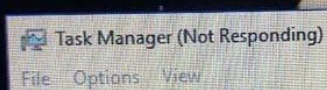


Асинхронно програмиране с Async/await



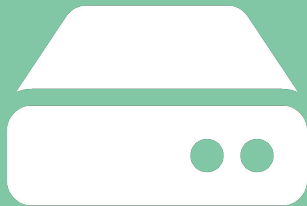
**YOU WERE SUPPOSED
TO DESTROY THEM**

NOT JOIN THEM



Task Manager (Not Responding)
File Options View

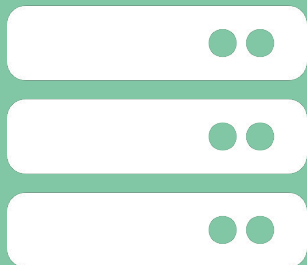
За какво ни е асинхронното програмиране?



ДИСКОВИ
ОПЕРАЦИИ



ОПЕРАЦИИ
СВЪРЗАНИ С
ПАМЕТТА



РАБОТА С
WEB API



ОПЕРАЦИИ
С БД



Асинхронно програмиране

- Често извършваме операции, които изискват да изчакаме приключването на действие
- Вместо да „замръзнем“ в чакане, действието може да се случи *асинхронно*
- Такова асинхронно изпълнение се случва чрез двойката ключови думи `async/await`



Ключова дума `async`

- Маркира метод като асинхронен
- Необходима, ако някъде в методите ще имаме действие, което ще изчакваме
- Един от начините за реализиране на асинхронно програмиране в C#
- Използва се заедно с ключовата дума `await`
- В други програмни езици (JavaScript, например) също има `async/await` концепция



Ключова дума await

- await поставя на пауза изпълнението на метода до получаването на резултат, но без да блокира викащата нишка
- await позволява изпълнението на задача без то да води до блокиране на нишката, в която е извикана
- Може да използвате await в рамките на метод отбелязан като async

Пример: Асинхронно зареждане на уеб съдържание

```
public async Task<string> GetFacts(int count)
{
    using (var client = new HttpClient())
    {
        try
        {
            var response = await client.GetAsync($"https://cat-fact.herokuapp.com/facts/random?
            animal_type=cat&amount={count}");
            response.EnsureSuccessStatusCode();
            string content = await response.Content.ReadAsStringAsync();
            return content;
        }
        catch (HttpRequestException exception)
        {
            return "Error: " + exception.Message;
        }
    }
}
```

Благодаря за вниманието!
Автор: Петър Р. Петров,
ПГЕЕ „К. Фотинов“, гр. Бургас

