

Таймери

## Таймери

- ***setTimeout*** -- дозволяє нам запускати вказану функцію ***один раз*** через певний інтервал часу (у мілісекундах)
- ***setInterval*** -- дозволяє нам запускати вказану функцію ***багаторазово***, постійно (поки не зупинимо ) через вказаний інтервал часу (у мілісекундах).

Опис	Загальна форма	Приклад
<p><b><u>Встановлення таймера</u></b></p> <p><b><i>setTimeout</i></b></p> <p>дозволяє нам запускати вказану функцію <b><i>func</i></b> <b><i>один раз</i></b> через певний інтервал часу <b><i>delay</i></b> (у мілісекундах)</p> <p>При цьому функція повертає числовий ідентифікатор таймера, що може бути використано для його зупинки</p>	<p>----- Встановлення таймера -----</p> <pre>let зм._ід. = setTimeout(     func,     delay )</pre>	<p>Приклад. Вивести слово «Hello» через 2 секунди (2000 мілісекунд)</p> <pre>let timeoutID = setTimeout(     () =&gt; alert("Hello"),     2000 )</pre>
<p><b><u>Передача додаткових параметрів у функції</u></b></p> <pre>arg0, arg1, ...</pre> <p>ці парметри передаються функції <b><i>func</i></b> під час виклику (тобто, коли пройде вказаний період часу <b><i>deLay</i></b> )</p>	<p>----- Встановлення таймера -----</p> <pre>let зм._ід. = setTimeout(     func,     delay,     arg0,     arg1,     ... )</pre>	<p>Приклад. Вивести привітання <b><i>один раз</i></b> через секундупісля відкриття</p> <pre>const intervalID = setTimeout(     myFunc,     1000,     "Hello",     "Ivan" )</pre> <pre>function myFunc(greeting, name) {     console.log(`         \${greeting} \${name}     `); }</pre>

## Встановлення таймера

Опис	Загальна форма	Приклад
<p><b><u>Встановлення таймера</u></b>  <b><i>setTimeout</i></b>          дозволяє нам запускати вказану функцію <b><i>func</i></b> <b><i>один раз</i></b> через певний інтервал часу <b><i>delay</i></b> (у мілісекундах)          При цьому функція повертає числовий ідентифікатор таймера, що може бути використано для його зупинки</p>	<p>----- Встановлення таймера -----</p> <pre>let <span style="border: 1px solid black;">зм._ід</span>. = <b>setTimeout</b>(     func,     delay )</pre>	<p>Приклад. Вивести слово «Hello» через 2 секунди (2000 мілісекунд)</p> <pre>let timeoutID = <b>setTimeout</b>(     () =&gt; alert("Hello"),     2000 )</pre>
<p><b><u>Зупинка таймера</u></b>  <b><i>clearTimeout</i></b>          який був попередньо створений і ідентифікатор якого було збережено у деякій змінній <b><i>timeoutID</i></b></p>	<p>----- Зупинка таймера -----</p> <pre><b>clearTimeout</b>(<span style="border: 1px solid black;">ідентифікатор</span>)</pre>	<pre><b>clearTimeout</b>(timeoutID)</pre>

Задача. Через 3 секунди після відкриття сторінки привітати користувача

Опис	Загальна форма	Приклад
<p><b><u>Встановлення таймера</u></b></p> <p><b><i>setInterval</i></b></p> <p>дозволяє нам запускати вказану функцію <b><i>func багато разів</i></b> (поки не зупинять) через певний інтервал часу <b><i>delay</i></b> (у мілісекундах)</p> <p>При цьому функція повертає числовий ідентифікатор таймера, що може бути використано для його зупинки</p>	<p>----- Встановлення таймера -----</p> <pre>let зм._ід.= setInterval (     func,     delay )</pre>	<p>Приклад. Вивести слово «Hello» через 2 секунди (2000 мілісекунд)</p> <pre>let timeoutID = setInterval (     () =&gt; alert("Hello"),     2000 )</pre>
<p><b><u>Передача додаткових параметрів у функції</u></b></p> <p>arg0, arg1, ...</p> <p>ці парметри передаються функції <b><i>func</i></b> під час виклику (тобто, коли пройде вказаний період часу <b><i>deLay</i></b>)</p>	<p>----- Встановлення таймера -----</p> <pre>let зм._ід.= setInterval (     func,     delay,     arg0,     arg1,     ... )</pre>	<p>Приклад. Виводити привітання кожну секунду</p> <pre>const intervalID = setInterval(     myFunc,     1000,     "Hello",     "Ivan" )  function myFunc(greeting, name) {     console.log(`         \${greeting} \${name}     `); }</pre>

## Встановлення таймера

Опис	Загальна форма	Приклад
<p><b><u>Встановлення таймера</u></b>  <b><i>setInterval</i></b></p> <p>дозволяє нам запускати вказану функцію <b><i>func багато разів</i></b> (поки не зупинять) через певний інтервал часу <b><i>delay</i></b> (у мілісекундах)          При цьому функція повертає числовий ідентифікатор таймера, що може бути використано для його зупинки</p>	<p>----- Встановлення таймера -----</p> <pre>let <span style="border: 1px solid black;">зм._ід</span>. = setInterval (     func,     delay )</pre>	<p>Приклад. Вивести слово «Hello» через 2 секунди (2000 мілісекунд)</p> <pre>let timeoutID = setInterval (     () =&gt; alert("Hello"),     2000 )</pre>
<p><b><u>Зупинка таймера</u></b>  <b><i>clearInterval</i></b></p> <p>який був попередньо створений і ідентифікатор якого було збережено у деякій змінній <b><i>timeoutID</i></b></p>	<p>----- Зупинка таймера -----</p> <pre>clearInterval(<span style="border: 1px solid black;">ідентифікатор</span>)</pre>	<pre>clearInterval (timeoutID)</pre>

Задача. Кожні 10 секнуд нагадувати користувачу, що бажано зареєструватися (виводити повідомлення)



Задача з співбесіди

Що побачить користувач?

```
for (let i = 1; i < 5; i++) {  
  setTimeout(function () {  
    alert(i)  
  }, 2000)  
}
```

Задача з співбесіди

Що побачить користувач?

```
let i
for (i = 1; i < 5; i++) {
  setTimeout(function () {
    alert(i)
  }, 2000)
}
```

Задача з співбесіди

Що побачить користувач?

```
let num = 0
let myTimer = setInterval(() => {
  if (num === 5) {
    alert('Stop')
    clearInterval(myTimer)
  } else alert(num++)
}, 1000)
```

Задача з співбесіди

Що побачить користувач?

```
function myFunc(num) {  
  setTimeout(() => {  
    if (num < 5) {  
      alert(num)  
      myFunc(num + 1)  
    }  
  }, 2000)  
}  
  
myFunc(1)
```