

Celery + Redis/RabbitMQ Django

№ урока: 9 **Курс:** Django Essential

Средства обучения: Pycharm Community, GIT, аккаунт в GitHub, Python 3.6 и выше, Django проект, Docker образ.

Обзор, цель и назначение урока

Научиться подключать к проекту Celery, Redis а также RabbitMQ. Научиться создавать задания (task) и работников (workers). Получить знание о данных технологиях, их преимуществах, где они используются и для чего. Рассмотреть запуск задач по расписанию, а также рассмотреть сессии пользователей.

Изучив материал данного занятия, учащийся сможет:

- Подключать к проекту Celery.
- Подключать к проекту RabbitMQ.
- Подключать к проекту Redis.
- Создавать Celery задания.
- Создавать Celery Workers.
- Создавать задания по расписанию.
- Создавать сессии.

Содержание урока

1. Краткий обзор всех инструментов и для чего они служат
2. Установка Celery
3. Установка Erlang
4. Установка RabbitMQ
5. Установка Redis
6. Подключение всех вышеперечисленных инструментов к проекту
7. Запуска серверов RabbitMQ и Redis
8. Создание celery.py
9. Создание первой задачи
10. Создание работников
11. Создание задачи по расписанию
12. Вызов задачи из WSGI
13. Создание сессии

Резюме

- **Celery** — это инструмент, который позволяет решать проблемы с большим количеством данных. Так как Django имеет ограниченное количество пользователей, которым он может отвечать на запрос, используется Celery, который позволяет увеличить это количество, и уменьшить время ожидания ответа пользователем.
- Страницы с аналитикой, отчеты, видео, картинки, много данных — это основные случаи, когда нужно использовать Celery.
- В качестве брокеров для Celery можно использовать Redis и RabbitMQ.

- **RabbitMQ** — это приложение для работы с очередями сообщений. Это инструмент, который определяет очередь, к которой могут подключаться различные приложения и получать или передавать сообщение (данные). Используется для уменьшения времени отклика веб-приложения. Он служит прослойкой между несколькими серверами.
- **Redis** — это высокопроизводительная база данных, которая предоставляет доступ к данным по ключу доступа и хранит данные в памяти. Redis можно использовать как:
 - База данных - очень быстрая, нет реляций, нет разделения на группы.
 - Кеш - не более 512мб, пользование разными сервисами, нужно очищать.
 - Брокер сообщений - отсутствие очередей, модель "издатель - подписчик", стабильная работа.
- Для установки Celery используем команду `pip install Celery`.
- Для запуска RabbitMQ нужно установить язык Эрланг и сервер rabbitmq.
- Для запуска сервера rabbitmq используем на линуксе команды:
 - `systemctl enable rabbitmq-server`
 - `systemctl start rabbitmq-server`
 - `systemctl status rabbitmq-server` - проверить статус
- Для установки Redis используем или Docker, или нужно скачать с сайта архив и, с использованием следующих команд, установить и запустить сервер:
 - `wget http://download.redis.io/releases/redis-5.0.7.tar.gz`
 - `tar xzf redis-5.0.7.tar.gz`
 - `cd redis-5.0.7`
 - `make`
 - `src/redis-server` - запуск сервера
 - `redis-cli ping` - проверить статус
- Чтобы подключить Celery к проекту, используем файл с настройками. Также нужно создать файл `celery.py` и конфигурационный файл. С помощью конфигураций можно менять брокера.
- В файле `celery.py` нужно определить окружение (`os.environ.setdefault`), создать приложение с помощью Celery объекта.
- Чтобы при каждом запуске запускался Celery, нужно добавить в файл `__init__.py` celery приложение.
- Для создания задач используем `tasks.py`.
- Для запуска задач в фоновом режиме нужно использовать `delay()`.
- Для запуска работника используем `celery -A "название сайта" worker -l info`.
- Чтобы запустить задачу по расписанию нужно настроить `CELERYBEAT_SCHEDULE` в файле с настройками. Необходимо задать время запуска и название задачи.
- Для вызова WSGI необходимо реализовать следующую структуру: проверку на время, если лимит превышен - нужно запускать задачу.
- Для настройки сессий используем файл с настройками.

Закрепление материала

- Для чего используют Celery + Redis/RabbitMQ?
- Что такое Redis?
- Что такое Worker в Celery?
- Что значит вызов задачи из WSGI?
- Какие отличия Redis от RabbitMQ?

Дополнительное задание

Задание

Установить, настроить и подключить к проекту Celery + Redis/RabbitMQ, создать worker.

Самостоятельная деятельность учащегося

Задание 1

Выучить и понять какие есть преимущества от использования Celery + Redis/RabbitMQ.

Задание 2

Создать задание по расписанию. Оно должно подключаться каждый час к API с погодой, считывать температуру и сохранять в модель.

Задание 3

Ознакомиться с тем, какие есть аналоги Celery, Redis и RabbitMQ.

Рекомендуемые ресурсы

Документация:

- <https://dramatiq.io/>
- <https://docs.celeryproject.org/en/latest/index.html>
- <https://www.rabbitmq.com/>
- <https://redis.io/>
- <https://docs.djangoproject.com/en/2.2/topics/http/sessions/>