1. Найти в проекте повторяющиеся вызовы методов для одного экземпляра модели и применить к ним декоратор «@cached\_property». Оценить, насколько уменьшилось число дублей при выполнении SQL-запросов и каков прирост производительности.

Переделал функции подсчета количества и сумм, в заказе, выигрыш микроскопический) так же как в лекции, один запрос экономится. Но может я что-то делаю не так.

2. Применить тег «with» в одном из шаблонов. Оценить, насколько уменьшилось число дублей при выполнении SQL-запросов и каков прирост производительности.

Добавил метод «get\_summary» в класс заказов, и использовал его в шаблоне «inc\_order\_summary.html» увы это не принесло ожидаемого результата, думаю потому, что значения используются один раз и прибавляется время на создание кэша.

3. Установить и настроить приложение «Memcached». Реализовать кеширование на низком уровне для функций, возвращающих редко изменяющиеся данные (продукты каталога, список категорий и т.д.). Оценить прирост производительности.

Sudo apt install memcached

Sudo apt install libmemcached-dev

source /home/django/venv/bin/activate

/etc/memcached.conf

-m 64 -> -m 256

systemctl restart memcached.service