Название проекта – Plagiat\_Notion\_2.0

Автор проекта: Парфёнова Лидия

Описание идеи: Telegram bot, написанный на библиотеке aiogram главная суть, чтобы пользователь мог записать важные дела и придать им описание и другие параметры, не скачивая никакое приложение, а пользуясь только telegram. Также есть функции: создание изображение, присылаемое ботом, на основе название профессии, отметка метки на карте, получение информации о погоде, информация о репозиториях и активностей в них какого-то пользователя

Описание реализации:

Бот написан не на классах, так как ботостроение не предлагает использование классов из-за асинхронности и особенности библиотек по ботам. Классов в проекте нет за исключение классов для ORM-моделей.

Проект разделен по папкам.

--API

----Github

----Hh

----Weather

----Maps

--Bot\_logging

--common

--database

--handlers

Основной файл app.py

В основном файле app.py, через который происходит запуск бота (версия от 15.04, пока бот не залит на хостинг), а роутеры, отвечающие за хэндлеры каждый импортирует в главный файл. Всеми роутерами руководит диспечер. У бота есть определенный набор команд, с которыми он умеет взаимодействовать с пользователем, эти команды хранятся в папке common файл bot\_cmd\_list.py. При запуске асинхронной функции main() в файле сразу создаются все таблицы базы данной, удаляются все попытки взаимодействия с ботом, когда бот был недоступен (не включен)

Database. Особенности реализации

База данных для хранения информация о задачах пользователя, написанный в на orm-моделях, используя самый популярный для этого язык sqlaclhemy

В папке database есть файл models.py. В котором создаётся движок и «делатель асинхронный сессий (доступ к базе данных)»

Стоит отметить, что sqlachemy посторена на декларативном подходе. Удобство в **декларативном** **подходе** в том, что вся метаинформация и дополнительные атрибуты, такие как связи с другими классами, также объявляются встроенными в определение класса

Есть абстрактный класс SqlaclhemyBase, который наследуется от AsyncAttrs и DeclarativeBase

От абстрактного классе наследуется класс User