SET D- Project-Employee Database

Consider the following Entities and their Relationships for ProjectEmployee database.
Project (pno integer, pname char (30), ptype char (20), duration integer)
Employee (eno integer, ename char (20), qualification char (15), joining_date date)
Relationship between Project and Employee is many to many with descriptive attribute start_date date, no_of_hours_worked integer.
Constraints: Primary Key,
duration should be greater than zero, pname should not be null.
Queries:
1. Write a stored function to find the number of employees whose joining date is before '01/01/2007'.
2. Write a stored function to accept eno as input parameter and count number of projects on which that employee is working.
3. Write a stored function to accept project name and display employee details who worked more than 2000 hours.
4. Write a stored function to display all projects started after date "01/01/2019".

```
create table project(pno int primary key, pname varchar(30) NOT NULL,
 ptype varchar(20),
 duration int check (duration > 0)
);

insert into project values (1,'ERP','Management',10);
insert into project values (2,'SAP','Civil',30);
insert into project values (3,'TTT','Manufacturing',4);
insert into project values (4,'TOM','Civil',5);
insert into project values (5,'HTI','Management',7);
insert into project values (6,'X-T','Civil',50);
```

```
  select * from project;

create table employee(
 eno int primary key,
 ename varchar(20),
 qualification varchar(15),
 joining_date date
);

insert into employee values (11,'John','MCA','2020-09-01');
insert into employee values (12,'Smith','MCS','1998-10-01');
insert into employee values (13,'Ron','BBA-CA','2000-03-01');
insert into employee values (14,'Roy','MCA','2003-12-01');
insert into employee values (15,'Alex','MCS','2000-11-10');
insert into employee values (16,'David','BBA-CA','2001-08-01');
insert into employee values (17,'Ben','MCS','1990-10-10');
insert into employee values (18,'Sam','MCA','1995-12-12');

  select * from employee;

create table pro_emp(pno int references project(pno),
 eno int references employee(eno),start_date date,no_of_hrs int);

 insert into pro_emp values (1,11,'2020-09-10',20);
insert into pro_emp values (2,12,'2000-10-12',700);
insert into pro_emp values (3,13,'2010-06-06',400);
insert into pro_emp values (4,14,'2000-10-16',350);
insert into pro_emp values (5,15,'1999-01-20',250);
insert into pro_emp values (1,16,'2000-10-12',25);
insert into pro_emp values (2,12,'2000-10-12',800);
insert into pro_emp values (3,13,'2010-06-06',320);
insert into pro_emp values (6,17,'1995-12-23',2010);
insert into pro_emp values (6,18,'1995-12-23',2010);
```

```
select * from pro_emp;
```

Write a stored function to find the number of employees whose joining date is before ‘01/01/2007’.

```
create or replace function noofemp() returns int as
'
 declare
 cnt int;
 begin
 select count(*) into cnt
 from employee
 where joining_date < "2007-01-01";

 return cnt;
 end;
'language 'plpgsql';

select noofemp();
 noofemp
---------
 7
```

3. Write a stored function to accept project name and display employee details who worked more than 2000 hours.

```
create or replace function empwork(name text) returns void as
'
 declare
 rec record;
 begin
```

```
 raise notice "ENO || Name || Qualification ||
 Joining Date";

 for rec in
 select employee.eno,ename,qualification,joining_date
 from employee,project,pro_emp
 where employee.eno = pro_emp.eno
 and project.pno = pro_emp.pno
 and pname = name loop
 raise notice ' '% || % || % ||
 %' ',rec.eno,rec.ename,rec.qualification,rec.joining_date;
 end loop;
 end;
'language 'plpgsql';

select empwork('X-T');
NOTICE: ENO||Name||Qualification||Joining Date
NOTICE: 17||Ben||MCS||1990-10-10
NOTICE: 18||Sam||MCA||1995-12-12
 empwork
---------
```

2. Write a stored function to accept eno as input parameter and count number of projects on which that employee is working.

```
create or replace function countp(n int) returns int as
'
 declare
 cnt int;
```

```
 begin
 select count(*) into cnt
 from pro_emp
 where eno = n;
 return cnt;
 end;
'language 'plpgsql';
select countp(12);
 countp
--------
 2
```