

1. タイムアウト処理¹

ある時間が経過した時点で何かの処理を始める。

こうした「タイマー」設定には「**setTimeout**」メソッドが使われる。【PR378】²

サンプル：

3 秒後に関数 **disp()** を実行する。

文法は、

setTimeout (処理, ミリ秒単位の時間);

この **setTimeout** メソッドは、設定した時間に 1 度だけ指定した処理をするメソッドです。

したがって、

setTimeout("disp()",3000); は、

3000 ミリ秒 (=3 秒) 後に、**disp** 関数を呼び出す事を示しています。

サンプルでは、定義した関数 **disp()** を呼び出しているので、事前に設定する必要がある。

```
function disp(){    //関数 disp の定義
    document.write("3 秒経ったよ！"); //文字を表示する document オブジェクト。
}
```

<body>の中で使う **setTimeout** メソッドは、1 行だけ。

setTimeout("disp()",3000);

関数 **disp()** が、3000 ミリ秒 (3 秒) 経過した時点で、呼び出される。

¹ **Date** オブジェクト【PR124】以降参照

² これらタイマーに関するメソッドの上位オブジェクトは「**window**」です。通常は、上位オブジェクトを記述しなくても動作します。また、**window** オブジェクトを使うことによって、このウィンドウ別にタイマーの設定もできます。

プログラム 1 : 「3 秒後にメッセージが、表示される」

上のサンプルをまとめて、3 秒後にメッセージを表示するプログラムを作成する。

```
<html>
<head>
<TITLE>setTimeout</TITLE>
<script LANGUAGE="JavaScript">
  function disp(){ // 関数の事前定義
    document.write("<H1>Booom! 3 秒経ちました</H1>");
  }
</script>
</head>
<body>
  3 秒後に表示が変わります。
<script>
  setTimeout("disp()",3*1000); // タイムアウト処理で、disp()関数を読み込む
</script>
</body>
</html>
```

また、設定した時間に繰り返し処理を行う「**setInterval**」メソッド³もある。

```
setInterval("disp()", 5*100)
```

この場合、1/2 秒ごとに disp()処理を行う。

³ PR378

1-2. タイムアウト処理のクリア

ある一定時間が経過した時、その時間経過や時間の進行を止めたい時には、「タイムアウト処理のクリア」を実行する。

タイムアウトの処理をクリアするには、`clearTimeout` メソッドを使う。【PR378】

このメソッドで、クリアする場合は、`setTimeout` メソッドから ID を取得しなければならない。

`timerID=setTimeout("処理する関数 ", ミリ秒時間);`

処理する関数には「`“`」で囲むことが必要

例 : `timerID=setTimeout("timer()",1000);`

タイムアウト処理をクリアするためには、メソッド `clearTimeout` に、ID を使う。

`clearTimeout(setTimeout で取得したタイマーID) ;`

例 : `clearTimeout(timerID);`

サンプル :

```
var timerID=setTimeout("disp()", 10*1000);
```

実行して、この場合は、10 秒以内に下の `clearTimeout()` を実行すれば、

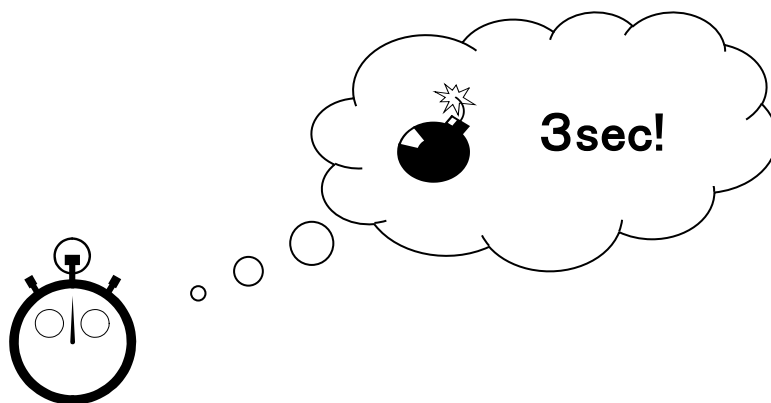
```
clearTimeout(timerID);
```

タイマーは停止し、関数 `disp()` は、実行されない。

プログラム2 : ストップウォッチ

ストップボタンを押された時、つまり、onClick のイベントが発生した時、timeoutID で得られた「タイマーID」で、clearTimeout()メソッドを働かせ、ストップウォッチを止める。

```
<html>
<head>
<script>
var cnt;      // カウンタ変数
function StopW(){
    cnt++;
    document.myForm.display.value =cnt;      //カウンターの値をテキストボックスに渡す
    timeoutID = setTimeout("StopW()",1000);  // 1 秒ごとに自分を呼び出す
}
</script>
</head>
<body>
<center>
<h2>ストップウォッチ<hr>
<form NAME=myForm>
<input type="text" NAME="display" SIZE=10>秒</h2><br>
<input type="button" value="START" onClick="cnt=0;StopW()">
<input type="button" value="STOP" onClick="clearTimeout(timeoutID)">
</form>
</center>
</body>
</html>
```



このプログラムでは、function StopW()の中で、自分自身「StopW()」を呼び出している⁴。こうした自分自身を呼び出すことを「再帰（さいき）」と呼ぶ。

⁴ ここでの働きは「1 秒おきに」