Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования

«Московский технический университет связи и информатики»

(МТУСИ)

Кафедра «Программная инженерия»

дисциплина «Рефакторинг баз данных и приложений»

Лабораторная работа № 5

**«Рефакторинг приложений с целью оптимизации запросов и**

**индексов базы данных»**

Выполнил: студент группы БПИ2303

Берездовец Илья Сергеевич

Москва 2025

**Цели лабораторной работы**

1. Изучить основные принципы оптимизации SQL-запросов.

2. Научиться выявлять узкие места в работе базы данных.

3. Освоить создание, модификацию и удаление индексов для ускорения запросов.

4. Применить знания на практике для оптимизации базы данных и приложения.

**Ход работы:**

Изначальный код:

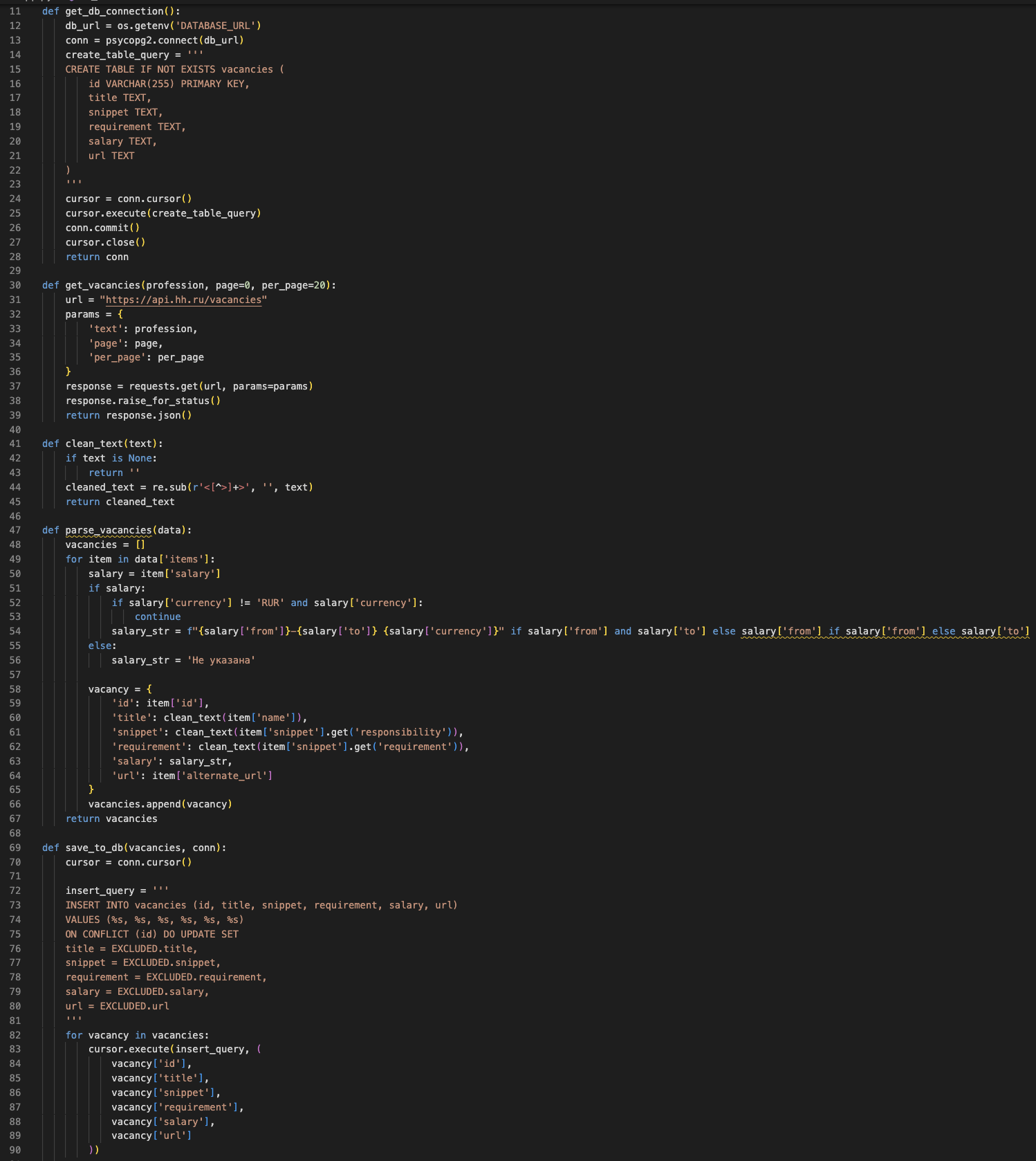


Рис. 1 – (первая часть изначального кода)

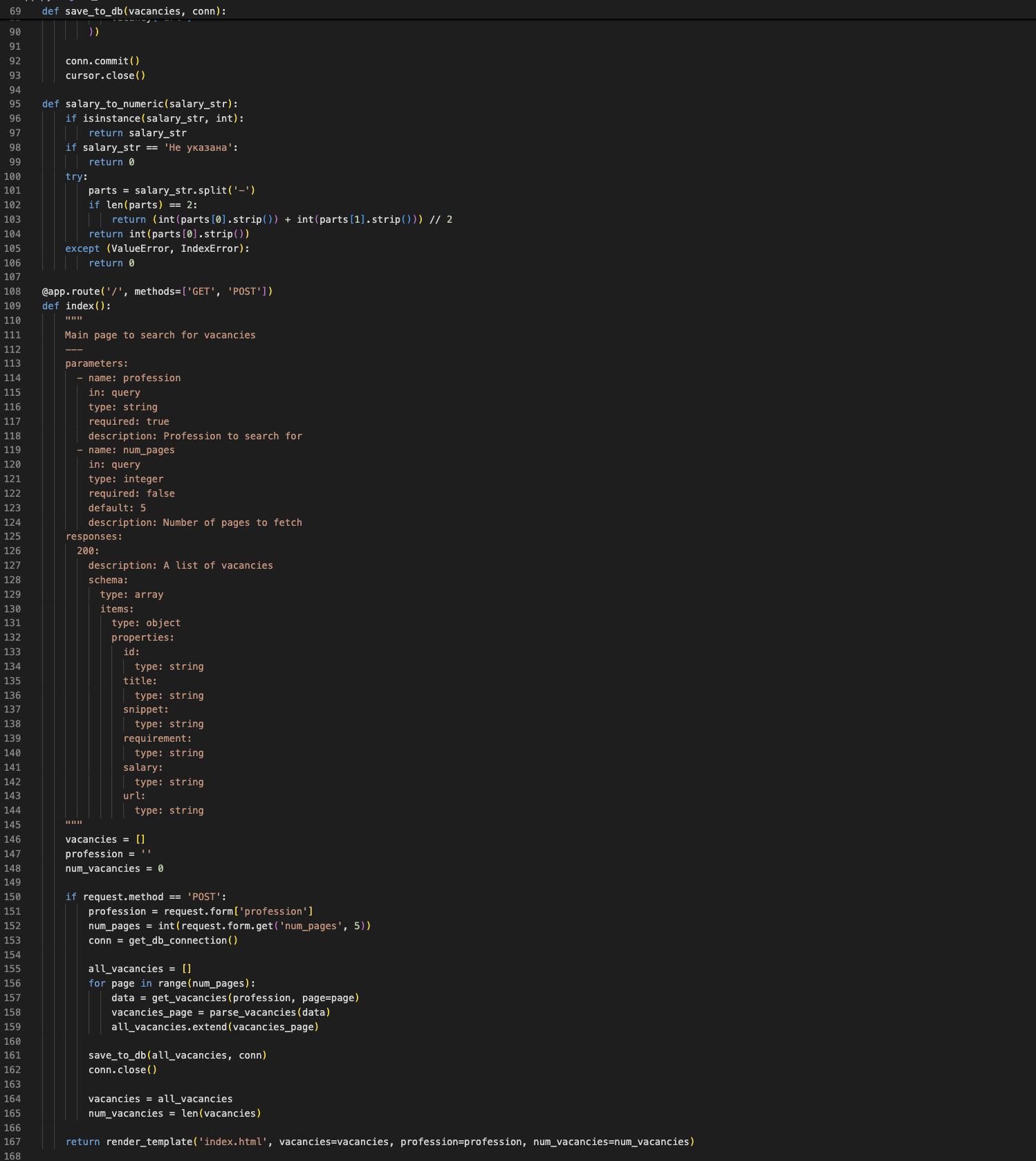


Рис. 2 – (вторая часть изначального кода)



Рис. 3 – (третья часть изначального кода)

Применю следующие улучшения коду, а именно SQL – запросам.

1. Индексы: добавлю индекс на title и salary — часто используются в выборке и группировке.

2. Уберу SELECT \*: заменю выборки на конкретные столбцы.

3. Оптимизирую SQL-запросы: исправлю выражения с substring, чтобы избежать ненужных операций.

4. Добавлю анализ через EXPLAIN.

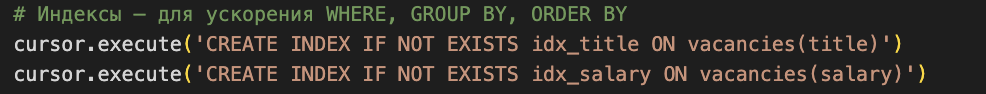


Рис. 4 - (Добавил индексы)ё

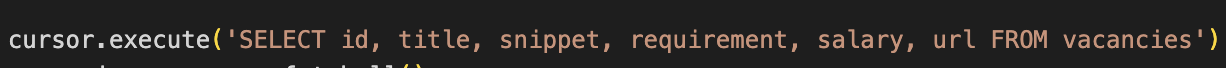
2. 

Рис. 5 – (Добавил выборку на конкретные столбцы)

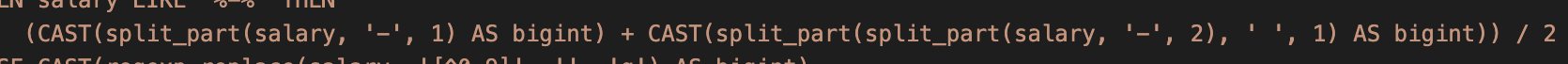
3. 

Рис. 6 – (Исправил substring на split\_part)

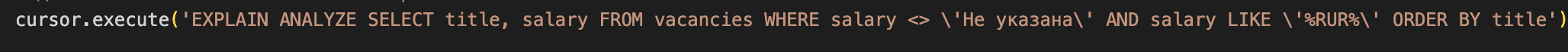
4. 

Рис. 7 – (Добавил анализ через EXPLAIN)

Резюмирую улучшения:  
Вот кратко от твоего лица:

1. Добавил индексы на `title` и `salary`, так как они часто участвуют в фильтрации и группировке — это ускоряет запросы.

2. Заменил `SELECT \*` на выбор конкретных столбцов — уменьшает нагрузку и повышает читаемость.

3. Оптимизировал SQL-запросы, переписал `substring`-выражения, чтобы убрать лишние операции.

4. Добавил `EXPLAIN ANALYZE` для анализа запросов и оценки их производительности.

Вывод:  
1. Изучил основные принципы оптимизации SQL-запросов.

2. Научился выявлять узкие места в работе базы данных.

3. Освоил создание, модификацию и удаление индексов для ускорения запросов.

4. Применил знания на практике для оптимизации базы данных и приложения.

Ссылка на GitHub: https://github.com/iberezaa/RBDiP