aa

**1. Написать простое приложение на Django**

В этом задании огромное значение имеет нейминг, декомпозиция, читаемость и логика кода.

Главное — избежать хардкодинга, что будет легче сделать, если регулярно задаваться вопросом: «что, если я буду таким же образом работать не с 3, а со 100 объектами?»

**1.1** Создать новый project, в нем — новый app.

**1.2** Создать модели (все поля — django.models.CharField):

• модель ModelA с полями a\_one, a\_two, a\_three;

• модель ModelB с полями b\_one, b\_two, b\_three;

• модель ModelC с полями c\_one, c\_two, c\_three.

Создать и провести миграции.

**1.3** Создать страницу, на которой в виде html-таблицы выводятся все строки выбранной модели.

Вверху страницы должно быть название модели, из которой выводятся строки.

В шапке таблицы должны стоять названия столбцов.

Порядок вывода столбцов для моделей:

• ModelA — one, two, three

• ModelB — two, three, one

• ModelC — three, one, two

По умолчанию выводятся строки из ModelA.

**1.4** Добавить в html-файл «переключатель» (любого формата) с возможностью выбора другой модели.

После выбора другой модели страница должна быть перезагружена, а ее содержимое — обновлено в соответствии с выбранной моделью.

**1.5** Добавить строку, содержащую пустые input-боксы и кнопку, при нажатии на которую отправляется POST запрос, который приводит к добавлению строки в модель, после чего страница перезагружается.

**1.6** Для всех компонентов страницы применить bootstrap классы.

**2. Написать декоратор на Python**

Декорируемая функция (ее нельзя изменять) с декоратором:

@print\_num\_gt(3) def print\_num(x: int):   print(x)

При вызове функции `print\_num` в консоль должен быть выведен аргумент `x`, если он больше параметра, передаваемого в `print\_num\_gt`, иначе — «error».

**3. Написать запросы к приведенным таблицам на PostgreSQL**

create table department (  id integer primary key,  name varchar );  create table employee (  id integer primary key,  department\_id integer references department(id),  chief\_id integer references employee(id),  name varchar,  salary numeric );

A. Вывести список сотрудников, получающих заработную плату большую чем у непосредственного руководителя

B. Вывести список сотрудников, получающих максимальную заработную плату в своем отделе

C. Вывести список ID отделов, количество сотрудников в которых не превышает 3 человек

D. Вывести список сотрудников, не имеющих назначенного непосредственного руководителя, работающего в том-же отделе

E. Вывести список ID отделов с максимальной суммарной зарплатой сотрудников