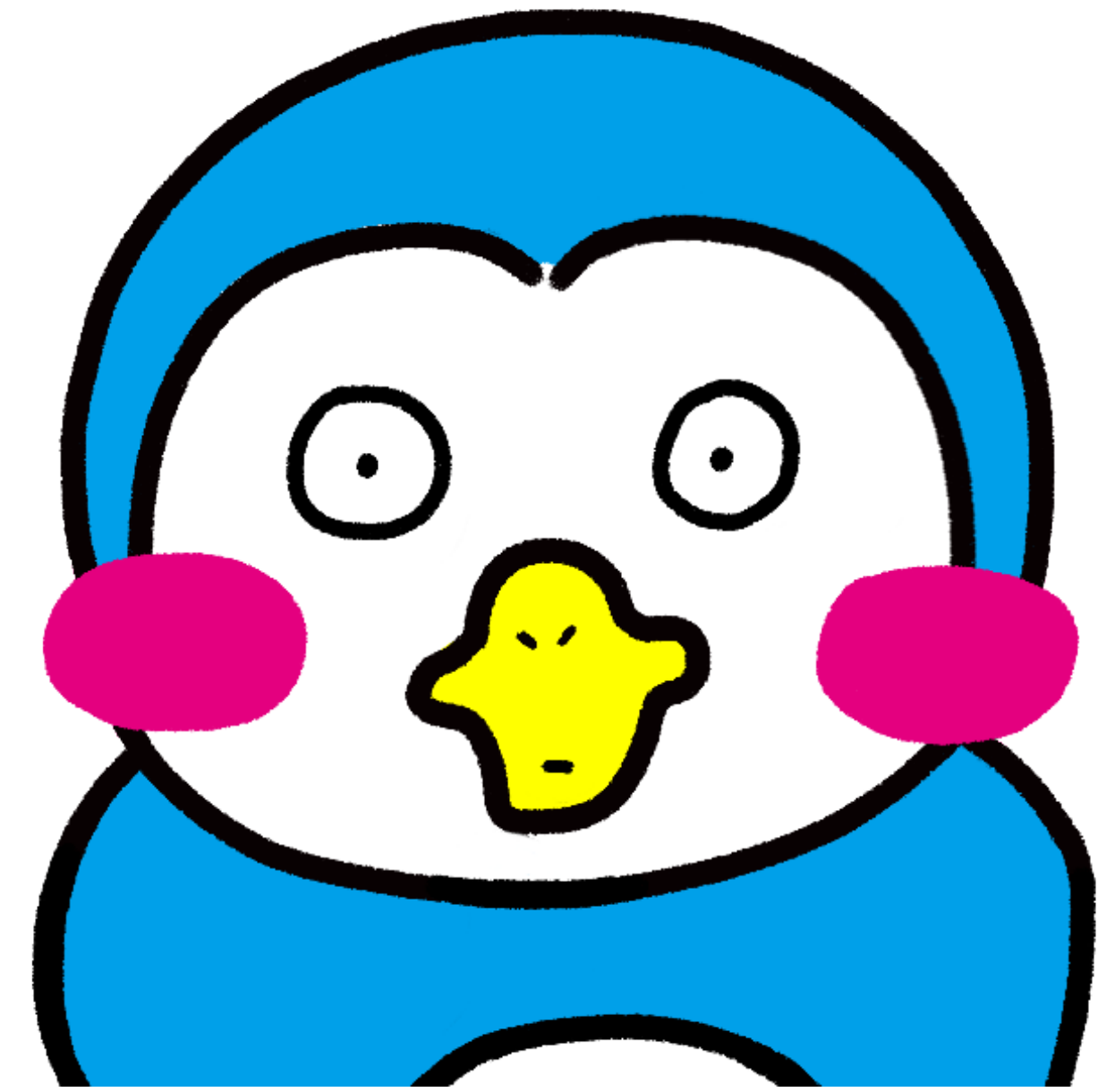


타이머를
만들어 보자



타이머 관련 기능들

setTimeout



정해진 시간이 지나고 나면 주어진 함수를 실행 해주는 타이머 메소드!

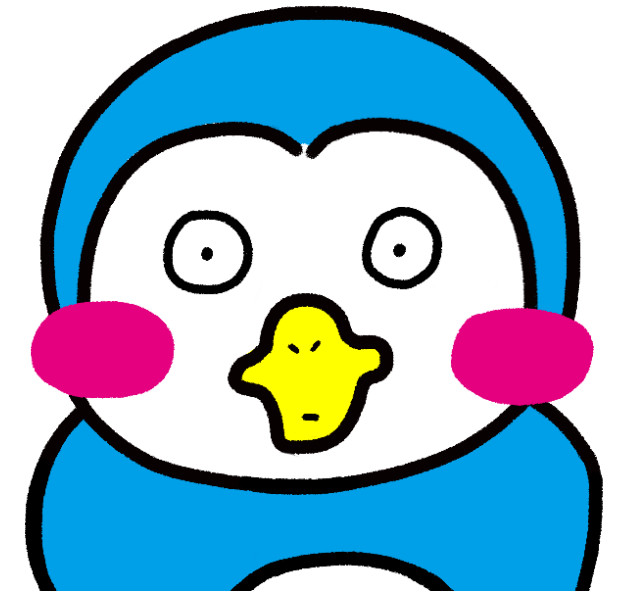
사용 방법

```
setTimeout(실행할_함수, ms_단위의_시간)
```

사용 예

```
// 1000ms 가 지나고 나면 함수를 실행한다!  
setTimeout(function(){  
  console.log("재미있다!")  
}, 1000)
```

setInterval



일정한 시간 간격에 따라 함수를 반복 실행할 수 있도록 해주는 타이머 메소드!

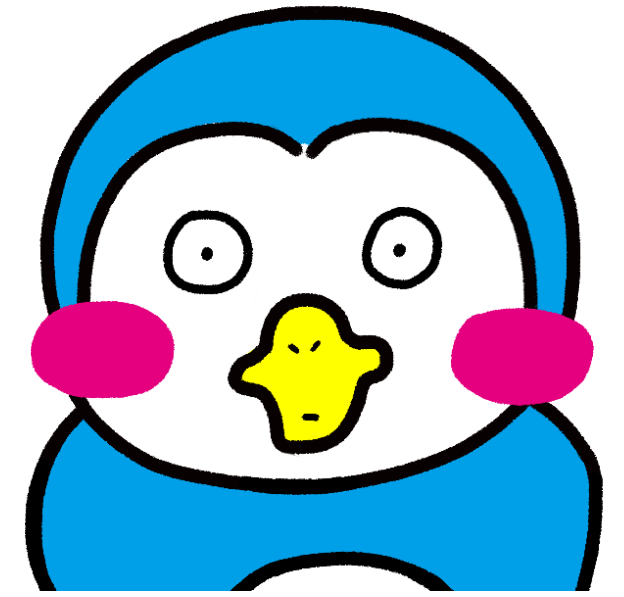
사용 방법

```
setInterval(반복_실행할_함수, ms_단위의_시간)
```

사용 예

```
// 500ms 마다 함수를 반복 실행한다!  
setInterval(function(){  
  | console.log("안녕하세요^^")  
}, 500)
```

clearInterval

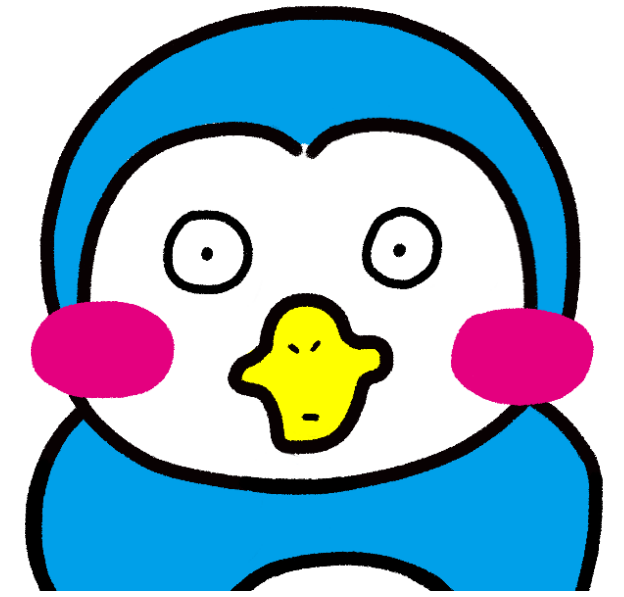


setInterval 메소드가 호출되어 반복 실행할 함수 타이머 를 등록하면, 타이머 는 0이 아닌 숫자를 반환한다. 숫자는 타이머의 ID를 의미하며, 이를 clearInterval 메소드에 전달하면 해당 타이머의 반복 실행이 취소된다.

```
// 셋팅된 타이머의 반환값(ID)을 변수에 저장하자
let timer;
timer = setInterval(function(){
  console.log("안녕하세요^^")
}, 500)

// 셋팅된 타이머를 멈춰주세요(취소해주세요)!
clearInterval(timer)
```

내용 정리



- setTimeout 메소드는 정해진 시간이 지나고 나면 주어진 함수를 실행한다.
- setInterval 메소드는 시간 간격에 따라 주어진 함수를 반복 실행한다.
- setInterval 메소드는 0이 아닌 숫자를 반환하는데, 이는 타이머의 ID이다.
- clearInterval 메소드는 주어진 ID에 해당하는 타이머를 제거한다(멈춘다).