Stack Overflow на русском — это сайт вопросов и ответов для программистов. Присоединяйтесь! Регистрация займёт не больше минуты.
Присоединиться к сообществу
Любой может задать вопрос
Любой может ответить
Лучшие ответы получают голоса и поднимаются наверх

Проблема с asyncio: RuntimeWarning: coroutine 'message_send' was never awaited self.tk.mainloop(n)

Задать вопрос

Вопрос задан 2 года 5 месяцев назад Последняя активность 2 года 5 месяцев назад Просмотрен 6k раз

stackoverflow на русском
Лучше спросить и побыть дураком пять минут, чем промолчать и остаться им на всю жизнь!
andreycha
Задать вопрос »

Важное на Мете
Hot Meta Posts: Allow for removal by moderators, and thoughts about Italy…
Goodbye, Prettify. Hello highlight.js! Swapping out our Syntax Highlighter
Итоги выборов модераторов 2020

1

Решил написать Discord бота, но вышла проблема с библиотекой `asyncio`, которая выдаёт ошибку:

> RuntimeWarning: coroutine 'message_send' was never awaited self.tk.mainloop(n)

Сама ошибка возникает при вызове функции `message_send()`, которая активируется после нажатия кнопки в самом клиенте.

```
from discord.ext import commands
from discord import Embed
import asyncio
import tkinter
from tkinter import messagebox
import tkinter.ttk

client_token = "NDE3MzQ4MjMyNDQ4OTY2NjY2.DX8JdA.rNz3HTnAIVie0MBliVewPUPRqus"
bot_prefix = "!"
Client = discord.Client()
client = commands.Bot(command_prefix=bot_prefix)

@client.event
async def on_ready():
    print("ready")

    async def message_send(): # При вызове этой функции всё ломается
        message = entry_message.get()
        channel = combobox.get()
        channels = {"#channel": "394132322372419597", "#glob": "359289092917166080
        await client.send_message(discord.Object(id=channels[channel]), message)
    root = tkinter.Tk()
    root.title("Bot client")
    root.geometry("500x400")
    root.resizable(False, False)
    entry_message = tkinter.Entry(root)
    entry_message.place(x=40, y=100, width=200, height=20)
    button_send_message = tkinter.Button(root, text='Отправить', command=message_s
    button_send_message.place(x=140, y=140, width=100, height=25)
    combobox = tkinter.ttk.Combobox(root, values=[u"#test", u"#glob"], height=2, s
    combobox.place(x=40, y=140, width=80)
    root.mainloop()

client.run(client_token)
```

Заранее спасибо.

python tkinter asyncio discord-api

поделиться улучшить этот вопрос отслеживать изменён 25 мар '18 в 20:15

задан 25 мар '18 в 17:36
Sanic
31 • 1 • 8

Ваш же вопрос Проблема с синхронизацией python – Twiss 25 мар '18 в 17:41

Да, но по скольку я не могу удалить его или отредактировать, пришлось создать новый. Извиняюсь сразу за тот вопрос, подскажите, как удалить его, если знаете. – Sanic 25 мар '18 в 17:43

создайте *минимальный* пример кода (ошибку в несколько строк можно воспроизвести: одна строка `loop.run_until_complete(root.mainloop())` плюс строки чтоб loop, root создать). Не нужно весь код без разбора в вопрос добавлять. минимальный,воспроизводимый,пример Нажмите править. – jfs 25 мар '18 в 17:53

добавить комментарий

1 ответ

Текущие | По дате публикации | Голоса

4

Чтобы воспроизвести ошибку, ваш случай сводится к следующим строкам:

```
#!/usr/bin/env python3
import asyncio
from tkinter import Tk, Button


async def main():  # XXX BROKEN: DO NOT USE!!!
    async def coro():
        pass

    root = Tk()
    Button(text='press me', command=coro).pack()
    root.mainloop()
asyncio.get_event_loop().run_until_complete(main())
```

Если нажать на кнопку, то выводится предупреждение:

```
.../lib/python3.6/tkinter/__init__.py:1277: RuntimeWarning: coroutine 'coro' was ne
  self.tk.mainloop(n)
```

Здесь как минимум пара вещей неправильные:

1. Никогда, никогда не вызывайте блокирующие функции такие как `root.mainloop()` в `async` контексте
2. Создавая корутины, не забывайте их на выполнение в цикл добавлять.

Чтобы избежать блокирования asyncio цикла событий из-за `root.mainloop()` вызова, можно руками его эмулировать с помощью периодических `root.update()` вызовов. Чтобы корутину в цикл событий добавить, достаточно `asyncio.ensure_future()` вызвать (это просто на выполнение ставит, не дожидаясь окончания):

```
#!/usr/bin/env python3
import asyncio
from tkinter import Tk, Button


async def main():
    async def coro():
        done.append(1)

    root = Tk()
    done = []
    root.protocol("WM_DELETE_WINDOW", lambda: done.append(2))
    Button(text='press me', command=lambda: asyncio.ensure_future(coro())).pack()

    # emulate root.mainloop() without blocking the asyncio event loop
    while not done:
        root.update()  # process tkinter's events
        await asyncio.sleep(.04)

asyncio.get_event_loop().run_until_complete(main())
```

поделиться улучшить этот ответ отслеживать

ответ дан 25 мар '18 в 21:17
jfs
47.6k • 9 • 73 • 256

добавить комментарий

Ваш ответ

Фантомный баг не даёт уснуть? Ищи его на StackOverflow.
Ella Svetlaya
Задать вопрос »

Связанные

3 Как в tkinter вставить картинку с сайта?

Похожие

0 Ошибка с Asyncio. Task was destroyed but it is pending!
1 Непонятный порядок выполнения asyncio-кода
1 Canvas Python Не выполняется событие
1 Проблема с asyncio Python парсинг
0 Asyncio и многопоточность
0 Python Thread Multiprocessing asyncio
0 Python3 проблема с threading и asyncio

Лента вопроса

Зарегистрируйтесь или войдите

Регистрация через Google
Регистрация через Facebook
Регистрация через почту

Отправить без регистрации

Имя

Почта
Необходима, но никому не показывается

Отправить ответ

Нажимая на кнопку «Отправить ответ», вы соглашаетесь с нашими пользовательским соглашением, политикой конфиденциальности и политикой о куки

Всё ещё ищете ответ? Посмотрите другие вопросы с метками python tkinter asyncio discord-api или задайте свой вопрос.

дизайн сайта / логотип © 2020 Stack Exchange Inc;

Используя наш сайт, вы подтверждаете, что прочитали и поняли политику о куки, политику конфиденциальности и пользовательское соглашение.