1. Добавление сотрудника - процедура

CREATE OR REPLACE FUNCTION addnewstaff(stname VARCHAR, surname VARCHAR, patr VARCHAR, tel VARCHAR,

stlogin VARCHAR, rassw VARCHAR, serv VARCHAR, typeserv VARCHAR, pos VARCHAR) RETURNS INT AS $$

DECLARE

idserv INT;

userrole INT;

sal NUMERIC;

BEGIN

idserv := (SELECT id\_service FROM service WHERE specialization\_service=$7 AND type\_service=$8);

IF $9='Директор' THEN userrole = 1;

sal = 30000;

ELSIF $9='Администратор фитнес клуба' THEN userrole = 2;

sal = 22000;

ELSIF $9='Тренер' THEN userrole = 3;

sal = 15000;

ELSIF $9='Массажист' THEN userrole = 3;

sal = 10000;

END IF;

INSERT INTO staff (service\_id, name\_staff, surname\_staff, patronomyc\_staff,

mobile\_telephone\_staff, position\_staff, salary\_₴, login, passw, role\_to\_login)

VALUES (idserv, $1, $2, $3, $4, $9, sal, $5, $6, userrole);

RETURN 1;

END;

$$ LANGUAGE plpgsql;

SELECT \* FROM addnewstaff('Андрей','Иваненко','Максиимович','380421302356','AndrIvanMaxi','andr2', 'Zumba', 'Групповая','Тренер');

1. Удаление сотрудника - процедура

CREATE OR REPLACE FUNCTION deletestaff(stname VARCHAR, surname VARCHAR, patr VARCHAR) RETURNS INT AS $$

BEGIN

DELETE FROM staff

WHERE name\_staff=$1 AND surname\_staff=$2 AND patronomyc\_staff=$3;

RETURN 1;

END;

$$ LANGUAGE plpgsql;

SELECT \* FROM deletestaff('Алексей','Григорьев','Сергеевич');

1. Просмотреть доступные услуги

CREATE VIEW servicesfordirector AS

(SELECT id\_service AS "Номер", specialization\_service AS "Услуга", type\_service AS "Вид" FROM service);

select \* from servicesfordirector;

1. Добавить услугу

CREATE OR REPLACE FUNCTION addserv(spec VARCHAR, type VARCHAR ) RETURNS TABLE("Номер" INT, "Услуга" VARCHAR, "Вид" VARCHAR) AS $$

BEGIN

INSERT INTO service(specialization\_service, type\_service)

VALUES ($1,$2);

RETURN QUERY(SELECT \* FROM servicesfordirector);

END;

$$ LANGUAGE plpgsql

SELECT \* FROM addserv('Администрирование', 'Индвидуальная');

1. Удалить услугу

CREATE OR REPLACE FUNCTION delserv(spec VARCHAR, type VARCHAR ) RETURNS TABLE("Номер" INT, "Услуга" VARCHAR, "Вид" VARCHAR) AS $$

DECLARE

staff\_id int;

spec\_id int;

cost\_id int;

BEGIN

select id\_service into spec\_id from service WHERE type\_service = $2 and specialization\_service = $1;

FOR staff\_id IN SELECT id\_staff FROM staff WHERE service\_id = spec\_id loop

update staff

set service\_id = NUll

where id\_staff = staff\_id;

end loop;

FOR cost\_id IN SELECT id\_ca FROM cost\_ab WHERE serv\_id = spec\_id loop

update cost\_ab

set serv\_id = NUll

where id\_ca = cost\_id;

end loop;

DELETE FROM service WHERE specialization\_service=$1 AND type\_service=$2;

RETURN QUERY(SELECT \* FROM servicesfordirector);

END;

$$ LANGUAGE plpgsql;

SELECT \* FROM delserv('Администрирование', 'Индвидуальная');

1. Информация о рабочей смене

CREATE OR REPLACE FUNCTION wsinfo(d1 TIMESTAMP) RETURNS TABLE

("Имя" VARCHAR, "Фамилия" VARCHAR, "Отчество" VARCHAR, "День" VARCHAR, "Начало" TIME WITHOUT TIME ZONE, "Конец" TIME WITHOUT TIME ZONE) AS $$

DECLARE

d\_o\_w INT;

dayoftheweek VARCHAR;

BEGIN

d\_o\_w = (SELECT EXTRACT(DOW FROM $1));

IF d\_o\_w = 0 THEN dayoftheweek = 'Воскресенье';

ELSIF d\_o\_w = 1 THEN dayoftheweek = 'Понедельник';

ELSIF d\_o\_w = 2 THEN dayoftheweek = 'Вторник';

ELSIF d\_o\_w = 3 THEN dayoftheweek = 'Среда';

ELSIF d\_o\_w = 4 THEN dayoftheweek = 'Четверг';

ELSIF d\_o\_w = 5 THEN dayoftheweek = 'Пятница';

ELSIF d\_o\_w = 6 THEN dayoftheweek = 'Суббота';

END IF;

RETURN QUERY(SELECT name\_staff, surname\_staff, patronomyc\_staff, date\_workshift,

begin\_time\_workshift, end\_time\_workshift FROM workshift JOIN staff ON workshift\_staff=id\_staff WHERE date\_workshift=dayoftheweek);

END;

$$ LANGUAGE plpgsql

SELECT \* FROM wsinfo(now()::TIMESTAMP);

1. Добавить клиента

CREATE OR REPLACE FUNCTION addnewclient(cname VARCHAR, surname VARCHAR, patr VARCHAR, tel VARCHAR) RETURNS TABLE

("Номер карты" INT,"Имя" VARCHAR, "Фамилия" VARCHAR, "Отчество" VARCHAR, "Телефон" CHAR) AS $$

DECLARE

cardn INT;

BEGIN

cardn = (SELECT max(card\_number\_client) FROM client)+1;

INSERT INTO client (card\_number\_client,name\_client, surname\_client, patronomyc\_client, mobile\_telephone\_client) VALUES

(cardn, $1, $2, $3, $4);

RETURN QUERY(SELECT card\_number\_client, name\_client, surname\_client, patronomyc\_client, mobile\_telephone\_client FROM client WHERE card\_number\_client=cardn);

END;

$$ LANGUAGE plpgsql

SELECT \* FROM addnewclient('Еблан', 'Ебланенко', 'Ебланович', '380432113060');

1. Информация о клиенте

CREATE OR REPLACE FUNCTION infoclient(tel VARCHAR) RETURNS TABLE

("Услуга" VARCHAR, "Дата начала" DATE, "Дата конца" DATE) AS $$

BEGIN

RETURN QUERY

(SELECT specialization\_service, begindate\_abonement, enddate\_abonement FROM abonement

JOIN service ON service\_id=id\_service

JOIN client ON cardnumber\_abonement=card\_number\_client

WHERE mobile\_telephone\_client=$1);

END;

$$ LANGUAGE plpgsql

SELECT \* FROM infoclient('380956435896');

1. Записать клиента на аобнемент

CREATE OR REPLACE FUNCTION recab(tel CHAR, serv VARCHAR, months INT) RETURNS int AS $$

DECLARE

cardn INT;

servid INT;

servab INT;

edate DATE;

r INT;

am INTERVAL;

BEGIN

servid = (SELECT id\_service FROM service WHERE specialization\_service=$2);

cardn = (SELECT card\_number\_client FROM client WHERE mobile\_telephone\_client=$1);

servab = (SELECT id\_ca FROM cost\_ab WHERE durationinmonths=$3 and serv\_id=servid );

r := $3;

am = r || ' months';

edate = now()::date + am;

INSERT INTO abonement (service\_id, cardnumber\_abonement, begindate\_abonement, enddate\_abonement, isactive\_abonement, cost\_abonement)

VALUES (servid, cardn, now()::date, edate, 't', servab);

RETURN 1;

END;

$$ LANGUAGE plpgsql;

SELECT \* FROM recab('380956435896', 'Zumba', 6);

1. Закрыть абонемент

CREATE OR REPLACE FUNCTION closeab(tel CHAR, serv VARCHAR) RETURNS int AS $$

BEGIN

UPDATE abonement

SET enddate\_abonement=now()::date

WHERE id\_abonement=(

SELECT id\_abonement from abonement

JOIN client ON cardnumber\_abonement=card\_number\_client

JOIN service ON service\_id=id\_service

WHERE mobile\_telephone\_client=$1 AND specialization\_service=$2);

RETURN 1;

END;

$$ LANGUAGE plpgsql;

SELECT \* FROM closeab('380956435896', 'Zumba');

1. Расписание тренера

select name\_staff || ' ' || surname\_staff || ' ' || patronomyc\_staff as staff\_fio ,

type\_service, date\_workshift, begin\_time\_workshift, end\_time\_workshift

from staff

join workshift on workshift\_staff = id\_staff

join service on service\_id = id\_service

where id\_staff = 3;

1. Посещение клиента

select name\_client || ' ' || surname\_client || ' ' || patronomyc\_client as client\_fio ,

day\_record, time\_beginrecord, time\_endrecord from client

join public.record on cardnumber\_client\_record = card\_number\_client

where card\_number\_client = 51322