



3.11.5



Quick search

Go

# asyncio — Асинхронный ввод-вывод

asyncio - это библиотека для написания **параллельного** кода с использованием синтаксиса **async** / **await**.

asyncio используется в качестве основы для множества асинхронных фреймворков Python, которые предоставляют высокопроизводительные сетевые и веб-серверы, библиотеки подключений к базе данных, распределенные очереди задач и т.д.

asyncio часто идеально подходит для ввода-вывода и высокоуровневого **структурированного** сетевого кода.

asyncio предоставляет набор **высокоуровневых** API для:

- **запускайте** **сопрограммы Python** одновременно и полностью контролируйте их выполнение;
- выполнение **сетевого ввода-вывода и IPC**;
- управление **подпроцессами**;
- распределять задачи по **очередям**;
- **синхронизировать** параллельный код;

Кроме того, существуют **низкоуровневые** API для *разработчиков библиотек и фреймворков*, позволяющие:

- создавайте **циклы событий** и управляйте ими, которые предоставляют асинхронные API для **networking** запуска **subprocesses**, обработки **OS signals** и т.д.;
- реализация эффективных протоколов с использованием **транспортов**;
- **соедините** библиотеки на основе обратного вызова и код с синтаксисом **async** / **await**.

Вы можете поэкспериментировать с asyncio параллельным контекстом в REPL:

```
$ python -m asyncio
asyncio REPL ...
Используйте "await" напрямую вместо "asyncio.run()".
Введите "справка", "авторское право", "кредиты" или "лицензия" для получения дополнит
>>> импорт asyncio
>>> ожидание asyncio.sleep(10, результат='hello')
'привет'
```

**Доступность:** не Emscripten, не WASI.

Этот модуль не работает или недоступен на платформах WebAssembly wasm32-emscripTEN и wasm32-wasi. Смотрите **Платформы WebAssembly** для получения дополнительной информации.

## Ссылка

Высокоуровневые API

- **Бегуны**
- **Сопрограммы и задачи**
- **Стримы**
- **Примитивы синхронизации**

Привет, мир!

```
импортируйте asyncio

async def main():
    print('Привет ...')
    ожидайте asyncio.sleep(1)
    print('... Мир!')

asyncio.выполнить(main())
```



- [Исключения](#)

## Низкоуровневые API

- [Цикл событий](#)
- [Фьючерсы](#)
- [Транспортные средства и протоколы](#)
- [Политики](#)
- [Поддержка платформы](#)
- [Расширение](#)

## Руководства и tutorиалы

- [Высокоуровневый индекс API](#)
- [Низкоуровневый индекс API](#)
- [Разработка с использованием asyncio](#)

**Примечание:** Исходный код asyncio можно найти в [библиотеке/asyncio/](#).