  revoke execute on procedure student\_delete from rl\_administrator;  revoke execute on procedure department\_update from rl\_administrator;  revoke execute on procedure department\_insert from rl\_administrator;  revoke execute on procedure department\_delete from rl\_administrator;  revoke execute on procedure bussines\_role\_update from rl\_administrator;  revoke execute on procedure bussines\_role\_insert from rl\_administrator;  revoke execute on procedure bussines\_role\_delete from rl\_administrator;  revoke execute on procedure post\_update from rl\_administrator;  revoke execute on procedure post\_insert from rl\_administrator;  revoke execute on procedure post\_delete from rl\_administrator;  revoke execute on procedure territory\_delete from rl\_administrator;  revoke execute on procedure territory\_insert from rl\_administrator;  revoke execute on procedure territory\_update from rl\_administrator;  revoke execute on procedure audience\_delete from rl\_administrator;  revoke execute on procedure audience\_insert from rl\_administrator;  revoke execute on procedure audience\_update from rl\_administrator;  revoke execute on procedure audiens\_territ\_delete from rl\_administrator;  revoke execute on procedure audiens\_territ\_insert from rl\_administrator;  revoke execute on procedure audiens\_territ\_update from rl\_administrator;  revoke execute on procedure employee\_delete from rl\_administrator;  revoke execute on procedure employee\_insert from rl\_administrator;  revoke execute on procedure employee\_update from rl\_administrator;  revoke execute on procedure combination\_delete from rl\_administrator;  revoke execute on procedure combination\_insert from rl\_administrator;  revoke execute on procedure combination\_update from rl\_administrator;  revoke execute on procedure schedule\_update from rl\_administrator;  revoke execute on procedure schedule\_insert from rl\_administrator;  revoke execute on procedure schedule\_delete from rl\_administrator;  drop procedure audience\_delete(p\_id\_audience);  drop procedure audience\_insert(p\_number\_audience);  drop procedure audience\_update(p\_id\_audience, p\_number\_audience);  drop procedure audiens\_territ\_delete(p\_id\_audiens\_territ);  drop procedure audiens\_territ\_insert(p\_territory\_id, p\_audience\_id);  drop procedure audiens\_territ\_update(p\_audiens\_territ, p\_territory\_id, p\_audience\_id);  drop procedure bussines\_role\_delete(p\_id\_bussines\_role);  drop procedure bussines\_role\_insert(p\_name\_br, p\_department\_id);  drop procedure bussines\_role\_update(p\_id\_bussines\_role, p\_name\_br, p\_department\_id);  drop procedure combination\_delete(p\_id\_combination);  drop procedure combination\_insert(p\_login\_employee, p\_department\_id, p\_post\_id);  drop procedure combination\_update(p\_id\_combination, p\_login\_employee, p\_department\_id, p\_post\_id);  drop procedure dep\_discipl\_delete(p\_id\_dep\_discipl);  drop procedure dep\_discipl\_insert(p\_course\_n, p\_semester\_n, p\_n\_h\_dep\_discipl, p\_wrkcrr\_id, p\_discipline\_id);  drop procedure dep\_discipl\_update(p\_id\_dep\_discipl, p\_course\_n, p\_semester\_n, p\_n\_h\_dep\_discipl, p\_wrkcrr\_id, p\_discipline\_id);  drop procedure department\_delete(p\_id\_department);  drop procedure department\_insert(p\_name\_department);  drop procedure department\_update(p\_id\_department, p\_name\_department);  drop procedure discipline\_delete(p\_id\_discipline);  drop procedure discipline\_insert(p\_prefix\_discipline, p\_name\_discipline);  drop procedure discipline\_update(p\_id\_discipline, p\_prefix\_discipline, p\_name\_discipline);  drop procedure department\_delete(p\_id\_department);  drop procedure department\_insert(p\_name\_department);  drop procedure department\_update(p\_id\_department, p\_name\_department);  drop procedure discipline\_delete(p\_id\_discipline);  drop procedure discipline\_insert(p\_prefix\_discipline, p\_name\_discipline);  drop procedure discipline\_update(p\_id\_discipline, p\_prefix\_discipline, p\_name\_discipline);  drop procedure distrib\_grps\_delete(p\_id\_distrib\_grps);  drop procedure distrib\_grps\_insert(p\_student\_login, p\_study\_grpoup\_id);  drop procedure distrib\_grps\_update(p\_id\_distrib\_grps, p\_student\_login, p\_study\_grpoup\_id);  drop procedure employee\_delete(p\_employee\_login);  drop procedure employee\_insert(p\_employee\_login, p\_personal\_file\_n);  drop procedure employee\_update(p\_employee\_login\_old, p\_employee\_login\_new, p\_personal\_file\_n);  drop procedure post\_delete(p\_id\_post);  drop procedure post\_insert(p\_name\_post);  drop procedure post\_update(p\_id\_post, p\_name\_post);  drop procedure schedule\_delete(p\_id\_schedule);  drop procedure schedule\_insert(p\_d\_w\_schedule, p\_pair\_n\_schedule, p\_n\_s\_h\_schedule, p\_audiens\_territ\_id, p\_workload\_id);  drop procedure schedule\_update(p\_id\_schedule, p\_d\_w\_schedule, p\_pair\_n\_schedule, p\_n\_s\_h\_schedule, p\_audiens\_territ\_id, p\_workload\_id);  drop procedure student\_delete(p\_login\_student);  drop procedure student\_insert(p\_login\_student, p\_credit\_card\_n);  drop procedure student\_update(p\_login\_student\_old, p\_login\_student\_new, p\_credit\_card\_n);  drop procedure study\_grpoup\_delete(p\_id\_study\_grpoup);  drop procedure study\_grpoup\_insert(p\_name\_st\_grp, p\_date\_create\_sg, p\_wrkcrr\_id);  drop procedure student\_update(p\_login\_student\_old, p\_login\_student\_new, p\_credit\_card\_n);  drop procedure territory\_delete(p\_id\_territory);  drop procedure territory\_insert(p\_name\_territory);  drop procedure territory\_update(p\_id\_territory, p\_name\_territory);  drop procedure user\_profile\_delete(p\_up\_login);  drop procedure user\_profile\_insert(p\_up\_login, p\_up\_password, p\_u\_surname, p\_u\_name, p\_u\_patronymic);  drop procedure user\_profile\_update(p\_up\_login\_old, p\_up\_login\_new, p\_up\_password, p\_u\_surname, p\_u\_name, p\_u\_patronymic);  drop procedure workload\_delete(p\_id\_workload);  drop procedure workload\_insert(p\_n\_hr\_workload, p\_study\_grpoup\_id, p\_combination\_id, p\_dep\_discipl\_id);  drop procedure workload\_update(p\_id\_workload, p\_n\_hr\_workload, p\_study\_grpoup\_id, p\_combination\_id, p\_dep\_discipl\_id);  drop procedure wrkcrr\_delete(p\_id\_wrkcrr);  drop procedure wrkcrr\_insert(p\_n\_week\_wrkcrr, p\_n\_hours\_wrkcrr, p\_bussines\_role\_id);  drop procedure wrkcrr\_update(p\_id\_wrkcrr, p\_n\_week\_wrkcrr, p\_n\_hours\_wrkcrr, p\_bussines\_role\_id);  revoke usage, select on sequence department\_id\_department\_seq from rl\_architect;  revoke usage, select on sequence bussines\_role\_id\_bussines\_role\_seq from rl\_architect;  revoke usage, select on sequence discipline\_id\_discipline\_seq from rl\_architect;  revoke usage, select on sequence wrkcrr\_id\_wrkcrr\_seq from rl\_architect;  revoke usage, select on sequence dep\_discipl\_id\_dep\_discipl\_seq from rl\_architect;  revoke usage, select on sequence workload\_id\_workload\_seq from rl\_architect;  revoke usage, select on sequence study\_grpoup\_id\_study\_grpoup\_seq from rl\_managersd;  revoke usage, select on sequence distrib\_grps\_id\_distrib\_grps\_seq from rl\_managersd;  revoke usage, select on sequence schedule\_id\_schedule\_seq from rl\_managersd;  revoke usage, select on sequence department\_id\_department\_seq from rl\_administrator;  revoke usage, select on sequence post\_id\_post\_seq from rl\_administrator;  revoke usage, select on sequence territory\_id\_territory\_seq from rl\_administrator;  revoke usage, select on sequence audience\_id\_audience\_seq from rl\_administrator;  revoke usage, select on sequence bussines\_role\_id\_bussines\_role\_seq from rl\_administrator;  revoke usage, select on sequence audiens\_territ\_id\_audiens\_territ\_seq from rl\_administrator;  revoke usage, select on sequence combination\_id\_combination\_seq from rl\_administrator;  revoke usage, select on sequence schedule\_id\_schedule\_seq from rl\_administrator;  revoke select, insert, update on Department from rl\_architect;  revoke select, insert, update on Bussines\_Role from rl\_architect;  revoke select on Post from rl\_architect;  revoke select on Study\_Grpoup from rl\_architect;  revoke select on Student from rl\_architect;  revoke select on Distrib\_Grps from rl\_architect;  revoke select on Audiens\_Territ from rl\_architect;  revoke select on Combination from rl\_architect;  revoke select on Schedule from rl\_architect;  revoke select, insert, update, delete on Discipline from rl\_architect;  revoke select, insert, update, delete on WrkCrr from rl\_architect;  revoke select, insert, update, delete on Dep\_Discipl from rl\_architect;  revoke select, insert, update, delete on Workload from rl\_architect;  revoke select, update on User\_Profile from rl\_architect;  revoke select, update on User\_Profile from rl\_teacher;  revoke select on Student from rl\_teacher;  revoke select on Schedule from rl\_teacher;  revoke select, update on User\_Profile from rl\_student;  revoke select, update on Student from rl\_student;  revoke select on Schedule from rl\_student;  revoke select, update on User\_Profile from rl\_managersd;  revoke select, insert, update, delete on Study\_Grpoup from rl\_managersd;  revoke select, insert, update, delete on Student from rl\_managersd;  revoke select, insert, update, delete on Distrib\_Grps from rl\_managersd;  revoke select, insert, update, delete on Schedule from rl\_managersd;  revoke select, insert, update, delete on Department from rl\_administrator;  revoke select, insert, update, delete on Post from rl\_administrator;  revoke select, insert, update, delete on Territory from rl\_administrator;  revoke select, insert, update, delete on Audience from rl\_administrator;  revoke select, insert, update, delete on Bussines\_Role from rl\_administrator;  revoke select, insert, update, delete on Student from rl\_administrator;  revoke select, insert, update, delete on Employee from rl\_administrator;  revoke select, insert, update, delete on Audiens\_Territ from rl\_administrator;  revoke select, insert, update, delete on Combination from rl\_administrator;  revoke select, insert, update, delete on Schedule from rl\_administrator;  revoke select, insert, delete on User\_Profile from rl\_administrator;  drop index if exists index\_Pair\_N\_Schedule;  drop index if exists index\_D\_W\_Schedule;  drop index if exists index\_ID\_Schedule;  drop index if exists index\_ID\_Workload;  drop index if exists index\_ID\_Combination;  drop index if exists index\_ID\_Audiens\_Territ;  drop index if exists index\_ID\_Distrib\_Grps;  drop index if exists index\_Semester\_N;  drop index if exists index\_Course\_N;  drop index if exists index\_ID\_Dep\_Discipl;  drop index if exists index\_Personal\_File\_N;  drop index if exists index\_Employee\_Login;  drop index if exists index\_Credit\_Card\_N;  drop index if exists index\_Login\_Student;  drop index if exists index\_Date\_Create\_SG;  drop index if exists index\_Name\_St\_Grp;  drop index if exists index\_ID\_Study\_Grpoup;  drop index if exists index\_ID\_WrkCrr;  drop index if exists index\_Name\_BR;  drop index if exists index\_ID\_Bussines\_Role;  drop index if exists index\_Number\_Audience;  drop index if exists index\_ID\_Audience;  drop index if exists index\_Name\_Territory;  drop index if exists index\_ID\_Territory;  drop index if exists index\_U\_Surname\_Name\_Patronymic;  drop index if exists index\_UP\_Login\_Password;  drop index if exists index\_Prefix\_Name\_Discipline;  drop index if exists index\_ID\_Discipline;  drop index if exists index\_Name\_Post;  drop index if exists index\_ID\_Post;  drop index if exists index\_Name\_Department;  drop index if exists index\_ID\_Department;  drop table if exists Schedule;  drop table if exists Workload;  drop table if exists Combination;  drop table if exists Audiens\_Territ;  drop table if exists Distrib\_Grps;  drop table if exists Dep\_Discipl;  drop table if exists Employee;  drop table if exists Student;  drop table if exists Study\_Grpoup;  drop table if exists WrkCrr;  drop table if exists Bussines\_Role;  drop table if exists Audience;  drop table if exists Territory;  drop table if exists User\_Profile;  drop table if exists Discipline;  drop table if exists Post;  drop table if exists Department;  call Structure\_Create();  end;  $$; |

1. Создание резервной копии;



1. Версия базы данных:
   1. Отчёт о созданных объектах и записях;

Таблица 3 – Перечень созданных объектов

|  | Информация по объектам |
| --- | --- |
| Запрос | select  information\_schema.tables.table\_name as "Название таблиц",  string\_agg(information\_schema.routines.routine\_name, ', ') as "Список процедур",  n\_live\_tup as "Кол-во записей в таблицах"  from information\_schema.tables  inner join information\_schema.routines  on information\_schema.tables.table\_name = substring(information\_schema.routines.routine\_name, 1, length(information\_schema.tables.table\_name))  inner join pg\_stat\_user\_tables  on information\_schema.tables.table\_name = pg\_stat\_user\_tables.relname  where  information\_schema.tables.table\_schema = 'public'and  routine\_type = 'PROCEDURE'  group by  table\_name,  n\_live\_tup  union all  select  'Количество процедур: ',  count(information\_schema.routines.routine\_name)::text,  (select  sum(n\_live\_tup)  from pg\_stat\_user\_tables)  from information\_schema.routines  where  routine\_type = 'PROCEDURE' and  routine\_name not in ('structure\_create','structure\_re\_create'); |
| Результат локальной БД |  |
| Результат удалённой БД |  |

* 1. Версия БД.

Таблица 4 – Версия файла БД

|  |  |
| --- | --- |
| Параметры | PostgreSQL |
| Номер версии | 2.2.0.1 |
| Что сделано | * Созданы 51 хранимые процедуры; * Добавлено 103 строк во всю структуру БД; * Произведено распределение доступа ролей к хранимым процедурам; * Сопровождены процедуры, для быстрого создания и перезаписи всей структуры таблиц, индексов, прав доступа, заполнения данных, создания хранимых процедур; * Создан Backup файл. |