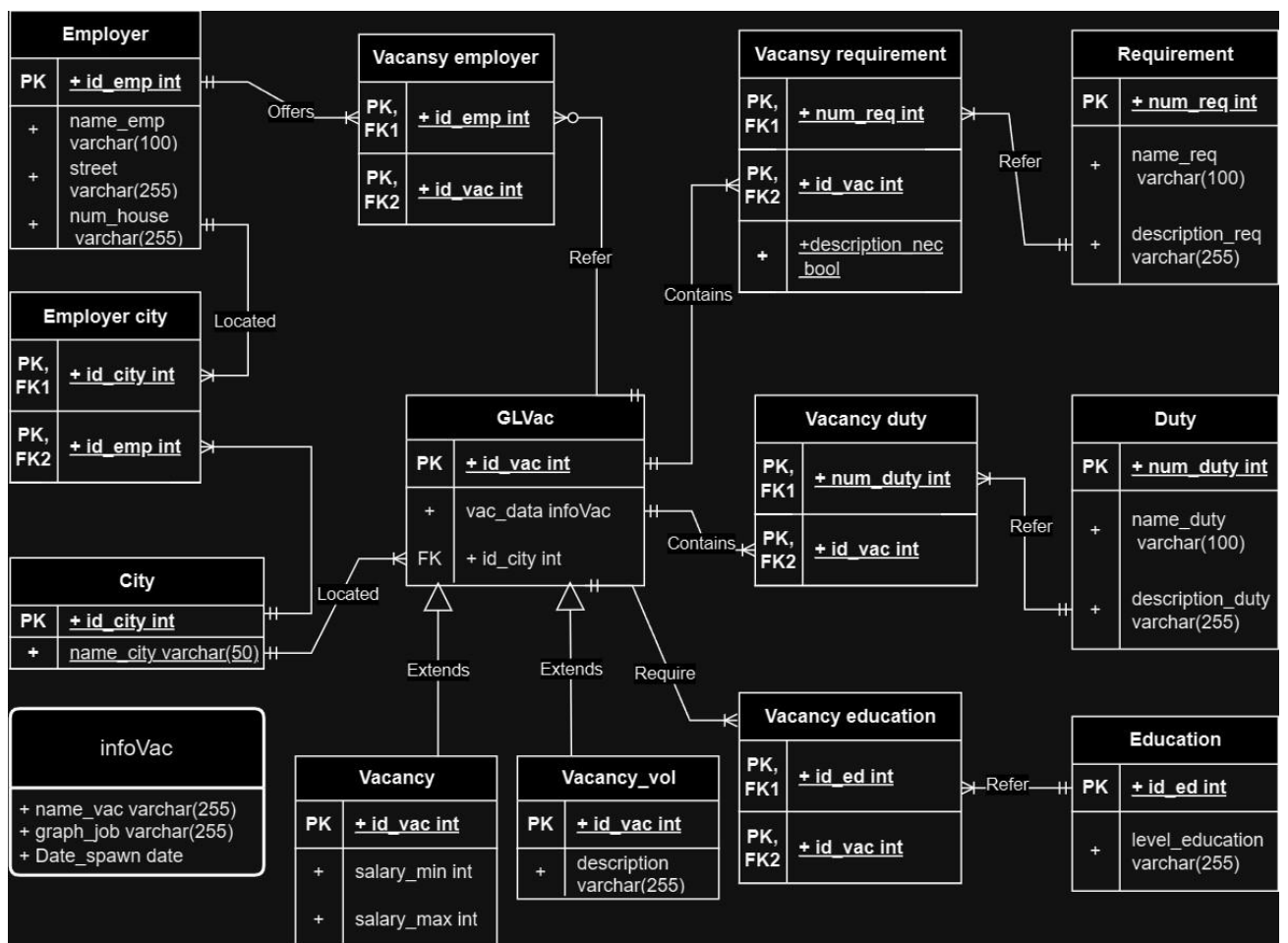


Объектно-реляционные базы данных. Манипуляция данными и пользовательские операторы

1. Вариант задания (12 вариант)

Вакансии: волонтерские позиции, название вакансии, организация работодатель, адрес работодателя, диапазон зарплаты, требования к образованию, Обязанности, график работы, требования обязательные, желательные, дата выставления вакансии.

- а. вакансии, имеющие в названии SQL, но не заканчивающиеся на него
- б. работодатели в Санкт-Петербурге, выставившие несколько вакансий
- в. вакансии с наибольшей зарплатой
- г. волонтерские позиции с максимальным количеством требований
- д. вакансии, в которых нет требования к опыту работы



3. Выполнение заданий

Вставка данных:

```
INSERT INTO City (name_city) VALUES ('Санкт-Петербург');
```

```
INSERT INTO City (name_city) VALUES ('Москва');
```

```
INSERT INTO Vacancy (vac_data, salary_min, salary_max, id_city) VALUES (ROW('frontend junior', '2/2', '2024-04-10'), 40000, 50000, 1);
```

```
INSERT INTO Vacancy (vac_data, salary_min, salary_max, id_city) VALUES (ROW('backend junior', '5/2', '2024-04-11'), 50000, 60000, 2);
```

```
INSERT INTO Vacancy_vol (vac_data, description, id_city) VALUES (ROW('Волонтер на хоккейную арену', '2/2', '2024-04-13'), 'Есть возможность бесплатно смотреть матчи', 2);
```

```
INSERT INTO Vacancy_vol (vac_data, description,id_city) VALUES (ROW('Волонтер на футбольный матч', '5/2', '2024-04-12'),'Обеды обеспечены', 2);
```

```
INSERT INTO Employer (name_emp, street, num_house) VALUES ('Pocket', 'Казанская', '5');
```

```
INSERT INTO Employer (name_emp, street, num_house) VALUES ('Omega', 'Ленина', '7');
```

```
INSERT INTO Employer_city VALUES (1, 1);
```

```
INSERT INTO Employer_city VALUES (2, 2);
```

```
INSERT INTO Employer_city VALUES (2, 3);
```

```
INSERT INTO Vacancy_employer VALUES (1, 1);
```

```
INSERT INTO Vacancy_employer VALUES (1, 2);
```

```
INSERT INTO Vacancy_employer VALUES (2, 3);
```

```
INSERT INTO Vacancy_employer VALUES (2, 4);
```

```
INSERT INTO Requirement (name_req, description_req) VALUES ('Опрятность', 'Необходимо иметь опрятный вид');
```

```
INSERT INTO Requirement (name_req, description_req) VALUES ('Пунктуальность', 'Необходимо всегда приходить вовремя');
```

```
INSERT INTO Requirement (name_req, description_req) VALUES ('Стаж 2г', 'Необходимо иметь стаж в этой области более двух лет');
```

```
INSERT INTO Requirement (name_req, description_req) VALUES ('Хорошая дикция', 'Необходимо обладать хорошей дикцией');
```

```
INSERT INTO Vacancy_requirement VALUES (1, 1, false);
```

```
INSERT INTO Vacancy_requirement VALUES (2, 1, true);
```

```
INSERT INTO Vacancy_requirement VALUES (2, 2, false);
```

```
INSERT INTO Vacancy_requirement VALUES (3, 2, false);
```

```
INSERT INTO Vacancy_requirement VALUES (1, 3, false);
```

```
INSERT INTO Vacancy_requirement VALUES (4, 4, false);
```

```
INSERT INTO Duty (name_duty, description_duty) VALUES ('Обеспечение it  
жизнедеятельности', 'Обеспечение стабильного и безотказного функционирования ИТ-  
инфраструктуры');
```

```
INSERT INTO Duty (name_duty, description_duty) VALUES ('Водить автомобиль', 'Управлять  
автомобилем категории B');
```

```
INSERT INTO Duty (name_duty, description_duty) VALUES ('Помощь людям', 'Необходимо  
отвечать на вопросы и всячески помогать людям');
```

```
INSERT INTO Vacancy_duty VALUES (1, 1);
```

```
INSERT INTO Vacancy_duty VALUES (1, 2);
```

```
INSERT INTO Vacancy_duty VALUES (2, 3);
```

```
INSERT INTO Vacancy_duty VALUES (3, 2);
```

```
INSERT INTO Education (level_education) VALUES ('Без образования');
```

```
INSERT INTO Education (level_education) VALUES ('Основное общее образование');
```

```
INSERT INTO Vacancy_education VALUES (2, 1);
```

```
INSERT INTO Vacancy_education VALUES (2, 2);
```

```
INSERT INTO Vacancy_education VALUES (1, 3);
```

```
INSERT INTO Vacancy_education VALUES (1, 4);
```

Запрос выборки с условием к таблицам и предку и потомку:

```
SELECT * FROM GLVac left join City on GLVac.id_city = City.id_city
```

WHERE City.name_city = 'Москва';

	id_vac integer	vac_data infovac	id_city integer	id_city integer	name_city character varying (50)
1	2	("backend junior",5/2,2024-04-11)	2	2	Москва
2	3	("Волонтер на хоккейную арену",2/2,2024-04-13)	2	2	Москва
3	4	("Волонтер на футбольный матч",5/2,2024-04-12)	2	2	Москва

Запрос выборки с условием только к таблице предку:

SELECT * FROM only GLVac left join City on GLVac.id_city = City.id_city

WHERE City.name_city = 'Москва';

	id_vac integer	vac_data infovac	id_city integer	id_city integer	name_city character varying (50)

Запрос выборки с условием только к таблице потомку:

SELECT * FROM Vacancy left join City on Vacancy.id_city = City.id_city

WHERE City.name_city = 'Москва';

	id_vac integer	vac_data infovac	id_city integer	salary_min integer	salary_max integer	id_city integer	name_city character varying (50)
1	2	("backend junior",5/2,2024-04-11)	2	50000	60000	2	Москва

Пользовательский оператор:

CREATE OR REPLACE FUNCTION vac_graph_eq(infoVac, infoVac)

RETURNS BOOLEAN LANGUAGE SQL IMMUTABLE

AS \$\$

SELECT \$1.graph_job = \$2.graph_job;

\$\$;

CREATE OPERATOR %= (

PROCEDURE = vac_graph_eq,

LEFTARG = infoVac,

RIGHTARG = infoVac

);

SELECT *,vac.vac_data %=(SELECT vac_data FROM GLVac AS another WHERE
another.id_vac = 1) as graphEq FROM GLVac AS vac

	id_vac [PK] integer	vac_data infovac	id_city integer	grapheq boolean
1	1	("frontend junior",2/2,2024-04-10)	1	true
2	2	("backend junior",5/2,2024-04-11)	2	false
3	3	("Волонтер на хоккейную арену",2/2,2024-04-13)	2	true
4	4	("Волонтер на футбольный матч",5/2,2024-04-12)	2	false

Агрегатная функция:

CREATE OR REPLACE FUNCTION data_max(ac1 infoVac ,ac2 infoVac)

RETURNS infoVac LANGUAGE plpgsql

AS

\$\$begin

IF ac1.date_spawn IS NULL

THEN

RETURN ac2;

ELSEIF ac2.date_spawn IS NULL

THEN RETURN ac1;

ELSEIF ac1.date_spawn>=ac2.date_spawn

THEN RETURN ac1;

else RETURN ac2;

END IF;

```
end; $$
```

```
IMMUTABLE RETURNS NULL ON NULL INPUT;
```

```
CREATE AGGREGATE max (infoVac)
```


```
(
```

```
  sfunc = data_max,
```

```
  stype = infoVac
```

```
);
```

```
SELECT max(GLVac.vac_data) from GLVac group by id_vac;
```

	max infovac 
1	("Волонтер на хоккейную арену",2/2,2024-04-13)
2	("Волонтер на футбольный матч",5/2,2024-04-12)
3	("backend junior",5/2,2024-04-11)
4	("frontend junior",2/2,2024-04-10)