**Yup Technology**

Final İmtahan

Abdurahmanov Soltan.

1. <https://jsonplaceholder.typicode.com/posts> - linkə daxil olaraq “JSON” – fayllarını “fetch” – istifadə edərək cədvələ dartın.
2. Javascript vasitəsilə slider yığın. Belə ki, next və prev iconlarına toxunduqda şəkillər dəyişilsin.

Nümunə:



1. Javascriptdə “callback” funksiyalarını və “promise” izah edin və nümünə göstərin.

CALLBACK

Kompüter proqramlaşdırmasında "zəngdən sonra" funksiyası kimi də tanınan geri çağırış digər koda arqument kimi ötürülən icra edilə bilən koda hər hansı istinaddır; digər kodun müəyyən bir zamanda kodu geri çağırması (icra etməsi) gözlənilir. Bu icra sinxron geri çağırışda olduğu kimi dərhal ola bilər və ya asinxron geri çağırışda olduğu kimi daha gec vaxtda baş verə bilər. Proqramlaşdırma dilləri geri çağırışları müxtəlif yollarla dəstəkləyir, çox vaxt onları alt proqramlar, lambda ifadələri, bloklar və ya funksiya göstəriciləri ilə həyata keçirir.

// Callback Function Example

function greet(name, myFunction) {

console.log('Hello world');

// callback function

// executed only after the greet() is executed

myFunction(name);

}

// callback function

function sayName(name) {

console.log('Hello' + ' ' + name);

}

// calling the function after 2 seconds

setTimeout(greet, 2000, 'John', sayName);

PROMISE

Təsəvvür edin ki, siz ən yaxşı müğənnisiniz və fanatlar gecə-gündüz gələcək mahnınızı soruşurlar.

Bir az rahatlıq əldə etmək üçün onu dərc olunanda onlara göndərəcəyinizə söz verirsiniz. Siz azarkeşlərinizə siyahı verirsiniz. Onlar e-poçt ünvanlarını doldura bilərlər ki, mahnı əlçatan olduqda, abunə olan bütün tərəflər onu dərhal qəbul etsinlər. Bir şey çox səhv getsə belə, deyək ki, studiyada yanğın baş versə də, mahnını dərc edə bilməsəniz, yenə də xəbərdar olacaqlar.

Bu, proqramlaşdırmada tez-tez rast gəldiyimiz şeylər üçün real həyat bənzətməsidir:

Bir şey edən və vaxt alan “kod istehsal edən”. Məsələn, məlumatları şəbəkə üzərindən yükləyən bəzi kodlar. Bu, "müğənni"dir.

Hazır olduqdan sonra "istehsal kodunun" nəticəsini istəyən "istehlak kodu". Bir çox funksiyalar bu nəticəyə ehtiyac duya bilər. Bunlar "azarkeşlər"dir.

Söz "istehsal kodu" və "istehlak kodu"nu bir-birinə bağlayan xüsusi JavaScript obyektidir. Bənzətməmiz baxımından: bu “abunə siyahısı”dır. "İstehsal kodu" vəd edilmiş nəticəni əldə etmək üçün nə vaxt lazımdırsa, onu alır və "vəd" bu nəticəni hazır olduqda bütün abunə olunan kodlara təqdim edir.

const promise1 = new Promise((resolve, reject) => {

setTimeout(() => {

resolve('foo');

}, 300);

});

promise1.then((value) => {

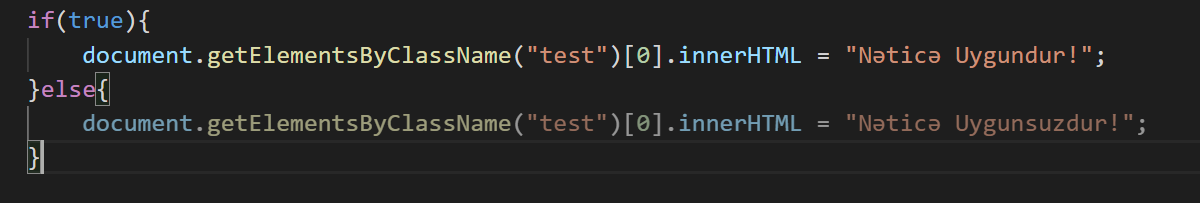
console.log(value);

// expected output: "foo"

});

console.log(promise1);

// expected output: [object Promise]

1. 

Yuxarıdakı kodu daha qısa necə yaza bilərsiniz.

1. İf(true){

let test1 = document.getElementByClassName(“test”)[0].innerHTML = “Nəticə Uyğundur!;

}else{

let test2 = document.getElementByClassName(“test”)[0].innerHTML = “Nəticə Uyğunsuzdur!;

}

test1;

test2;

1. ? "true" : "false"