



# Geek University

**Evolua seu lado geek!**

[www.geekuniversity.com.br](http://www.geekuniversity.com.br)



# Bibliotecas Async IO





# Bibliotecas Async IO

Quando estamos desenvolvendo uma aplicação e tendo aprendido os conceitos que vimos até o momento neste curso, temos que ficar atentos a utilizar corretamente os recursos de concorrência no nosso código.

Principalmente os conceitos de assincronismo com o recurso moderno do módulo *asyncio* do Python.

Contudo, muitas vezes precisamos utilizar bibliotecas de terceiros, como bibliotecas HTTP, acesso á bancos de dados, etc.

Lembre-se que só podemos usar *await* em funções/métodos assíncronos.

Então, por exemplo, se uma biblioteca de acesso à banco de dados não for assíncrona, não poderemos utilizar os recursos de assincronia pois estas bibliotecas por padrão não são suportadas pelo Event Loop e são portando blocking.



# Bibliotecas Async IO

Desta forma, você precisa sempre, ao utilizar uma biblioteca, se ela suporta assincronia ou caso não se encontra uma biblioteca similar mas com este suporte.

Claro que você vai buscar estas bibliotecas se fizer sentido.

Por exemplo aprendemos que usamos assincronia, geralmente, para tarefas que utilizem IO, como tarefas de bancos de dados, acesso à APIs, manipulação de arquivos em disco, etc.



# Bibliotecas Async IO

## Exemplos:

Uma biblioteca famosa para acesso HTTP em Python é a *requests*. Esta porém não suporta assincronia. Uma substituta com suporte Async é *aiohttp*

O mesmo vale para trabalhar com banco de dados PostgreSQL usamos *psycopg2*, que não suporta assincronia, porém temos a biblioteca *aiopg* que suporta Async.

Você encontra uma lista completa e atualizada de bibliotecas Async para Python em:

<https://github.com/aio-libs>

<https://github.com/python/asyncio/wiki/ThirdParty>

<https://pypi.org/search/?c=Framework+%3A%3A+AsyncIO>



# Bibliotecas Async IO

## Exemplos:

Uma biblioteca famosa para acesso HTTP em Python é a *requests*. Esta porém não suporta assincronia. Uma substituta com suporte Async é *aiohttp*

O mesmo vale para trabalhar com banco de dados PostgreSQL usamos *psycopg2*, que não suporta assincronia, porém temos a biblioteca *aiopg* que suporta Async.

Você encontra uma lista completa e atualizada de bibliotecas Async para Python em:

<https://github.com/aio-libs>

<https://github.com/python/asyncio/wiki/ThirdParty>

<https://pypi.org/search/?c=Framework+%3A%3A+AsyncIO>

Vamos para o código ver na prática alguns exemplos...



# Geek University

**Evolua seu lado geek!**

[www.geekuniversity.com.br](http://www.geekuniversity.com.br)