### 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. (Добавил в корзину 4 товара)

*До кэширования корзины:*

|  |  |  |
| --- | --- | --- |
| **Адрес** | **Число запросов** | **Число дублей** |
| '/users/profile/' | 12 | 4 |
| **Параметр** | **-r20 –c10** | |
| Число переходов | 2630 | |
| Доступность сервера | 100% | |
| Время теста, с | 53,34 | |
| Время отклика, с | 0,20 | |
| Запросов в секунду | 49,31 | |
| Пропускная способность,  МБ/сек | 1,57 | |
| Согласованность | 9,81 | |
| Удачных транзакций | 2630 | |
| Неудачных транзакций | 0 | |
| Самая долгая транзакция, с | 4.24 | |

*После Кэширования корзины:*

|  |  |  |
| --- | --- | --- |
| **Адрес** | **Число запросов** | **Число дублей** |
| '/users/profile/' | 7 | 4 |
| **Параметр** | **-r20 –c10** | |
| Число переходов | 2630 | |
| Доступность сервера | 100% | |
| Время теста, с | 53,67 | |
| Время отклика, с | 0,20 | |
| Запросов в секунду | 49,00 | |
| Пропускная способность,  МБ/сек | 1,56 | |
| Согласованность | 9,84 | |
| Удачных транзакций | 2630 | |
| Неудачных транзакций | 0 | |
| Самая долгая транзакция, с | 4,22 | |

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

*До кэширования окна создания заказа (в корзине 3 товара):*

|  |  |  |
| --- | --- | --- |
| **Адрес** | **Число запросов** | **Число дублей** |
| ' /orders/create/' | 30 | 27 |
| **Параметр** | **-r20 –c10** | |
| Число переходов | 2230 | |
| Доступность сервера | 100% | |
| Время теста, с | 174,78 | |
| Время отклика, с | 0,77 | |
| Запросов в секунду | 12,76 | |
| Пропускная способность,  МБ/сек | 0,52 | |
| Согласованность | 9,81 | |
| Удачных транзакций | 2230 | |
| Неудачных транзакций | 0 | |
| Самая долгая транзакция, с | 10.02 | |

*После кэширования окна создания заказа (в корзине 3 товара):*

|  |  |  |
| --- | --- | --- |
| **Адрес** | **Число запросов** | **Число дублей** |
| ' /orders/create/' | 28 | 25 |
| **Параметр** | **-r20 –c10** | |
| Число переходов | 2230 | |
| Доступность сервера | 100% | |
| Время теста, с | 158,42 | |
| Время отклика, с | 0,7 | |
| Запросов в секунду | 14,08 | |
| Пропускная способность,  МБ/сек | 0,57 | |
| Согласованность | 9,81 | |
| Удачных транзакций | 2230 | |
| Неудачных транзакций | 0 | |
| Самая долгая транзакция, с | 8,9 | |

### 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. (siege -i -f /home/django/geekshop/urls.txt -d0 -r17 -c100)

*До кэширования окна каталога:*

|  |  |
| --- | --- |
| **Параметр** | **-r17 –c125** |
| Число переходов | 2230 |
| Доступность сервера | 100% |
| Время теста, с | 174,78 |
| Время отклика, с | 0,77 |
| Запросов в секунду | 12,76 |
| Пропускная способность,  МБ/сек | 0,52 |
| Согласованность | 9,81 |
| Удачных транзакций | 2230 |
| Неудачных транзакций | 0 |
| Самая долгая транзакция, с | 10.02 |

*После кэширования окна каталога:*

|  |  |
| --- | --- |
| **Параметр** | **-r20 –c10** |
| Число переходов | 2230 |
| Доступность сервера | 100% |
| Время теста, с | 158,42 |
| Время отклика, с | 0,7 |
| Запросов в секунду | 14,08 |
| Пропускная способность,  МБ/сек | 0,57 |
| Согласованность | 9,81 |
| Удачных транзакций | 2230 |
| Неудачных транзакций | 0 |
| Самая долгая транзакция, с | 8,9 |

### 4. Реализовать кеширование в шаблоне для набора форм. Оценить эффект.

### *До кэширования страницы редактирования заказа (4 товара)*

|  |  |  |  |
| --- | --- | --- | --- |
| **Адрес** | **Число запросов** | **Число дублей** | **Время выполнения, мс** |
| ' /orders/update/' | 38 | 28 | 18,43 |

### *После кэширования страницы редактирования заказа (4 товара)*

|  |  |  |  |
| --- | --- | --- | --- |
| **Адрес** | **Число запросов** | **Число дублей** | **Время выполнения, мс** |
| ' /orders/update/' | 10 | 0 | 8,00 |

### 5. \*Реализовать работу с некоторыми пунктами меню через AJAX и кешировать соответствующие страницы. Оценить эффект от применения технологии AJAX и эффект от кеширования.

### *Поработаю с аяксом позже, т.к. очень отстаю от программы*

### 6. Попробовать реализовать кеширование всего сайта в проекте. Оценить прирост производительности и возникающие при этом проблемы с обновлением контента.

### *Закешировал весь сайт, установил never\_cache на контроллер /products/*