



# Minicurso de Bots Discord com Python

Aula 4



## Introdução aos eventos

Como já vimos, o `on_message` é uma função que é chamada sempre que uma nova mensagem é enviada no servidor.

este nome, `on_message` é o nome do evento o qual esta função está ligada

Um evento pode ser qualquer coisa que acontece no Discord.



# Introdução aos eventos

## Exemplo de eventos:

`on_member_join`: Entrada de membros no servidor

`on_member_leave`: Saída de membros no servidor

`on_message_delete`: Uma mensagem foi deletada

`on_guild_join`: O bot foi adicionado em um servidor

etc



## Introdução ao evento on\_voice\_state\_update

Este evento é responsável por qualquer ação em um canal de voz, podendo ser:

- Conectar/Desconectar de um canal de voz
- Mutar/Desmutar
- Abrir/Fechar câmera
- Abrir/Fechar stream de vídeo



# Introdução ao evento on\_voice\_state\_update

Um exemplo de utilização deste evento:

```
@client.event
async def on_voice_state_update(member, state_before, state_after):
    if state_before.self_mute != state_after.self_mute:
        # mutou ou desmutou
    elif state_before.self_video != state_after.self_video:
        # abriu ou fechou câmera
    elif not state_before.self_stream and state_after.self_stream:
        # iniciou uma stream
```



## Introdução ao evento on\_message

O evento on\_message é invocado a cada mensagem enviada no servidor, como já vimos.

Podemos acessar o objeto autor da mensagem com `message.author` e por conseguinte, `message.author.id` retorna um número inteiro que representa o identificador único deste usuário.



## Criação de um contador de mensagens

Seguindo a mesma ideia da aula passada, utilize o evento `on_message` para registrar a quantidade de mensagens enviadas de cada usuário, sem contar bots.



Criação de um comando para ranking de mensagens

Seguindo a mesma ideia da aula passada, utilize o evento `on_message` para registrar a quantidade de mensagens enviadas de cada usuário, sem contar bots.

Após, crie um comando que retorne os 10 usuários que mais enviaram mensagens no servidor.





Agora, com o `on_voice_state_update` (Avaliação #2)

Faça com que seja enviado uma mensagem no `#general` quando um usuário entrar em um canal de voz e quando sair.

Na mensagem deverá conter o nome e se entrou ou se saiu.

A função `on_voice_state_update` deve ser enviada no meu privado no Discord até às 23:55 de Domingo, 18/12.