

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

"Российский технологический университет"

МИРЭА

Институт «Кибербезопасности и цифровых технологий»

Кафедра КБ-3 «Разработка программных решений и системного программирования»

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 3**

**по дисциплине**

**«Проектирование Архитектуры программного обеспечения»**

**Выполнил**

студент группы БСБО-02-20 Смелкин Н.Д.

*ФИО*

**Принял**

к.т.н., доцент Волович М. Е.

**Работа выполнена**

« » декабря 2023 г.

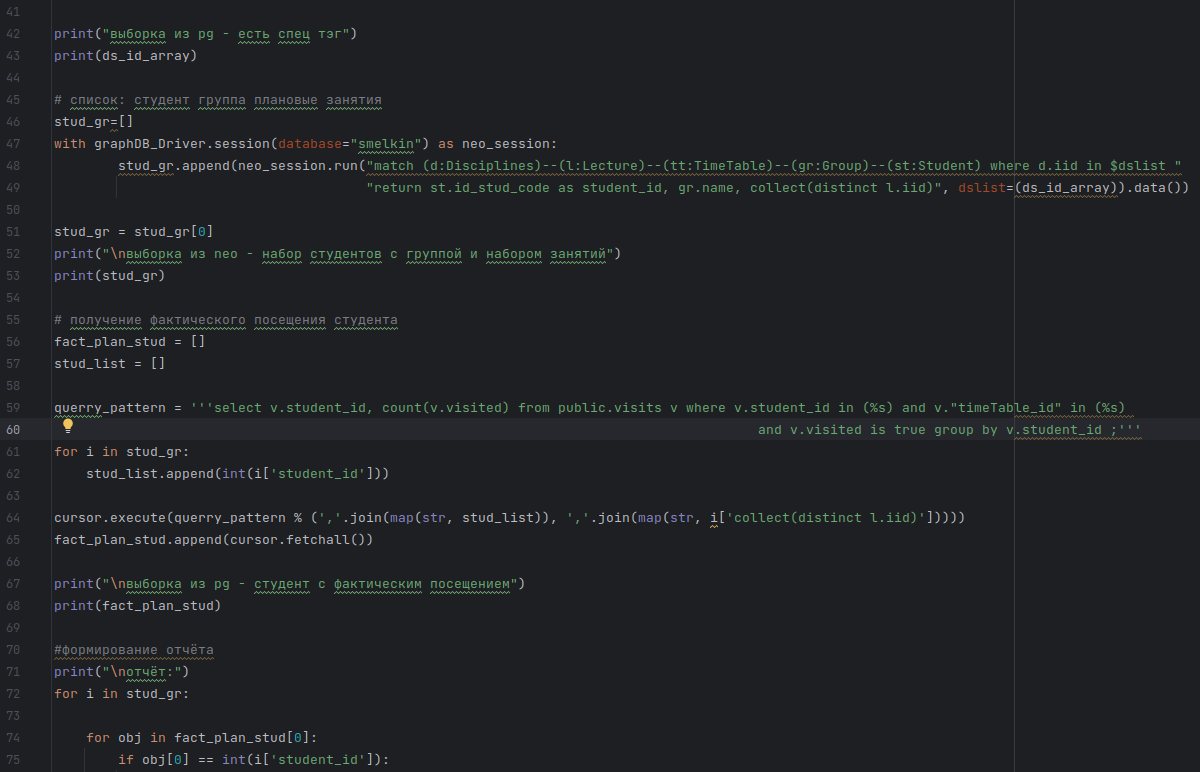
Москва 2023

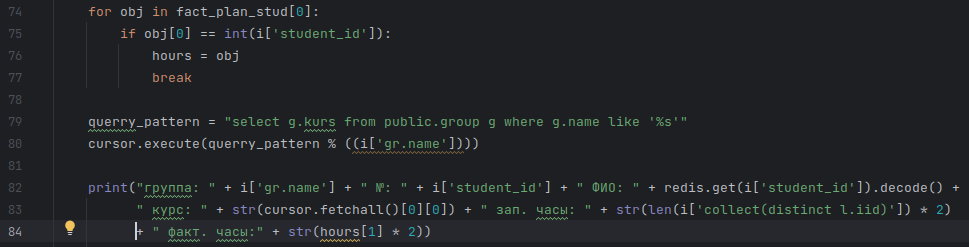
Задание:

Выполнить запрос к структуре хранения информации о группах учащихся, курсах обучения, лекционной программе и составу лекционных курсов и практических занятий, а также структуре связей между курсами, специальностями, студентами кафедры и данными о посещении студентами занятий, для извлечения отчета по заданной группе учащихся с указанием объема прослушанных часов лекций а также необходимого объема запланированных часов, в рамках всех курсов для каждого студента группы. Предполагается, что одна лекция равна 2-м академическим часам. В отчет должны попасть только лекции, которые содержат тег специальной дисциплины кафедры. В качестве результата необходимо вывести полную информацию о группе, студенте, курсе, количестве запланированных часов и посещенных часов занятий.

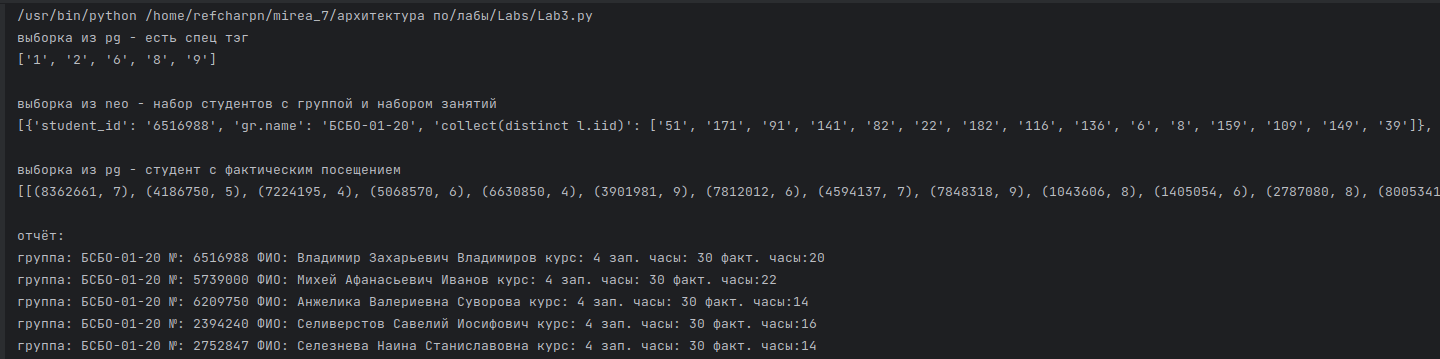
Код:







Результат работы(отчёт):



Обоснование структуры:

Для начала в программе инициализируются объекты для подключения к базам данных. Далее из PostgreSQL выбираются все дисциплины со специальным тегом. Другие СУБД не подходят из-за неэффективности:

* Neo4j — нет связанных данных
* ElasticSearch — нет полнотекстового поиска, плюс этих данных нет.
* Redis — нет поиска ключ:значение, плюс этих данных нет
* MongoDB — нет ни вложенных данных

После получения нужных дисциплин необходимо найти студентов, занятия(по сути — плановое посещение). Для этого используется Neo4j, т. к. данные связаны: (d:Disciplines)—(l:Lecture)—(tt:TimeTable)--(gr:Group)--(st:Student). Из такой связи дополнительно была взяты группы для каждого студента. Как только были получены эти данные можно выбрать фактическое посещение студента. Использовалась СУБД PostgreSQL(причины как и в пункте 1). По сути, фактическое посещение — занятия где отметка - true. Все данные получены, можно составлять отчёт. С помощью Redis студенческие номера преобразовались в ФИО студента. Данные вида: Группа Студбилет ФИО курс плановые часы фактические часы выводятся построчно для каждого студента.