**Promises and fetch() - JS**

1. Какво е Promise?

- обещание за нещо, което ще се случи в бъдещето

- обект, който ни се връща вследствие на започнато **асинхронно действие (action)**, който може би ще се изпълни. Пример: може да направим заявка, обаче сървъра отсреща да не работи. Тогава Promise-ът няма да се изпълни, а ще се reject-не.

- promise-ът може да се **resolve-не** (да се случи) или да **reject-не** (не се е случил по някаква причина)

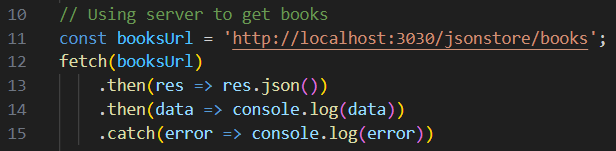
- има грижата да извика **callback-а** (функцията), който ние сме подали на **.then()** метода и да подаде бъдещата стойност

2. Какво е fetch()?

- функция, чрез която можем да направим заявка **(чрез JavaScript)**

- по дефолт прави **GET** заявка

- връща **Promise<Response>** (promise от response-a)



1) Правим заявка до някакъв сървър. Връща ни се Promise от fetch(). Ако статусът му е fulfilled, то влизаме в .then-a и res представлява информацията за response-а, който сме получили от направената заявка (след като Promise-ът се е изпълнил успешно). **Но не сме получили данните от response-а, а само информацията за този response.**

2) Трябва да конвертираме информацията от response-а в json формат. Но този .json() метод връща нов Promise. Ако статусът му е fulfilled, то влизаме в новия .then() и получаваме информацията за данните от конвертирането в json формата.

3) Ако при някой от двата Promise се появи грешка, ще отидем направо при .catch().

