2.5 PubSub

PubSub

Если вы создаете чат-приложение с помощью Flet, вам необходимо каким-то образом передавать сообщения пользователей между сессиями. Когда пользователь отправляет сообщение, оно должно быть транслировано всем другим сессиям приложения и отображаться на их страницах.

Flet предлагает простой встроенный механизм **PubSub** для асинхронного обмена сообщениями между сессиями страниц.

Flet **PubSub** позволяет транслировать сообщения всем сессиям приложения или отправлять их только определенным подписчикам "темы" (или "канала").

Типичное использование PubSub включает:

* **Подписку** на транслирование сообщений или подписку на тему при старте сессии приложения.
* **Отправку** транслируемого сообщения или отправку на тему при каком-либо событии, например, при клике на кнопку "Отправить".
* **Отписку** от транслируемых сообщений или отписку от темы при каком-либо событии, например, при клике на кнопку "Покинуть".
* **Отписку** от всего при page.on\_close . Ниже представлен пример простого чат-приложения:

import flet as ft

def main(page: ft.Page):

page.title = "Flet Chat"

# подписка на транслируемые сообщения

def on\_message(msg):

messages.controls.append(ft.Text(msg))

page.update()

page.pubsub.subscribe(on\_message)

def send\_click(e):

page.pubsub.send\_all(f"{user.value}: {message.value}")

# очистка формы

message.value = ""

page.update()

messages = ft.Column()

user = ft.TextField(hint\_text="Your name", width=150) # Ваше имя

message = ft.TextField(hint\_text="Your message...", expand=True)

# Ваше сообщение - заполняет все доступное пространство

send = ft.ElevatedButton("Send", on\_click=send\_click) # Отправить

page.add(messages, ft.Row(controls=[user, message, send]))

ft.app(target=main, view=ft.WEB\_BROWSER)

