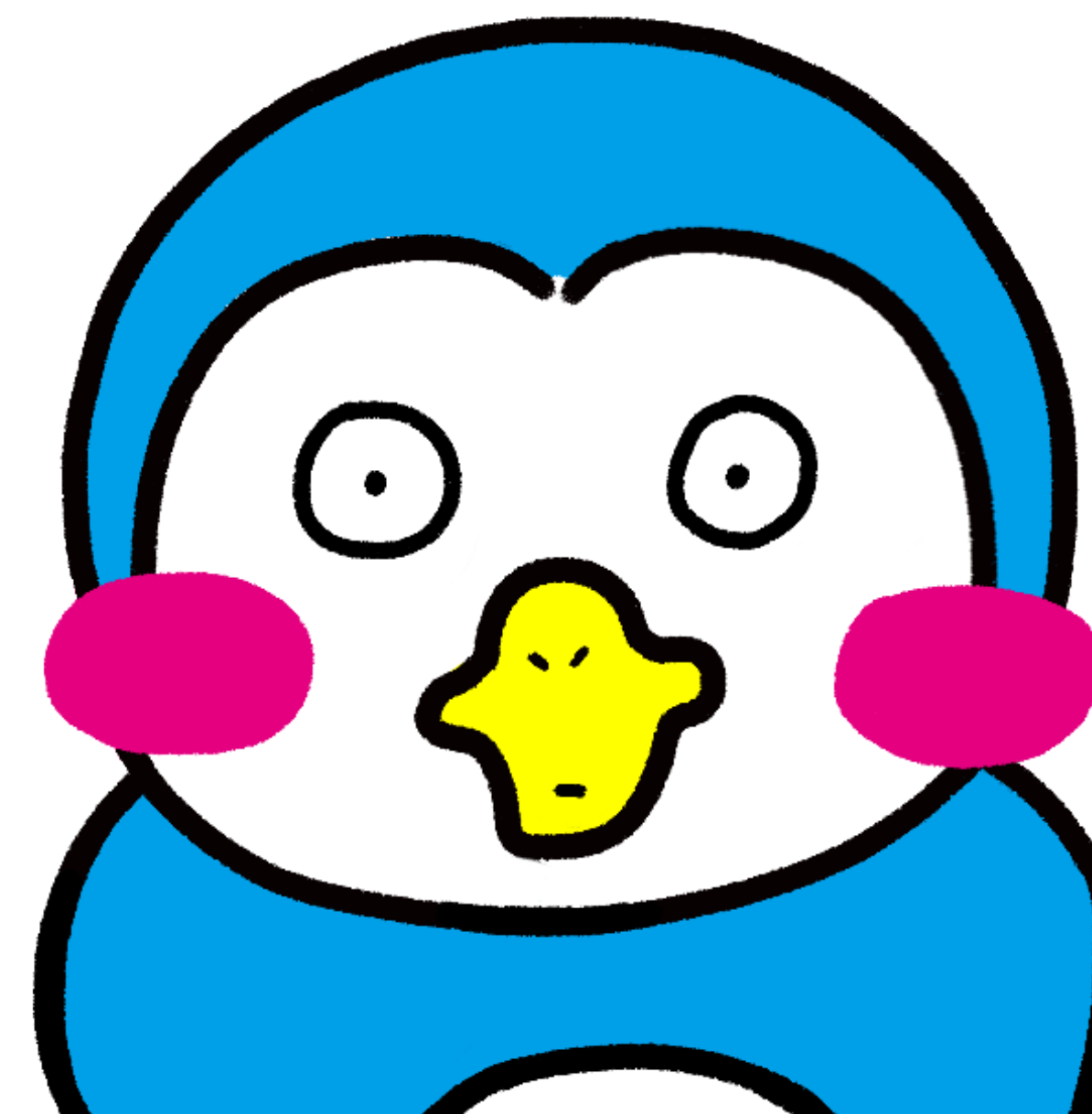


기능을 만들자
이름까지 붙여서



함수 1편

함수



함수는 '호출될 수 있는 코드 조각'이다. 변수를 선언하고 데이터를 대입하면 변수의 이름을 데이터 대신 사용할 수 있는 것처럼, 함수를 선언하면 함수의 이름을 코드 조각 대신 사용할 수 있다.

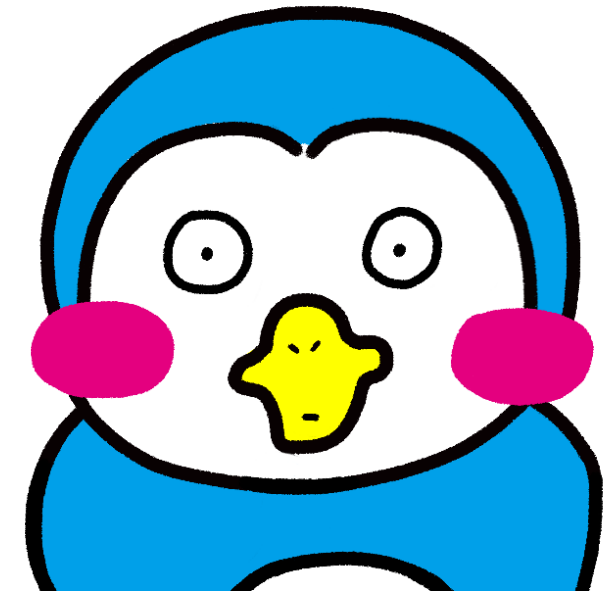
```
let value = "문자열"
```

```
// value는 "문자열" 대신 사용된다!  
console.log(value)
```

```
let work = console.log("!")
```

```
// work는 콘솔에 !를 출력하는 기능 대신 사용된다?  
work
```

함수



함수는 '호출될 수 있는 코드 조각'이다. 변수를 선언하고 데이터를 대입하면 변수의 이름을 데이터 대신 사용할 수 있는 것처럼, 함수를 선언하면 함수의 이름을 코드 조각 대신 사용할 수 있다.

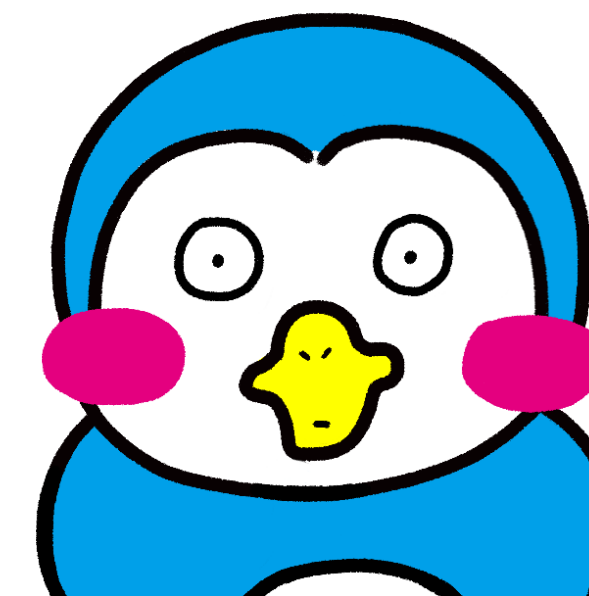
```
let value = "문자열"
```

```
// value는 "문자열" 대신 사용된다!  
console.log(value)
```

```
let work = console.log
```

```
// work는 콘솔에 !를 찍는 기능 대신 사용된다?  
work
```

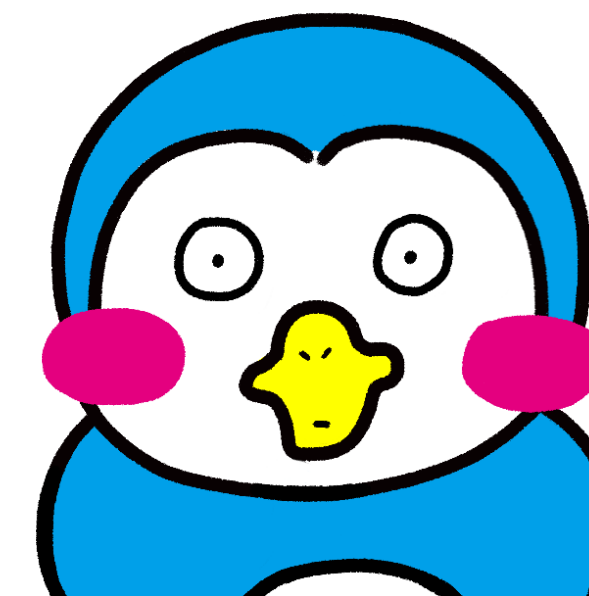
함수를 만드는 두 가지 방법



함수를 만드는 방법 중 대표적인 두 가지를 알아보자.

- 함수 선언식
- 함수 표현식

함수 선언식

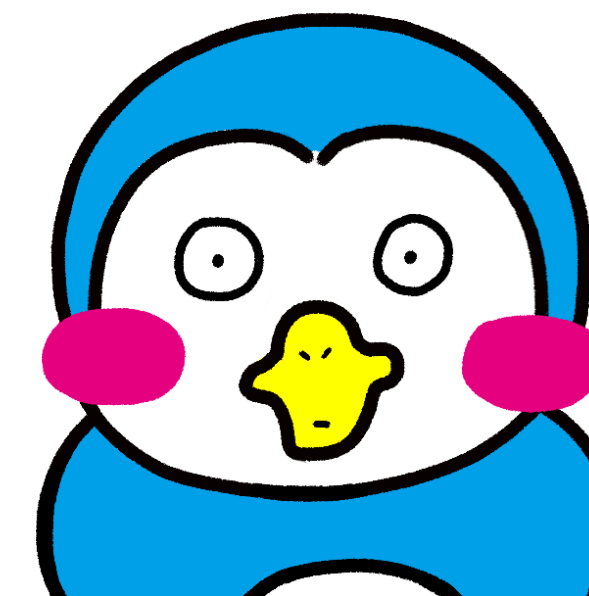


함수 선언식은 다음과 같은 형태를 가진다.

```
function 함수명() {  
    // 함수의 기능을 표현한 구문  
}
```

=> 위와 같은 선언 이후, 함수명은 중괄호 안의 기능 대신 사용될 수 있다!

함수 표현식

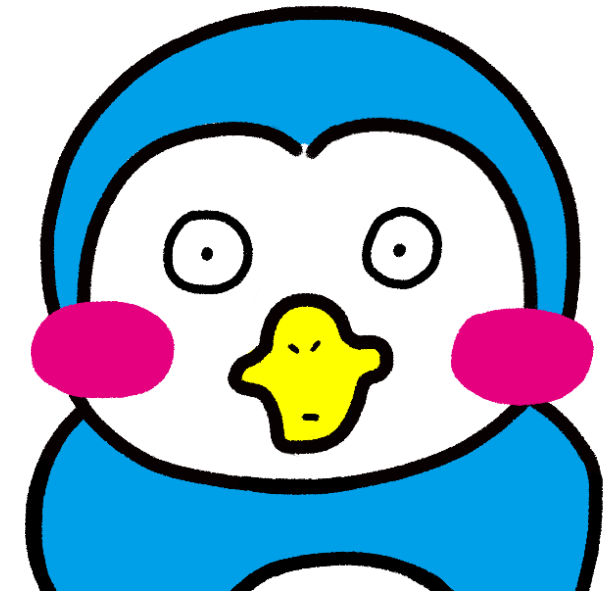


함수 표현식은 다음과 같은 형태를 가진다.

```
const 함수명 = function(){  
  // 함수의 기능을 표현한 구문  
}
```

=> 마찬가지로 선언 이후, 함수명은 중괄호 안의 기능 대신 사용될 수 있다!

만들었으면 사용을 해야지



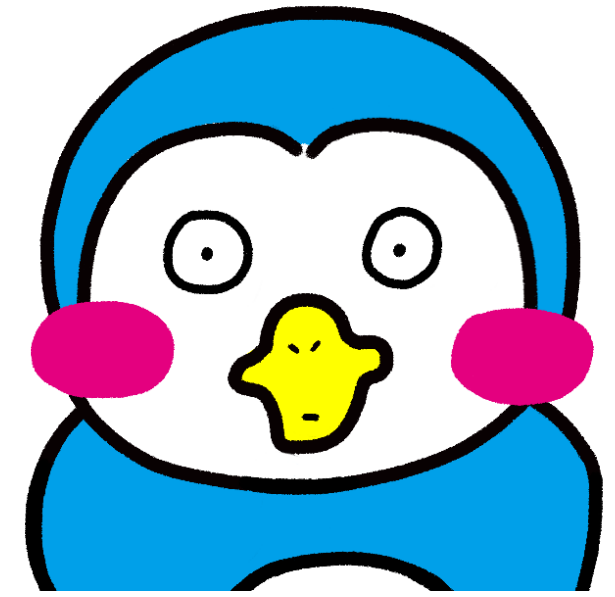
함수가 만들어지고 나면, 함수명은 스스로 보관하고 있는 구문 대신 사용될 수 있다. 그리고 함수가 사용되기 위해서는, 함수를 '호출'해야만 한다!

```
const sayHello = function(){  
  let number = 3 + 3  
  console.log(number)  
}
```

```
sayHello() // 함수 호출!
```

=> 함수를 호출할 때는 함수의 이름 뒤에 반드시 소괄호를 붙여줘야 합니다!

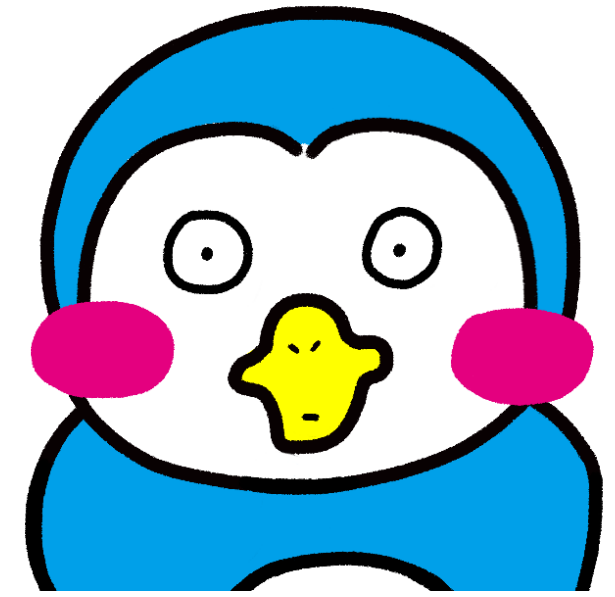
함수의 이름을 지을 때는



함수의 기능을 호출하기 위해서는 함수의 이름을 알아야 한다. 함수 이름을 정할 때의 규칙은 변수 이름 정할 때의 규칙과 유사하나, 다음과 같은 사항들을 추가적으로 고려하는 것이 좋다.

- 함수의 기능을 적절하게 표현할 수 있는 이름을 사용하자
- 명사보다는 동사로 된 이름을 사용하자(기능이므로)
- 소문자로 시작하되, 여러 단어가 섞인 경우 카멜표기법을 사용하자

내용 정리



- 함수는 '호출될 수 있는 코드 조각'이다.
- 함수는 선언식, 표현식 같은 방법을 이용해 만들 수 있고, 사용을 위해서는 함수를 호출해야 한다.
- 함수 호출시에는 반드시 함수명 뒤에 소괄호를 붙인다.
- 함수 안에서 선언한 변수는 함수 안에서만 쓸 수 있다.
- 선언식은 함수 정의보다 호출문을 앞서 작성할 수 있지만, 표현식은 그럴 수 없다.