МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«Московский институт электронной техники»

Кафедра СПИНТЕХ

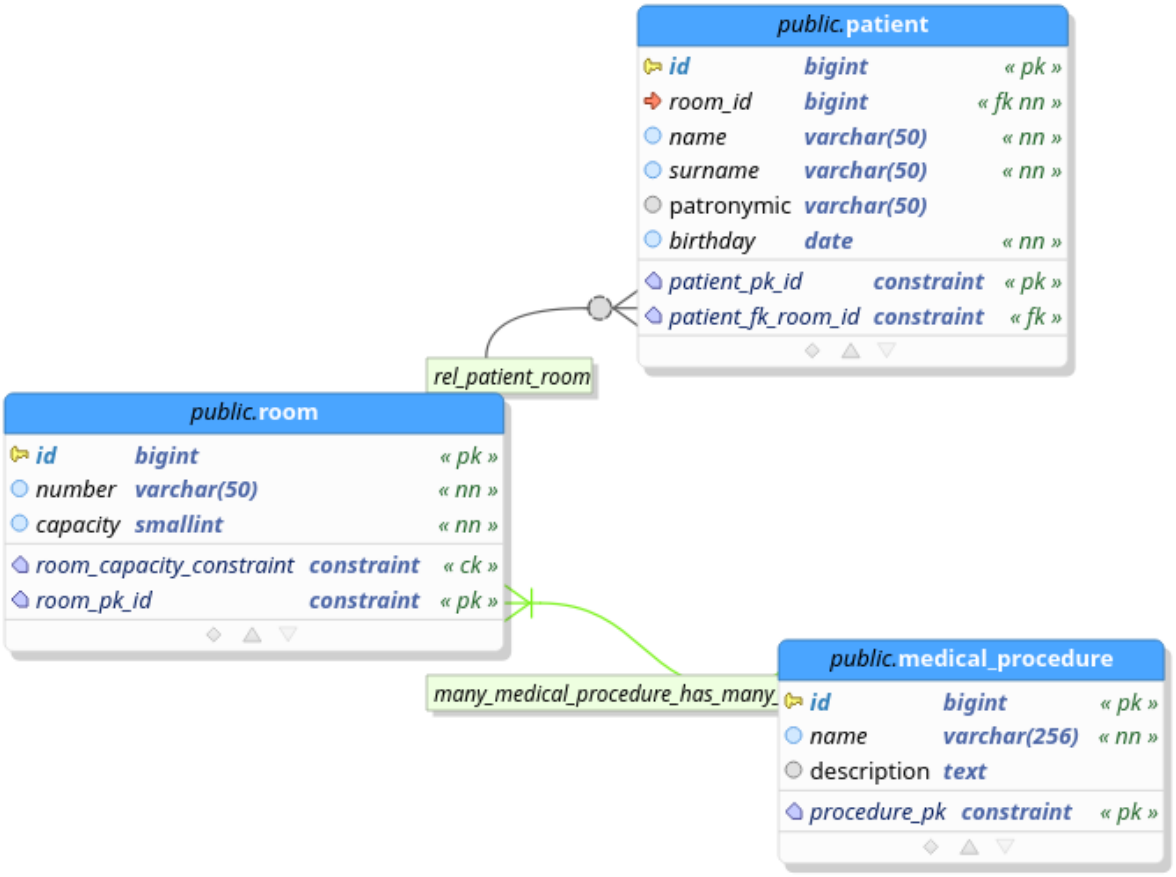
Отчёт по Лабораторной работе №3
по предмету: «СУБД»

“Проектирование БД для СУБД MySQL и оценка ее размеров средствами ERWIN”

Выполнил:
Студент группы ПИН-33
Новосельцев Ст. А.

г. Москва
2021

Спроектированная БД с помощью утилиты PGModeler



Сгенерированный SQL-запрос на создание БД

```
-- Database generated with pgModeler (PostgreSQL Database Modeler).
-- pgModeler version: 0.9.4
-- PostgreSQL version: 13.0
-- Project Site: pgmodeler.io
-- Model Author: ---

-- Database creation must be performed outside a multi lined SQL file.
-- These commands were put in this file only as a convenience.
--
-- object: new_database | type: DATABASE --
-- DROP DATABASE IF EXISTS new_database;
CREATE DATABASE new_database;
-- ddl-end --


-- object: public.room | type: TABLE --
-- DROP TABLE IF EXISTS public.room CASCADE;
```

```
CREATE TABLE public.room (
      id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ,
      number varchar(50) NOT NULL,
      capacity smallint NOT NULL,
      CONSTRAINT room_capacity_constraint CHECK (capacity > 0),
      CONSTRAINT room_pk_id PRIMARY KEY (id)
);
-- ddl-end --
ALTER TABLE public.room OWNER TO postgres;
-- ddl-end --

-- object: public.patient | type: TABLE --
-- DROP TABLE IF EXISTS public.patient CASCADE;
CREATE TABLE public.patient (
      id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ,
      room_id bigint NOT NULL,
      name varchar(50) NOT NULL,
      surname varchar(50) NOT NULL,
      patronymic varchar(50),
      birthday date NOT NULL,
      CONSTRAINT patient_pk_id PRIMARY KEY (id)
);
-- ddl-end --
ALTER TABLE public.patient OWNER TO postgres;
-- ddl-end --

-- object: public.medical_procedure | type: TABLE --
-- DROP TABLE IF EXISTS public.medical_procedure CASCADE;
CREATE TABLE public.medical_procedure (
      id bigint NOT NULL GENERATED ALWAYS AS IDENTITY ,
      name varchar(256) NOT NULL,
      description text,
      CONSTRAINT procedure_pk PRIMARY KEY (id)
);
-- ddl-end --
ALTER TABLE public.medical_procedure OWNER TO postgres;
-- ddl-end --

-- object: public.many_medical_procedure_has_many_patient | type: TABLE
--
-- DROP TABLE IF EXISTS public.many_medical_procedure_has_many_patient
CASCADE;
CREATE TABLE public.many_medical_procedure_has_many_patient (
      id_medical_procedure bigint NOT NULL,
      id_patient bigint NOT NULL,
      CONSTRAINT many_medical_procedure_has_many_patient_pk PRIMARY KEY
```

```
(id_medical_procedure,id_patient)
);
-- ddl-end --

-- object: medical_procedure_fk | type: CONSTRAINT --
-- ALTER TABLE public.many_medical_procedure_has_many_patient DROP
CONSTRAINT IF EXISTS medical_procedure_fk CASCADE;
ALTER TABLE public.many_medical_procedure_has_many_patient ADD
CONSTRAINT medical_procedure_fk FOREIGN KEY (id_medical_procedure)
REFERENCES public.medical_procedure (id) MATCH FULL
ON DELETE RESTRICT ON UPDATE CASCADE;
-- ddl-end --

-- object: patient_fk | type: CONSTRAINT --
-- ALTER TABLE public.many_medical_procedure_has_many_patient DROP
CONSTRAINT IF EXISTS patient_fk CASCADE;
ALTER TABLE public.many_medical_procedure_has_many_patient ADD
CONSTRAINT patient_fk FOREIGN KEY (id_patient)
REFERENCES public.patient (id) MATCH FULL
ON DELETE RESTRICT ON UPDATE CASCADE;
-- ddl-end --

-- object: patient_fk_room_id | type: CONSTRAINT --
-- ALTER TABLE public.patient DROP CONSTRAINT IF EXISTS
patient_fk_room_id CASCADE;
ALTER TABLE public.patient ADD CONSTRAINT patient_fk_room_id FOREIGN KEY
(room_id)
REFERENCES public.room (id) MATCH SIMPLE
ON DELETE NO ACTION ON UPDATE NO ACTION;
-- ddl-end --
```

Написанный вручную к ЛР2 SQL-запрос на создание БД

```
CREATE TABLE public.room (
    id bigint NOT NULL GENERATED ALWAYS AS IDENTITY,
    number varchar(32) NOT NULL,
    capacity smallint CHECK (capacity > 0),

    CONSTRAINT room_pk PRIMARY KEY (id)
);

CREATE TABLE public.patient (
    id bigint NOT NULL GENERATED ALWAYS AS IDENTITY,
    room_id bigint not null,
```

```
    name varchar(50) not null,
    surname varchar(50) not null,
    patronymic varchar(50) null,
    birthday date not null,

    CONSTRAINT patient_pk PRIMARY KEY (id),
    CONSTRAINT patient_fk_room_id FOREIGN KEY (room_id)
        REFERENCES public.room (id) MATCH SIMPLE
            ON DELETE NO ACTION ON UPDATE NO ACTION
);

CREATE TABLE public.medical_procedure (
    id bigint NOT NULL GENERATED ALWAYS AS IDENTITY,
    name varchar(256) NOT NULL,
    description TEXT NULL,

    CONSTRAINT medical_procedure_pk PRIMARY KEY (id)
);

CREATE TABLE public.patient_procedure (
    patient_id bigint NOT NULL,
    medical_procedure_id bigint NOT NULL,
    duration interval NULL,
    assigned_quantity smallint NOT NULL CHECK (assigned_quantity > 0),
    completed_quantity smallint NOT NULL DEFAULT 0 CHECK
(completed_quantity >= 0),

    CONSTRAINT patient_procedure_pk PRIMARY KEY (patient_id,
medical_procedure_id),
    CONSTRAINT patient_procedure_fk_patient_id FOREIGN KEY (patient_id)
       REFERENCES public.patient (id) MATCH SIMPLE
           ON DELETE NO ACTION ON UPDATE NO ACTION,

    CONSTRAINT patient_procedure_fk_m_p_id FOREIGN KEY
(medical_procedure_id)
       REFERENCES public.medical_procedure (id) MATCH SIMPLE
           ON DELETE NO ACTION ON UPDATE NO ACTION

);
```