Университет ИТМО

Факультет Инфокоммуникационных технологий

Направление подготовки «09.03.03 Мобильные и сетевые технологии»

Лабораторная работа №3

### **СОЗДАНИЕ БАЗЫ ДАННЫХ POSTGRESQL СРЕДСТВАМИ PGADMIN**

Выполнила:

Егоров М. П.

Группа К3240

Преподаватель:

Говорова М. М.

Санкт-Петербург

2020

**Цель работы**:

Овладеть практическими навыками создания таблиц базы данных PostgreSQL 1Х, заполнения их рабочими данными, резервного копирования и восстановления БД.

**Практическое задание:**

1. Создать базу данных с использованием pgAdmin 4 (согласно индивидуальному заданию).
2. Создать схему в составе базы данных.
3. Создать таблицы базы данных.
4. Установить ограничения на данные: *Primary Key, Unique, Check, Foreign Key*.
5. Заполнить таблицы БД рабочими данными.
6. Создать резервную копию БД.

*Указание:*

*Создать две резервные копии:*

* *с расширением CUSTOM для восстановления БД;*
* *с расширением PLAIN для листинга (в отчете);*
* *при создании резервных копий БД настроить параметры Dump options для Type of objects**и**Queries .*

1. Восстановить БД.

**Выполнение:**

1. **Название создаваемой БД**

«LotosLab»

1. **Django ORM**

Определим схемы через ORM фреймворка Python – Django. Нужно определить название таблицы и ее атрибуты, а также процедуры. В языке Python это соответственно название класса, название и типы атрибутов (django.fields), методы класса.

К примеру, сущность «Партнер».

Атрибуты: «фио», «специализация», «специализация на английском», «фио на английском», «описание».

Text

Description automatically generated

Рисунок 1. Определение названия и атрибутов сущности «Партнер»

Процедура: получить партнеров из определенной специализации. Соответствует запросу «SELECT \* FROM Partner WHERE specialization\_en = spec»:

Text

Description automatically generated with medium confidence

Рисунок 2. Определение метода сущности «Партнер»

Остальные сущности находятся в файле «models.py».

1. **Миграции базы данных и выгрузка базы данных**

После того, как все сущности были определены на языке Python нужно сделать миграции, чтобы модели были реализованы в PostgreSQL.

После миграции нужно добавить данные из csv файлов: списки анализов, списки магазинов и партнеров:

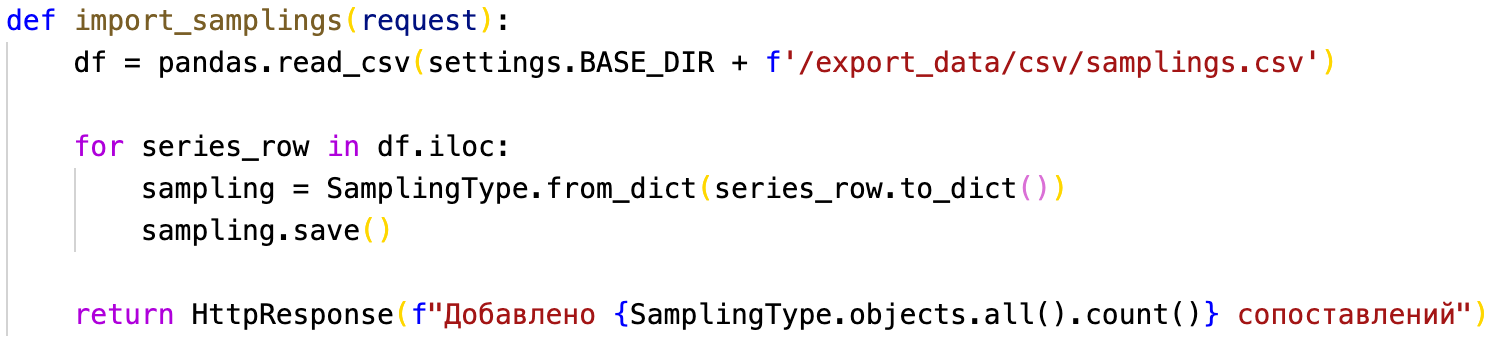


Рисунок 3. Импорт списка биоматериалов из файла csv.

Как только данные загрузятся, можно сделать выгрузку данных с помощью инструментов Python.

**Выводы:**

Была реализована инфологическая модель «LotosLab» в реляционную базу данных PostgreSQL с помощью фреймворка Django. Были загружены данные из csv файлов и сделана выгрузка всей базы для восстановления.