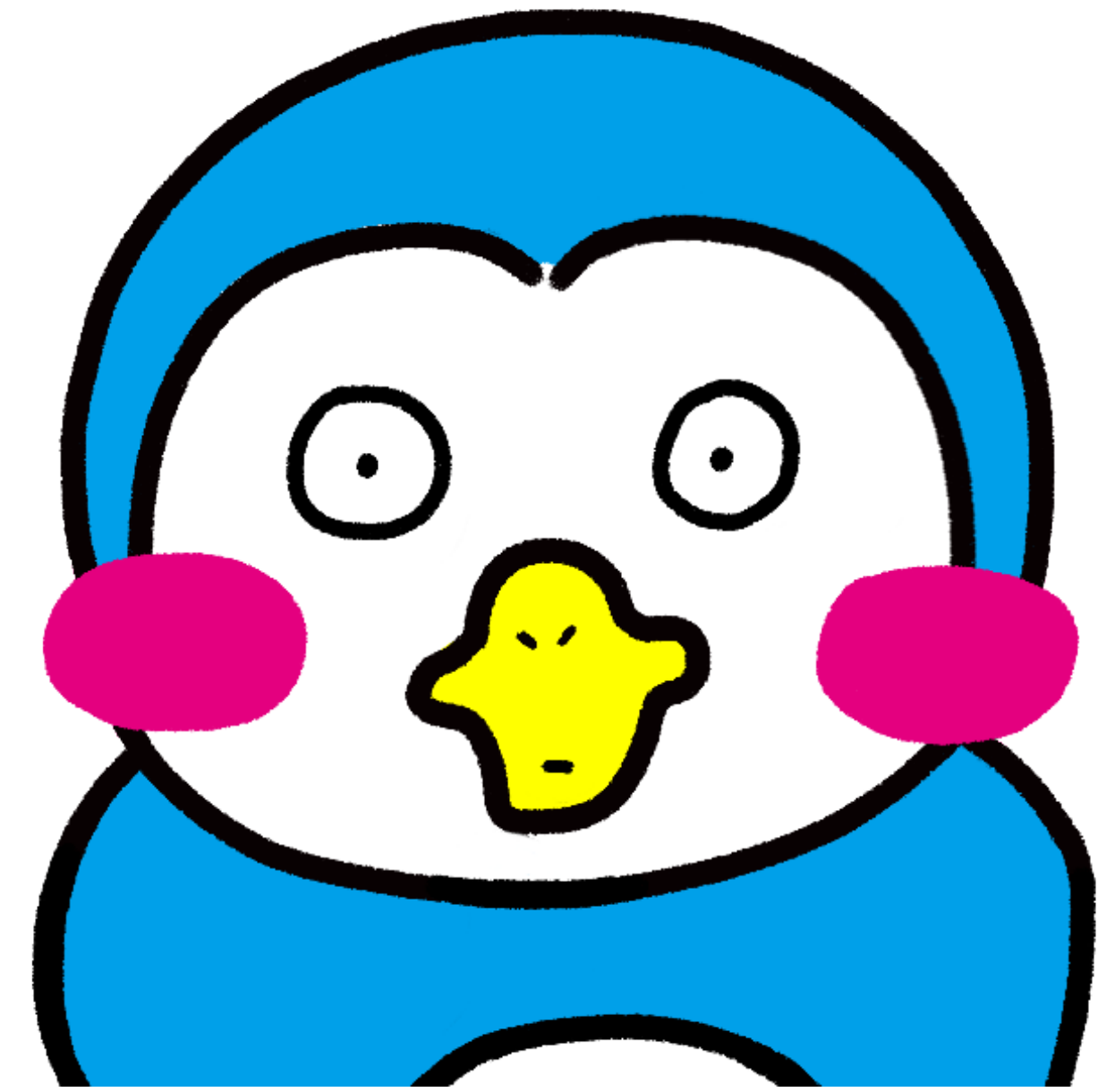
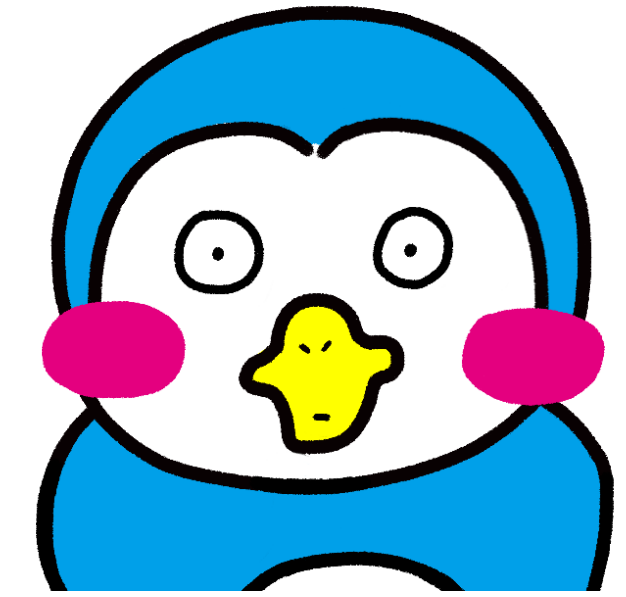


일 더하기 일은
일이 될 수 있다?!



템플릿 리터럴



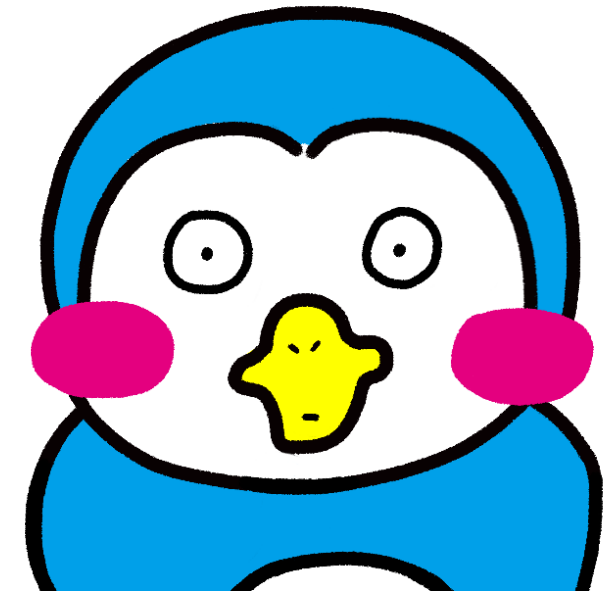
템플릿 리터럴

기존의 문자열은 따옴표를 이용해 표현했지만, 템플릿 리터럴은 백틱을 이용해 표현한다.

```
// 따옴표를 이용한 기존의 문자열  
const str1 = '작은 따옴표'  
const str2 = "큰 따옴표"  
  
// 백틱을 이용한 템플릿 리터럴  
const str3 = `이게 백팁입니다`
```

=> 백틱은 물결(~) 키를 쉬프트(shift) 없이 누르면 입력할 수 있습니다.

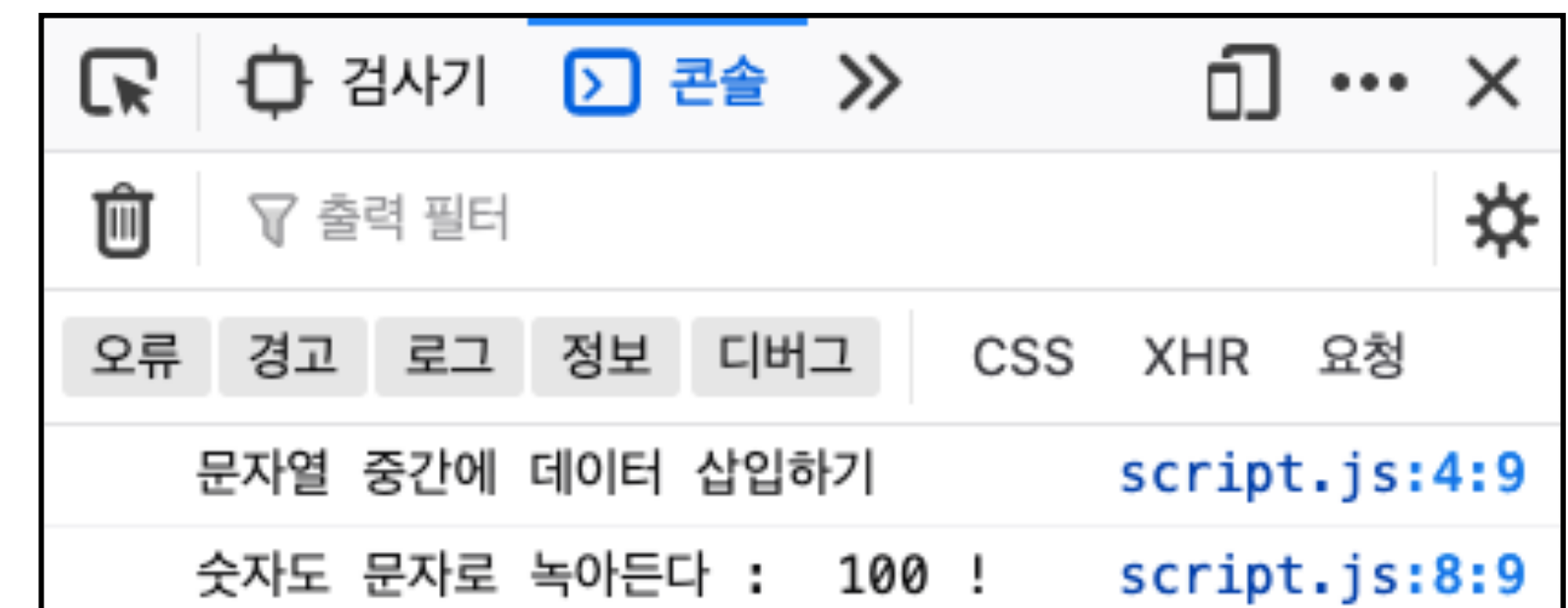
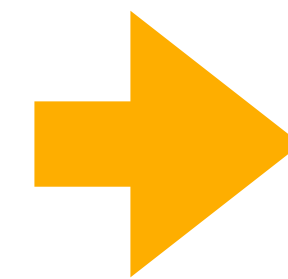
표현식 삽입 방법



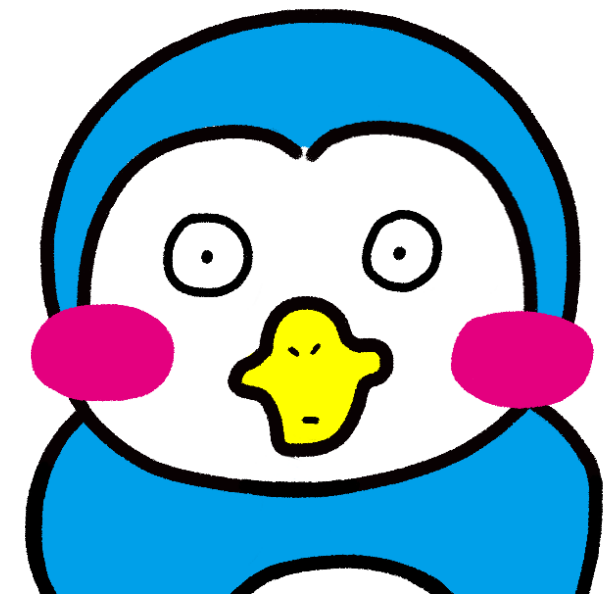
템플릿 리터럴은 표현식을 내장할 수 있는 문자열 표현법이다. 이는 문자열의 내용에 데이터를 삽입한다는 것을 의미한다. 템플릿 리터럴로 표현한 문자열 내부에 플레이스홀더(`${}`)를 기입하고, 그 안에 데이터를 기입하면 데이터는 문자열의 멤버가 된다.

```
const data1 = "데이터"
const str1 = `문자열 중간에 ${data1} 삽입하기`
console.log(str1)

const data2 = 100
const str2 = `숫자도 문자로 녹아든다 : ${data2} !`
console.log(str2)
```

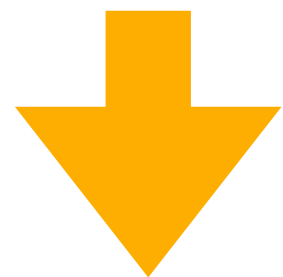


표현식 삽입 방법



데이터를 반환하는 메소드 또한 데이터와 마찬가지로 취급할 수 있다.

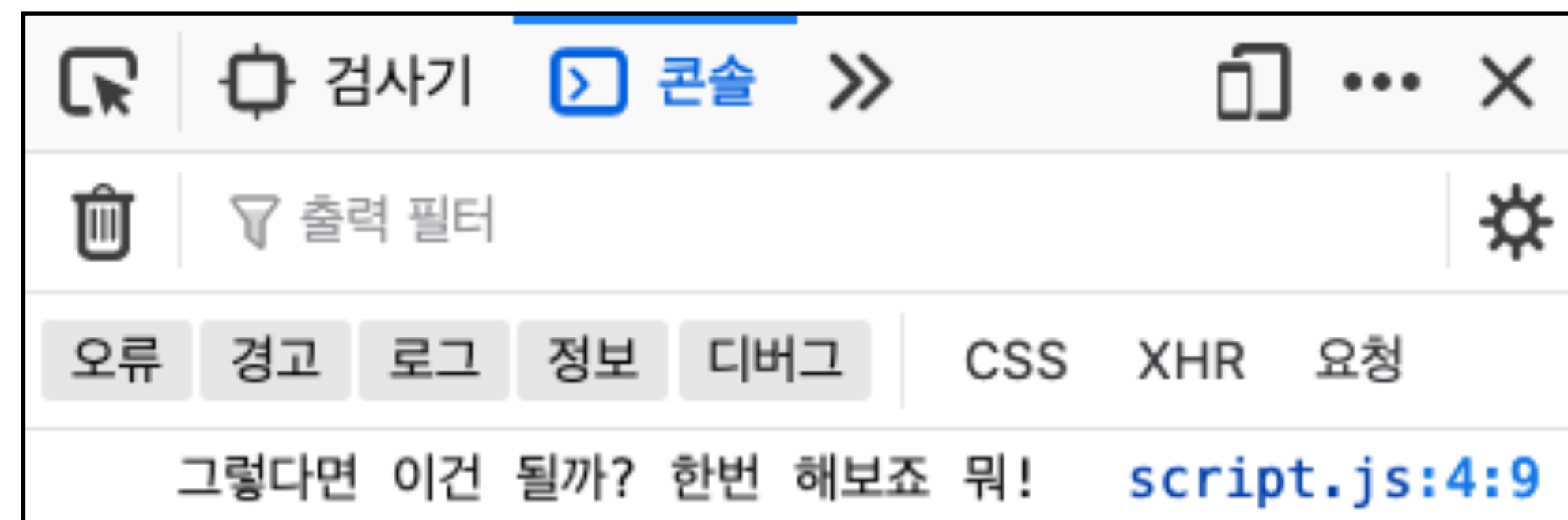
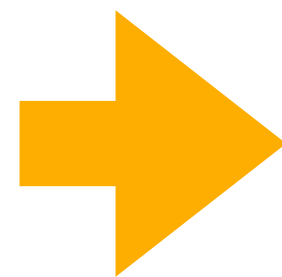
```
const str = `그렇다면 이건 될까? ${prompt('그렇다면 이건 될까?')}`  
console.log(str)
```



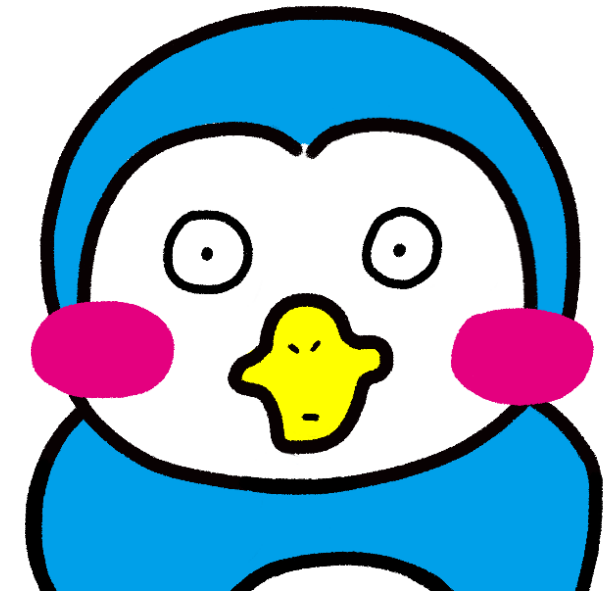
file://

그렇다면 이건 될까?

취소 확인



내용 정리



- 템플릿 리터럴은 표현식을 내장할 수 있는 문자열 표현법이다.
- 백틱을 이용해 표현하며, 문자열에 데이터를 삽입할 수 있다.
- 데이터 삽입 시에는 플레이스홀더(`${}`)를 사용하며, 삽입된 데이터는 문자열의 멤버가 된다.
- 데이터를 대신해 사용될 수 있는 것이라면 무엇이든 템플릿 리터럴에 삽입될 수 있다. 변수, 메소드 모두 가능.