JavaScript Базовий

Асинхронний код. Promise

Introduction

Охріменко Дмитро
MCT

_okhrimenko

dmitriy.okhrimenko

dokhrimenko

MCID: 9210561

Тема уроку

Асинхронний код. Promise

План уроку

1. Синхронний та асинхронний коди.
2. Функції зворотного виклику для асинхронного коду.
3. Promise.
4. Promise API.

Синхронний код

Синхронний код – код, який виконується послідовно. Кожна операція очікує на завершення попередньої.

```
function download() {
   Завантажує та повертає
   дані. Виконується ~10
   секунд
}
```

```
download();   ~10 с
download();   ~10 с
download();   ~10 с

// інші обчислення
foo();        виконається через ~30 с
```

Асинхронний код

Асинхронний код – код, який виконується паралельно, а не послідовно. Здебільшого асинхронне виконання коду передбачає виділення нових ресурсів для виконання асинхронних викликів – потоків.

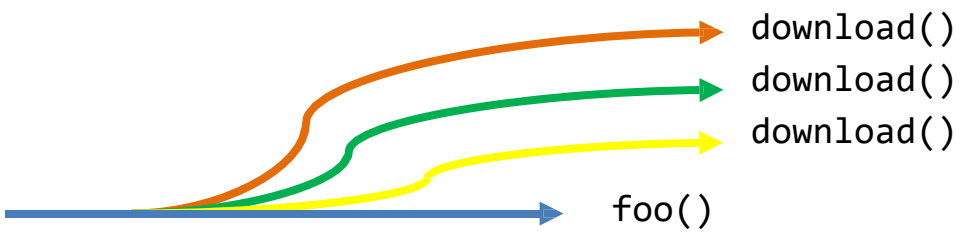
```
download();
download();
download();

// інші обчислення
foo();
```

Не чекаємо на завершення і переходимо до наступної операції.

Не чекаємо на завершення і переходимо до наступної операції.

Не чекаємо на завершення і переходимо до наступної операції.

Виконається до того, як закінчиться виконання трьох методів



Організація асинхронного коду

Варіанти організації асинхронного коду:

- Callback-функції, або Функції зворотного дзвінка.
- Promise.
- Шаблон Observer.

```
function callback(result) {
   ...
}

download(callback);
```


```
let promise = download();

promise.then(callback);
```

```
let observable = download();

observable.subscribe(callback);
```

Promise

Promise – це об'єкт, який зберігає кінцевий результат відкладеної операції. Promise – значення, яке ще не наявне. Daniel P. Friedman та David Wise запропонували термін Promise у 1976 році.

Можливі стани об'єкта promise:

- Fulfilled
- Rejected
- Pending

Може переходити із стану pendingабо в fulfilled, або в rejected.

`p.then(f, r)` – якщо p у стані **`fulfilled,`** функція f буде викликана.
`p.then(f, r)` – якщо p у стані **`rejected,`** функція r буде викликана.
У всіх інших випадках p у стані pending.

`settled` – promise перейшов у стан `rejected` або `Fulfilled.`

Висновки

- **Синхронний код** – операції виконуються послідовно.
- **Асинхронний код** – операції виконуються паралельно.
- Способи обробки асинхронної операції: **callback**, **promise**, **observer**.
- **Promise** – об'єкт, який представляє **результат асинхронної операції**.
- Основні методи promise: **then**, **catch**, **finally**.
- Якщо асинхронні операції мають виконуватися одна за одною, promise можна побудувати в **ланцюжок**.

Дякую за увагу! До нових зустрічей!

Охріменко Дмитро
MCT

MCID: 9210561


ITVDN
IT VIDEO DEVELOPERS NETWORK