
HTML에서 웹앱까지

7주차_01

한 동 대 학 교
김군오 교수

학습 목표:

자바스크립트 변수와 연산자 알아보기

학습내용:

1. 자바스크립트 소개
2. 변수의 정의와 사용
3. 연산자 종류와 사용

자바스크립트 소개



- 웹 브라우저에서 동작하는 스크립트 언어
- HTML 요소 조작, 사용자 입력 처리 가능
- 가장 인기있는 프로그래밍 언어
 - 앱 개발
 - 서버측 프로그램 개발
 - 사물인터넷 개발 등
 - 다양한 방면에서 이용됨

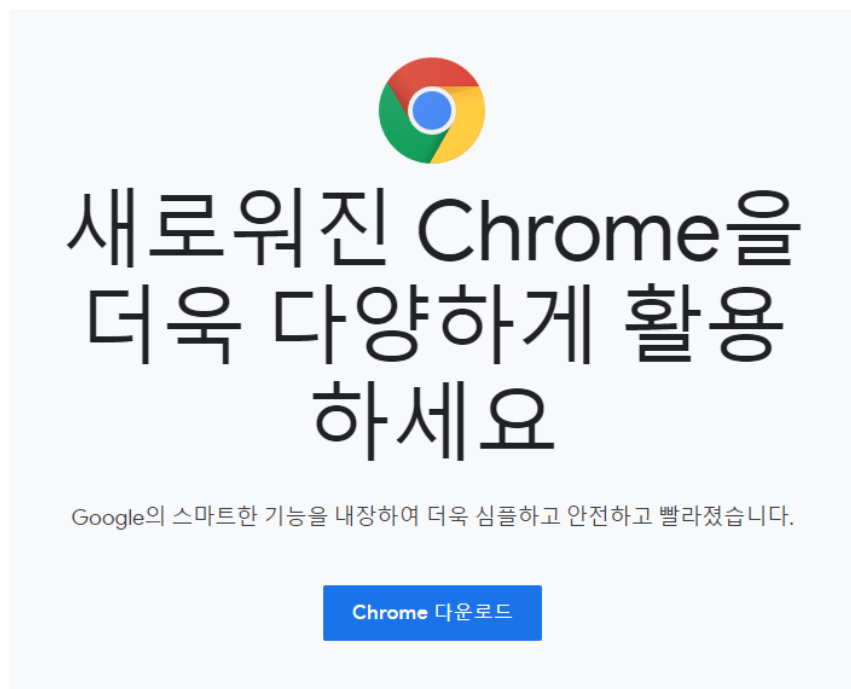
자바스크립트 장점



- 웹 브라우저에서 코드 작성 및 실행가능
- 데이터베이스부터 앱까지 모두 개발 가능
- 타 언어에 비해 배우기 쉬움

구글 크롬 설치

- <https://www.google.com/chrome>



구글 크롬 설치

- <https://www.google.com/chrome>



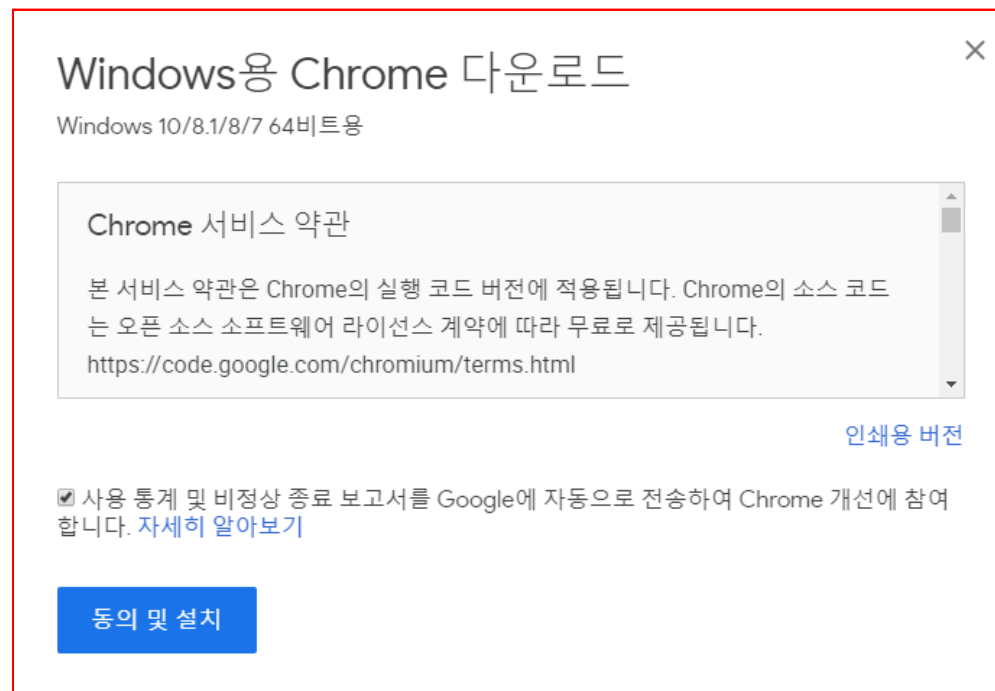
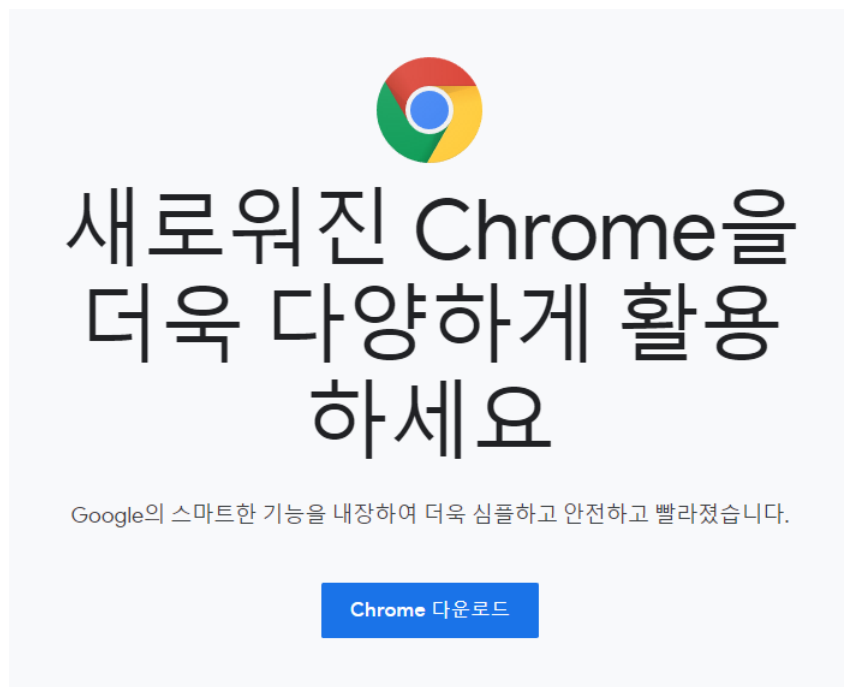
새로워진 Chrome을
더욱 다양하게 활용
하세요

Google의 스마트한 기능을 내장하여 더욱 심플하고 안전하고 빨라졌습니다.

Chrome 다운로드

구글 크롬 설치

- <https://www.google.com/chrome>



구글 크롬 설치

- <https://www.google.com/chrome>

Windows용 Chrome 다운로드

Windows 10/8.1/8/7 64비트용

Chrome 서비스 약관

본 서비스 약관은 Chrome의 실행 코드 버전에 적용됩니다. Chrome의 소스 코드는 오픈 소스 소프트웨어 라이선스 계약에 따라 무료로 제공됩니다.

<https://code.google.com/chromium/terms.html>

[인쇄용 버전](#)

☒ 사용 통계 및 비정상 종료 보고서를 Google에 자동으로 전송하여 Chrome 개선에 참여합니다. [자세히 알아보기](#)

동의 및 설치

구글 크롬 설치

- <https://www.google.com/chrome>

Windows용 Chrome 다운로드

Windows 10/8.1/8/7 64비트용

Chrome 서비스 약관

본 서비스 약관은 Chrome의 실행 코드 버전에 적용됩니다. Chrome의 소스 코드는 오픈 소스 소프트웨어 라이선스 계약에 따라 무료로 제공됩니다.

<https://code.google.com/chromium/terms.html>

[인쇄용 버전](#)

☒ 사용 통계 및 비정상 종료 보고서를 Google에 자동으로 전송하여 Chrome 개선에 참여합니다. [자세히 알아보기](#)

동의 및 설치



ChromeSetup.exe

구글 크롬 설치

- <https://www.google.com/chrome>

Windows용 Chrome 다운로드

Windows 10/8.1/8/7 64비트용

Chrome 서비스 약관


본 서비스 약관은 Chrome의 실행 코드 버전에 적용됩니다. Chrome의 소스 코드는 오픈 소스 소프트웨어 라이선스 계약에 따라 무료로 제공됩니다.

<https://code.google.com/chromium/terms.html>

[인쇄용 버전](#)

☒ 사용 통계 및 비정상 종료 보고서를 Google에 자동으로 전송하여 Chrome 개선에 참여합니다. [자세히 알아보기](#)

동의 및 설치

 ChromeSetup.exe

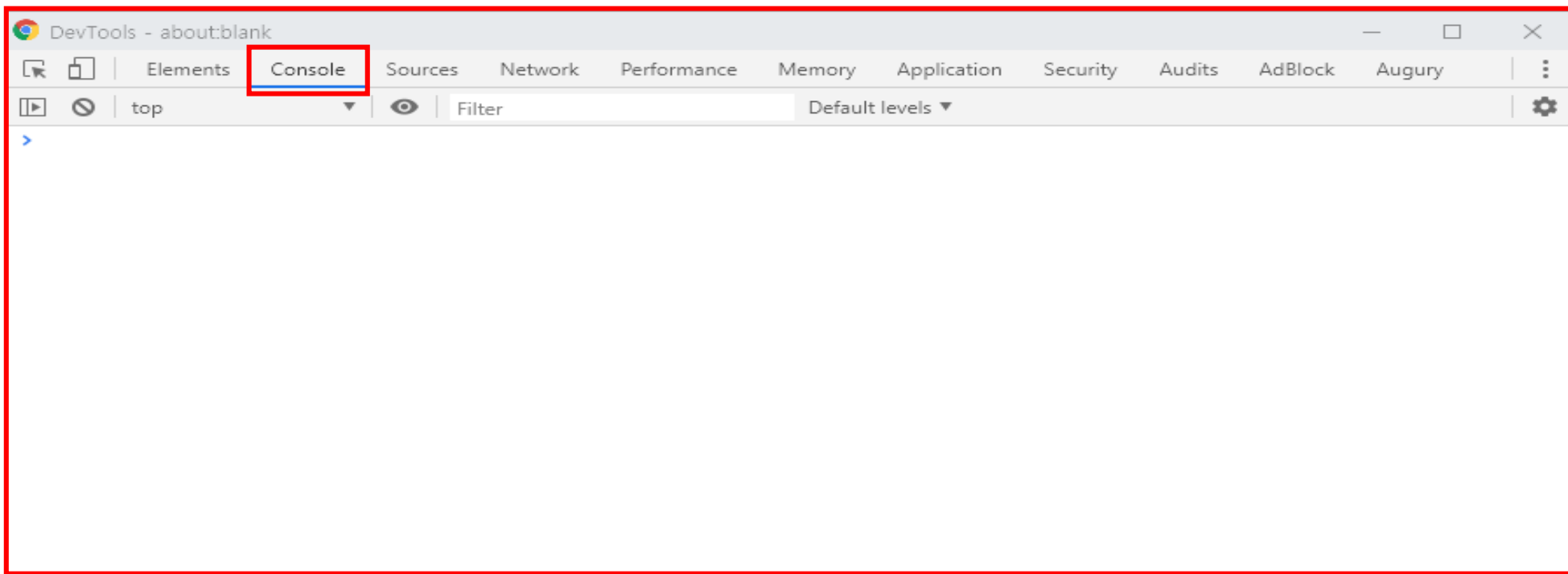
설치가 완료되었습니다.



닫기

구글 크롬 실행

- 크롬 실행 후 키보드의 F12 키 클릭



실습1 프로그램

- 프로그램 작성
- 줄을 바꾸려면 **Shift + Enter** 를 입력

```
1  var a = 10
```

```
2  
```

실습1 프로그램

```
1    var a = 10
```

```
2    var b = 3
```

```
3
```

실습1 프로그램

```
1  var a = 10
2  var b = 3
3  var c = (a + b) / 2
4  
```

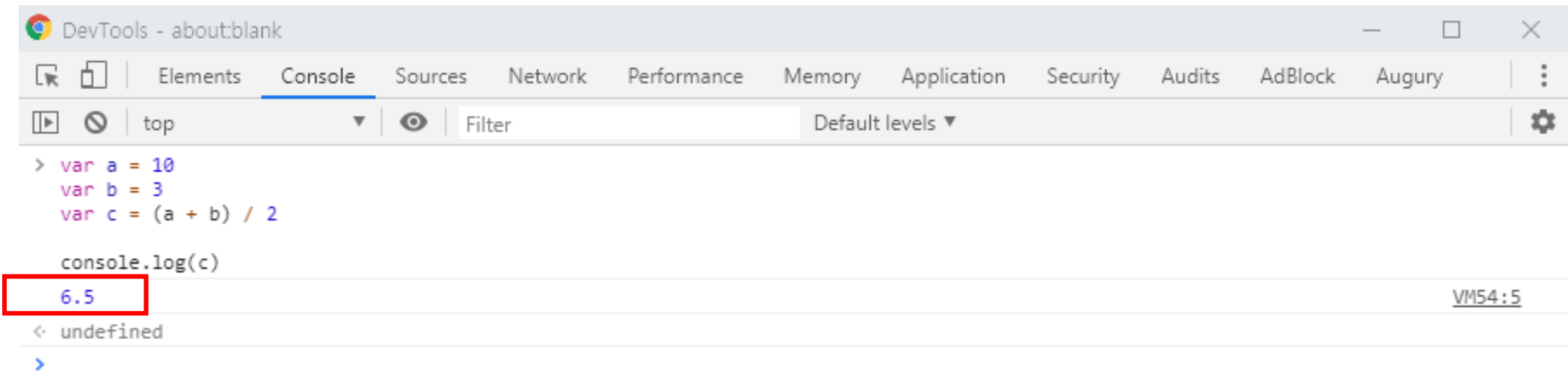
실습1 프로그램

```
1  var a = 10
2  var b = 3
3  var c = (a + b) / 2
4
5  console.log(c)
6
```

- Console.log(c) 입력 후 **enter** 클릭

실습1 프로그램

- Console.log(c) 입력 후 **enter** 클릭
- 결과 6.5



The screenshot shows the Chrome DevTools interface with the 'Console' tab selected. The console contains the following code and output:

```
> var a = 10  
  var b = 3  
  var c = (a + b) / 2  
  
  console.log(c)
```

The output of the `console.log(c)` statement is **6.5**, which is highlighted with a red box. The console also shows `< undefined` and a blue prompt character `>` on the next line. The right side of the console indicates the source as `VM54:5`.

변수 사용하기

- 변수란?
 - 어떤 값을 저장하는 임시저장 공간
 - 프로그램에서 값을 저장해야 할 때 사용

변수 사용하기

- 변수란?
 - 어떤 값을 저장하는 임시저장 공간
 - 프로그램에서 값을 저장해야 할 때 사용
- 형식
 - var {변수명} = {저장할 값}

변수 사용하기

- 변수란?
 - 어떤 값을 저장하는 임시저장 공간
 - 프로그램에서 값을 저장해야 할 때 사용
- 형식
 - `var {변수명} = {저장할 값}`
- 값 출력
 - `console.log({출력할 값})` 입력 후 Enter

변수 실습예제1

```
1  var name = '자바스크립트'
```

```
2
```

- name 이라는 변수에 '자바스크립트' 라는 문자를 저장

변수 실습예제1

```
1  var name = '자바스크립트'
2  console.log(name)
3
```

- console.log(name) 입력 후 Enter
- name을 출력하는 것이 아니라 name 변수에 저장되어 있던 '자바스크립트' 라는 문자를 출력

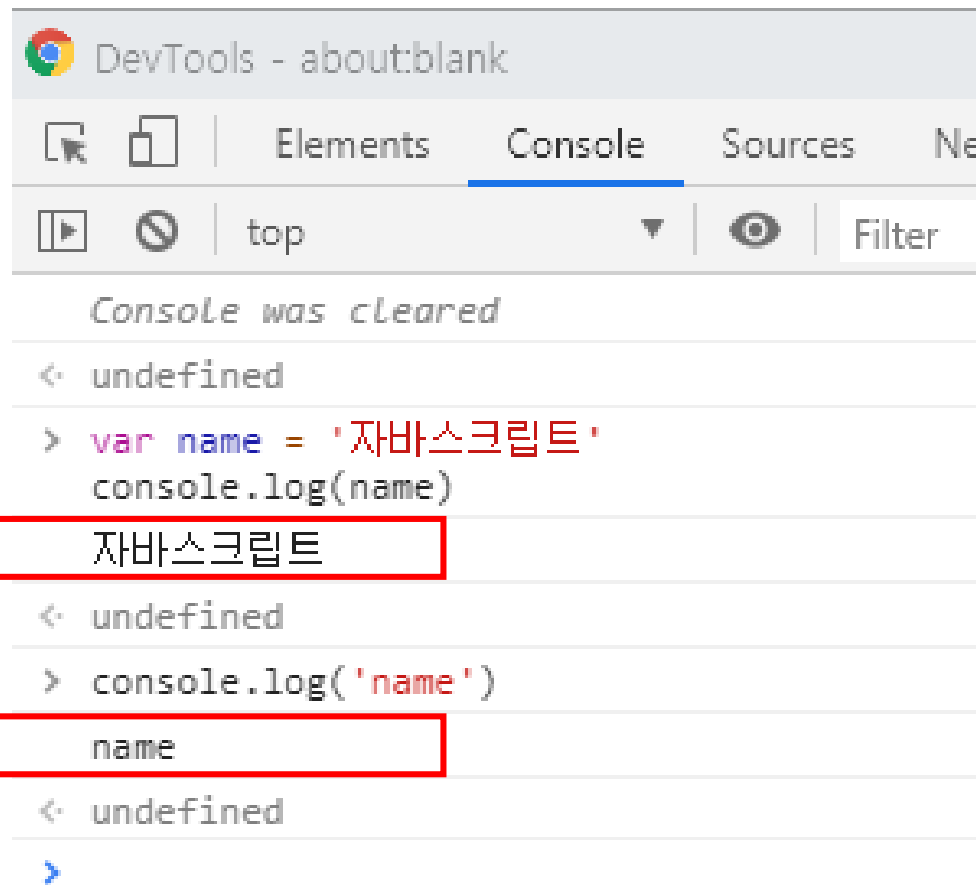
변수 실습예제1

```
1  var name = '자바스크립트'
2  console.log(name)
3  console.log('name')
4
```

- name 글자를 출력하기 위해서는 console.log('name') 처럼 따옴표를 사용해야 함

변수 실습예제1

실행결과



변수 실습예제2

- 변수의 값 언제든지 변경 가능

```
1  var a = 0
```

```
2  
```


변수 실습예제2

- 변수의 값 언제든지 변경 가능

```
1  var a = 0
2  a = 10 - 4
3  
```

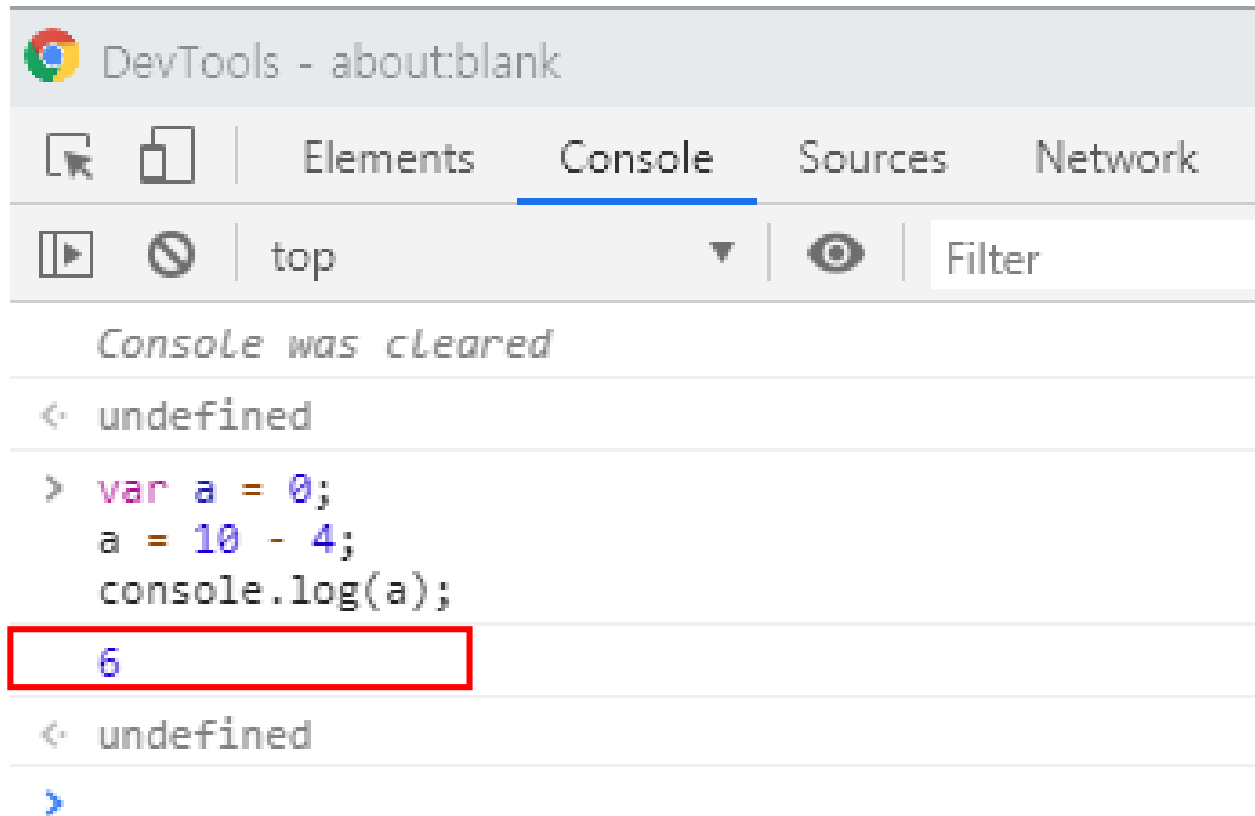
변수 실습예제2

- 변수의 값 언제든지 변경 가능

```
1  var a = 0
2  a = 10 - 4
3  console.log(a)
4  
```

변수 실습예제2

- 실행 결과



사칙 연산자

- 자바스크립트의 사칙연산
 - 덧셈, 뺄셈, 곱셈, 나눗셈
- 연산자 기호
 - + : 더하기
 - - : 빼기
 - * : 곱하기
 - / : 나누기

연산 순서

- 연산 순서
 - 수학과 동일
 - 덧셈, 뺄셈보다는 곱셈 나눗셈 먼저 실행
 - 왼쪽에서 오른쪽으로 계산
 - 중괄호, 대괄호 등은 구분하지 않고 (,) 만 사용

// 다음은
주석문으로
코딩으로
인식하지 않음

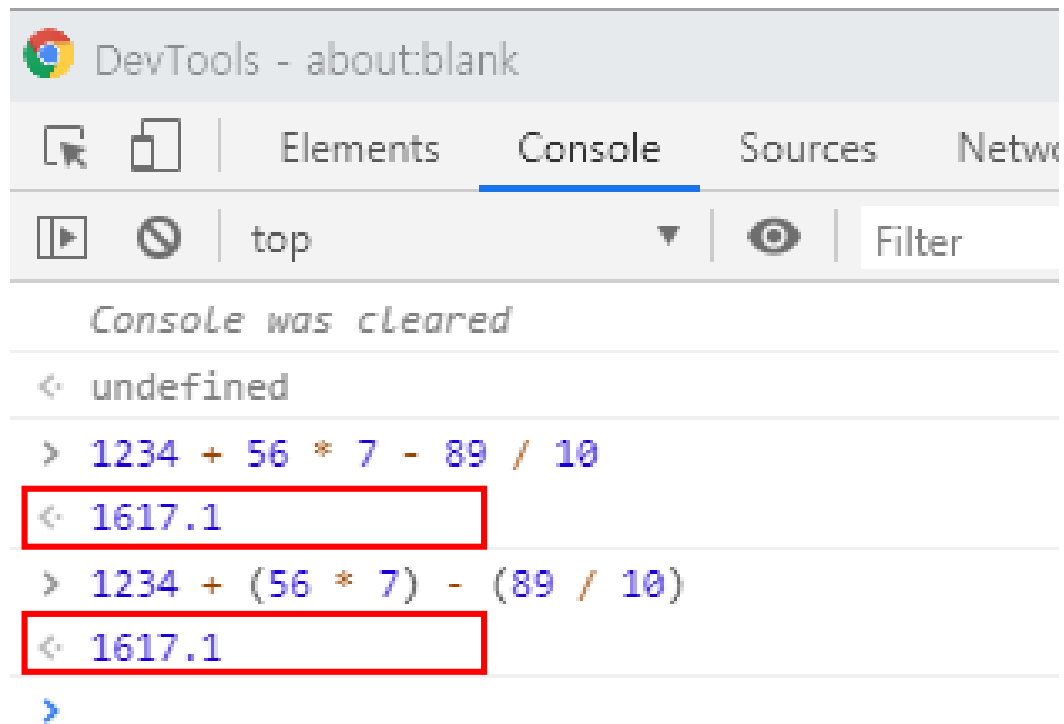
```
1 1234 + 56 * 7 - 89 / 10
```

```
2 1234 + (56 * 7) - (89 / 10) // 같은 결과
```

```
3
```

연산 순서

- 실행결과 동일함



The screenshot shows the Chrome DevTools Console with the 'Console' tab selected. The console displays the following sequence of commands and results:

```
Console was cleared
< undefined
> 1234 + 56 * 7 - 89 / 10
< 1617.1
> 1234 + (56 * 7) - (89 / 10)
< 1617.1
>
```

The two results, `1617.1`, are highlighted with red rectangular boxes to demonstrate that the two different expressions yield the same result.

증가/감소 연산자

- ++, -- 연산자
 - 변수의 값을 1 증가/감소 시킴

```
1  var a = 10
```

```
2  
```

증가/감소 연산자

- ++, -- 연산자
 - 변수의 값을 1 증가/감소 시킴

```
1  var a = 10
```

```
2  ++a
```

```
3  
```


증가/감소 연산자

- ++, -- 연산자
 - 변수의 값을 1 증가/감소 시킴

```
1  var a = 10
```

```
2  ++a
```

```
3  --a
```

```
4  
```

증가/감소 연산자

- 실행결과

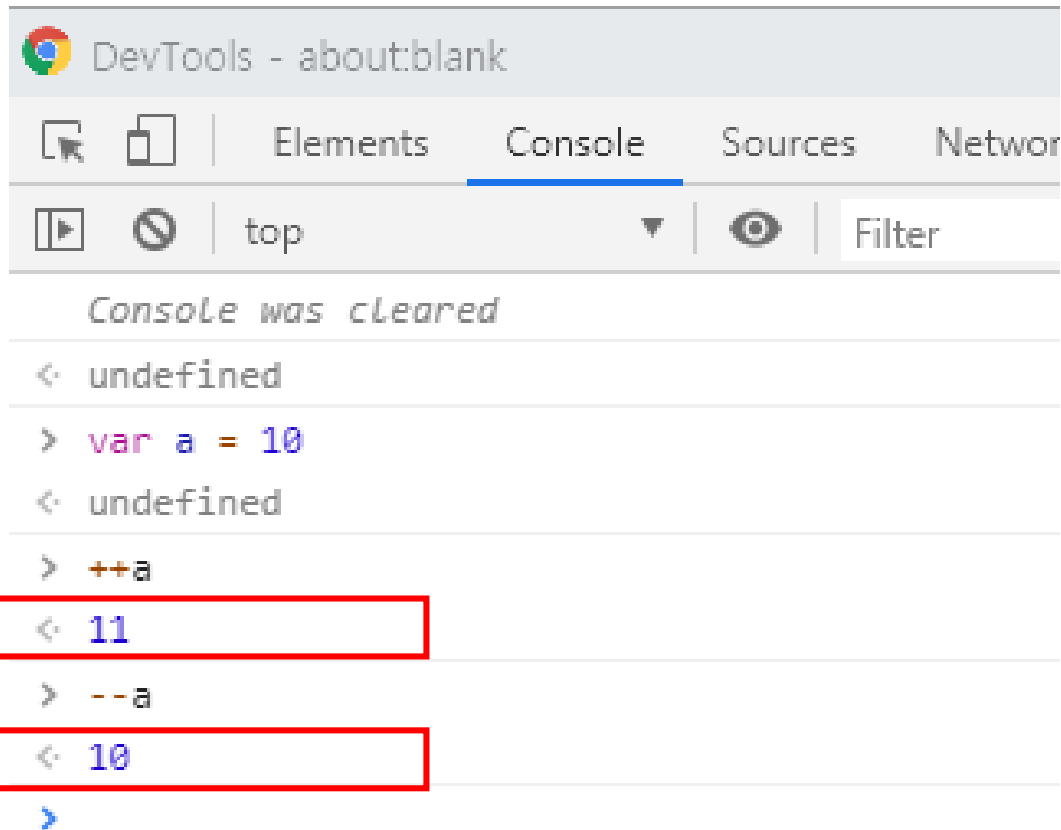
a=10

++a //a=a+1

//a=11

--a //a=a-1

//a=10



`+=, -=` 연산자

- `+=, -=` 연산자
 - 변수의 값을 원하는 만큼 증가/감소 시킴
 - `a += 5` 는 `a = a + 5` 과 같은 결과

```
1  var a = 10
```

```
2  
```

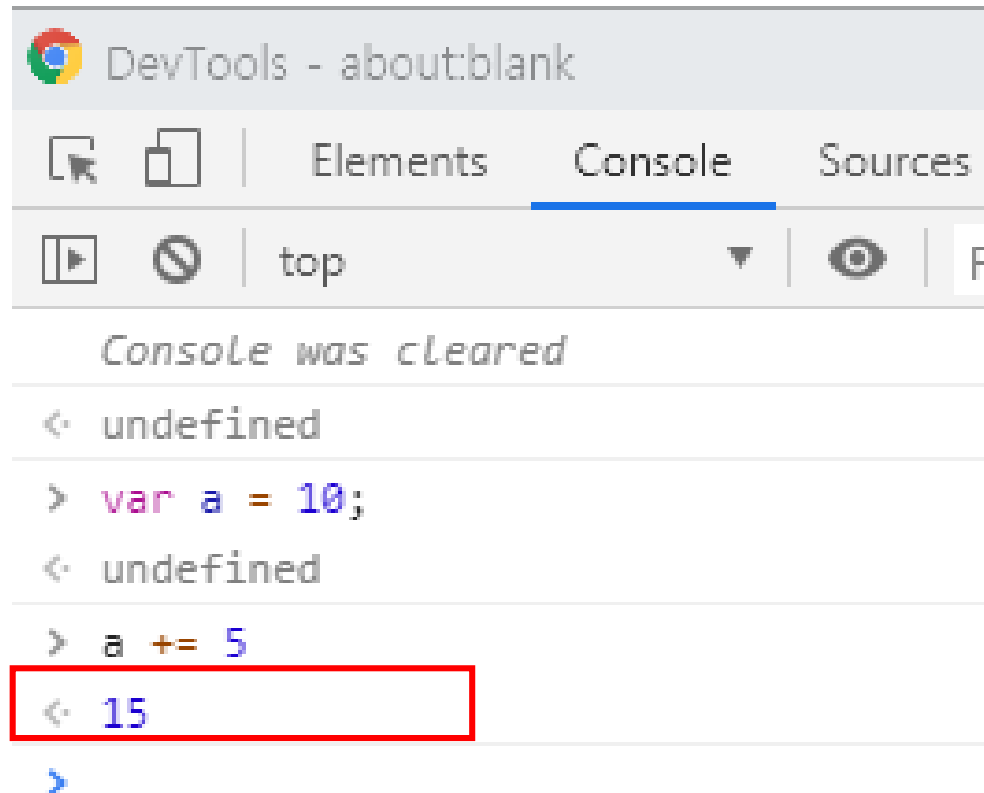
`+=, -=` 연산자

- `+=, -=` 연산자
 - 변수의 값을 원하는 만큼 증가/감소 시킴
 - `a += 5` 는 `a = a + 5` 과 같은 결과

```
1  var a = 10
2  a += 5
```

$+=$, $-=$ 연산자

- 실행결과



The screenshot shows the Chrome DevTools Console interface. The 'Console' tab is selected. The console log shows the following sequence of events:

- Console was cleared*
- Input: `<` undefined
- Input: `> var a = 10;`
- Output: `< undefined`
- Input: `> a += 5`
- Output: `< 15` (This line is highlighted with a red box)
- Input: `>`

$\ast=$, $/=$ 연산자

- $a \ast= 5$ 는 $a = a \ast 5$ 동일
- $a /= 2$ 는 $a = a / 5$ 동일

```
1  var a = 10
```

`*=`, `/=` 연산자

- `a *= 5` 는 `a = a * 5` 동일
- `a /= 2` 는 `a = a / 2` 동일

```
1  var a = 10
2  a *= 5
```

`*=`, `/=` 연산자

- `a *= 5` 는 `a = a * 5` 동일
- `a /= 2` 는 `a = a / 2` 동일

```
1  var a = 10
2  a *= 5
3  a /= 2
```

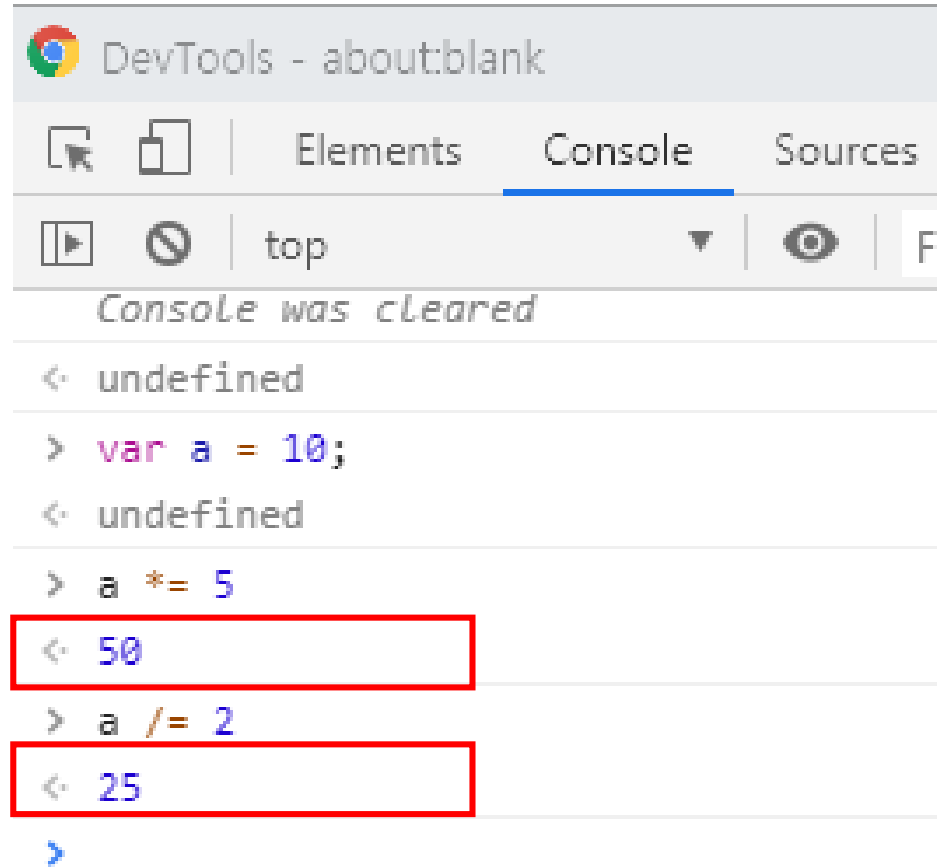

$\ast=$, $/=$ 연산자

- 실행 결과

`a=10`

`a*=5 // a=a*5`
`// a=50`

`a/=2 // a=a/2`
`// a=25`



문자열 합치기

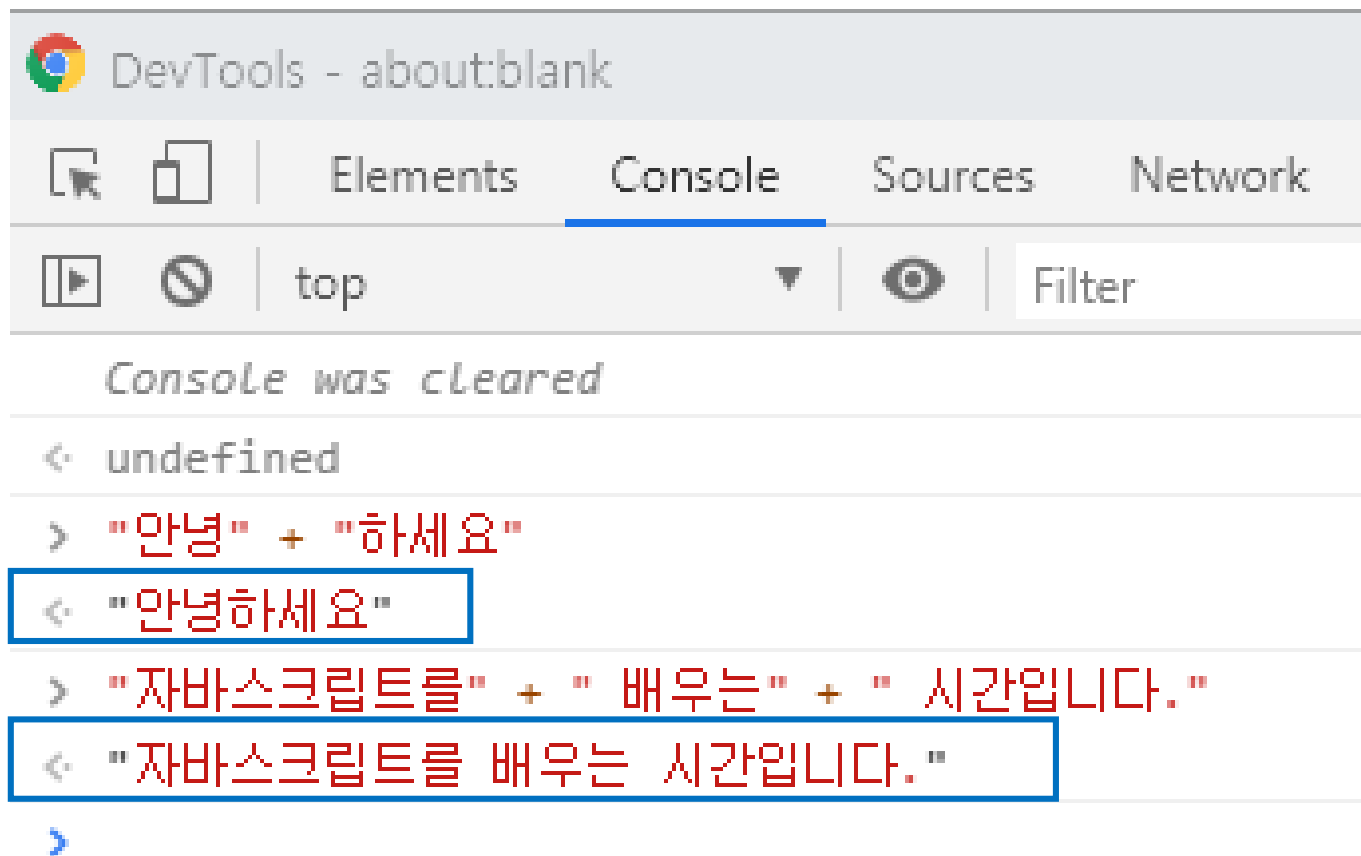
- 문자열 합치기 연산 가능
 - + 연산자로 문자도 연결 가능
 - + 연산자 여러 번 사용 가능

```
1 '안녕' + '하세요'
```

```
2 '자바스크립트를' + ' 배우는' + ' 시간입니다.'
```

문자열 합치기

- 실행결과



연산자와 자료형

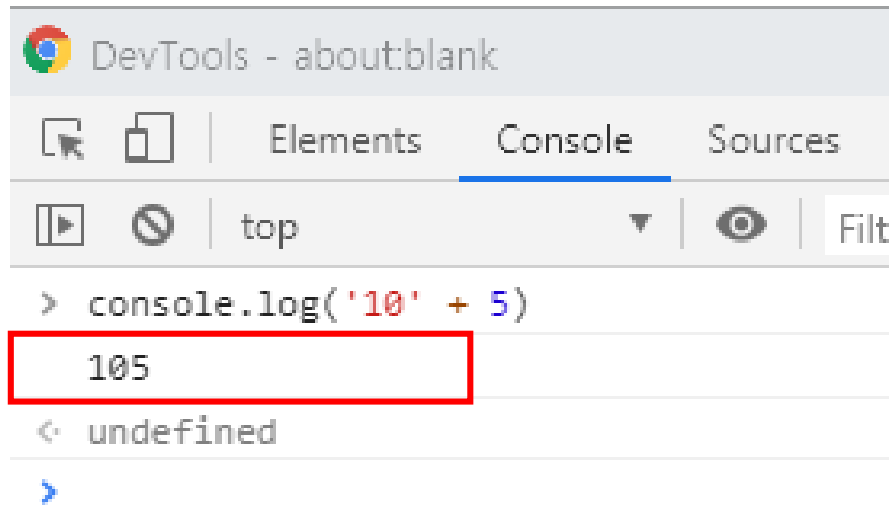
- “10” + 5 // 문자 + 숫자

```
1  console.log('10' + 5)
```

```
2  
```

연산자와 자료형

- 실행결과 - “10” 은 문자 1 과 0으로 인식



DevTools - about:blank

Elements Console Sources

top

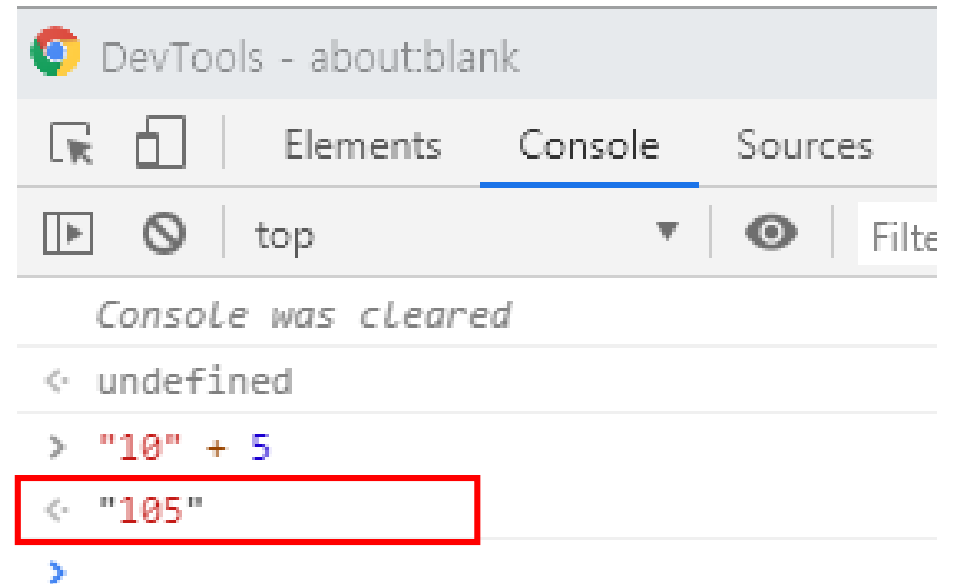
```
> console.log('10' + 5)
```

105

< undefined

>

Detailed description: This screenshot shows the Chrome DevTools Console. The 'Console' tab is selected. A command `console.log('10' + 5)` has been entered. The output is `105`, which is highlighted with a red rectangular box. Below the output, there is a prompt `< undefined` and a blue arrow indicating the next step.



DevTools - about:blank

Elements Console Sources

top

Console was cleared

```
< undefined
```

```
> "10" + 5
```

< "105"

>

Detailed description: This screenshot shows the Chrome DevTools Console after it has been cleared. The 'Console' tab is selected. The text 'Console was cleared' is displayed. Below it, there is a prompt `< undefined`. A new command `"10" + 5` has been entered. The output is `"105"`, which is highlighted with a red rectangular box. Below the output, there is a prompt `< "105"` and a blue arrow indicating the next step.

연산자의 종류

- 비교 연산자
 - 두 값의 대소 비교
- 논리 연산자
 - 논리곱(AND)
 - 논리합(OR)
 - 부정(NOT)

불리언(Boolean)

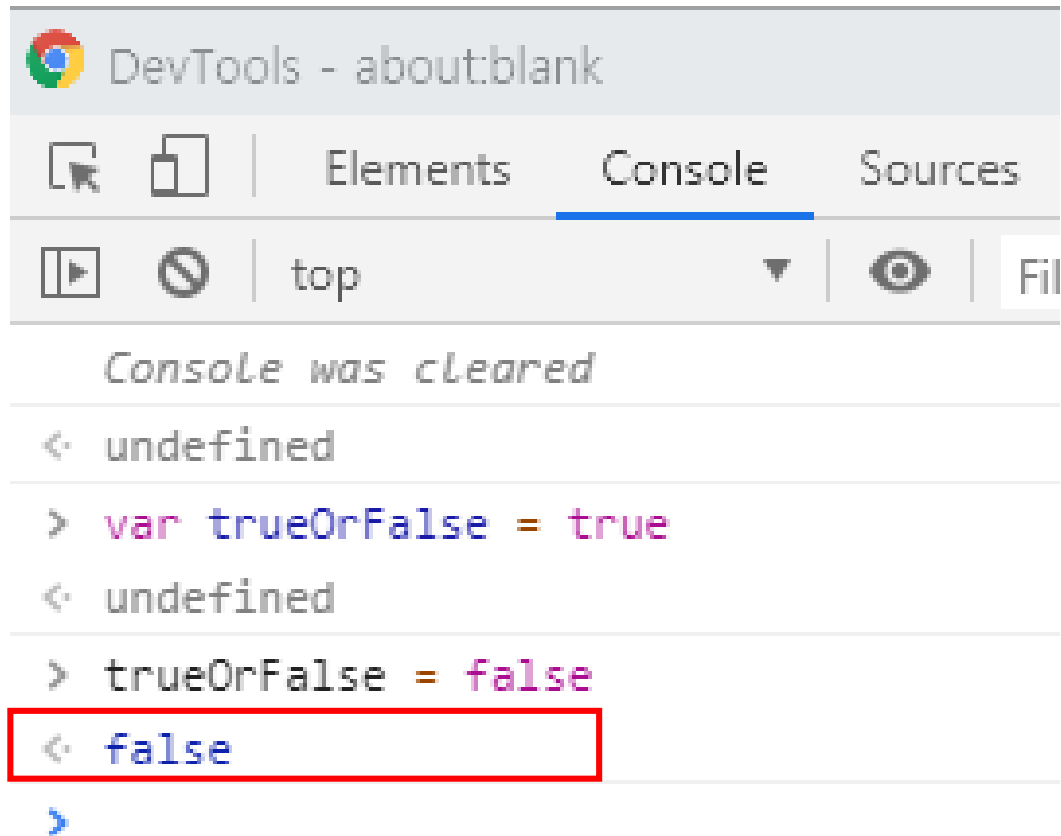
- 참이나 거짓을 표현
- 참은 true, 거짓은 false

```
1  var tureOrFalse = true
2  tureOrFalse = false
3  
```

불리언(Boolean)

- 실행결과

```
var trueOrFalse = true  
trueOrFalse = false  
  //true = false  
  //같지 않으므로 false
```



비교연산자

- 같다, 다르다: ==, !=
- 초과, 미만: >, <
- 이상, 이하: >=, <=

```
1  var speed = 100
```

```
2
```

비교연산자

- 같다, 다르다: ==, !=
- 초과, 미만: >, <
- 이상, 이하: >=, <=

```
1  var speed = 100
2  speed == 120
3  
```

비교연산자

- 같다, 다르다: ==, !=
- 초과, 미만: >, <
- 이상, 이하: >=, <=

```
1  var speed = 100
2  speed == 120
3  speed != 0
4  
```

비교연산자

- 실행결과

```
var speed=100
```

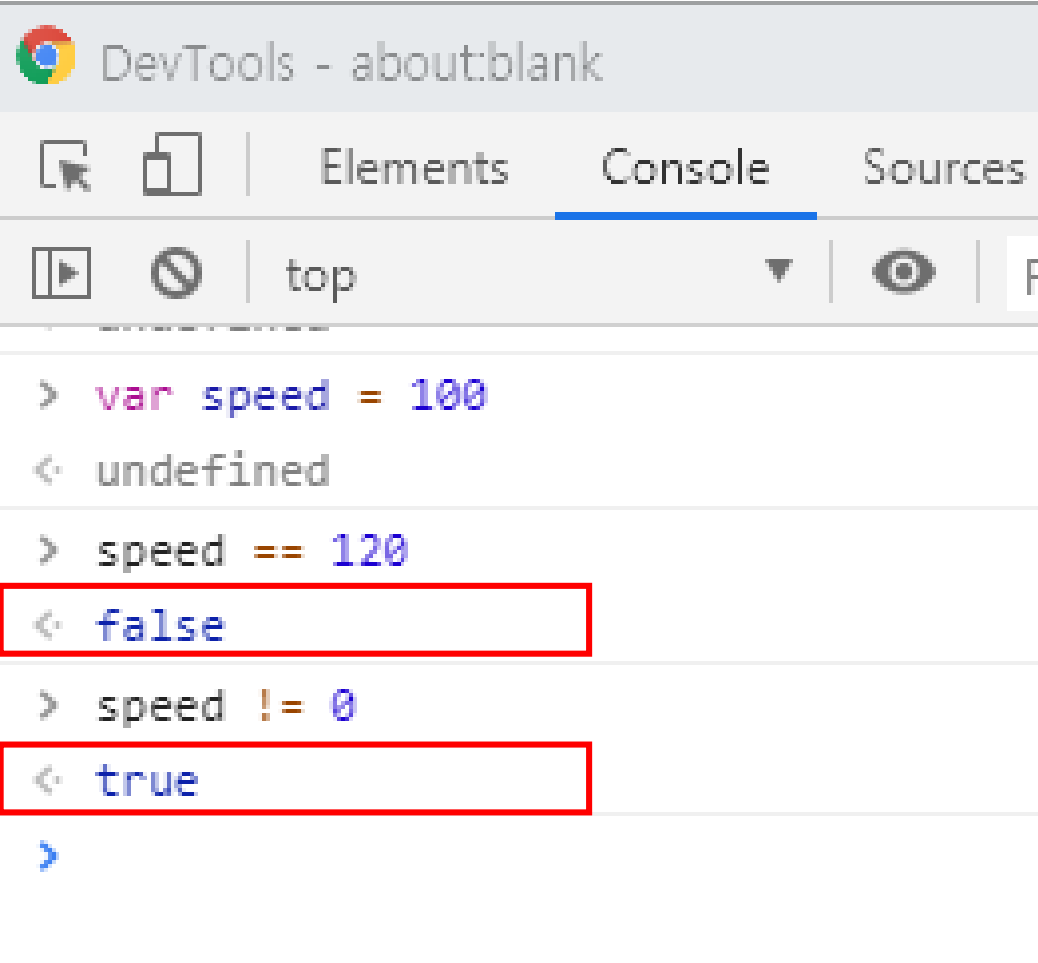
```
speed == 120
```

```
//100 == 120
```

```
//같지 않으므로 false
```

```
speed != 0
```

```
//다르므로 true
```



The screenshot shows the Chrome DevTools Console with the 'Console' tab selected. The title bar reads 'DevTools - about:blank'. The breadcrumb navigation shows 'top'. The console log contains the following entries:

- A prompt character followed by the command `var speed = 100`.
- A return value of `undefined`.
- A prompt character followed by the command `speed == 120`.
- A return value of `false`, which is highlighted with a red rectangular box.
- A prompt character followed by the command `speed != 0`.
- A return value of `true`, which is also highlighted with a red rectangular box.
- A final prompt character `>` at the bottom.

비교연산자

- 같다, 다르다: ==, !=
- 초과, 미만: >, <
- 이상, 이하: >=, <=

```
1  var speed = 100
2  speed < 120
3  
```

비교연산자

- 같다, 다르다: ==, !=
- 초과, 미만: >, <
- 이상, 이하: >=, <=

```
1  var speed = 100
2  speed < 120
3  speed > 110
4
```

비교연산자

- 실행결과

```
var speed=100
```

```
speed < 110
```

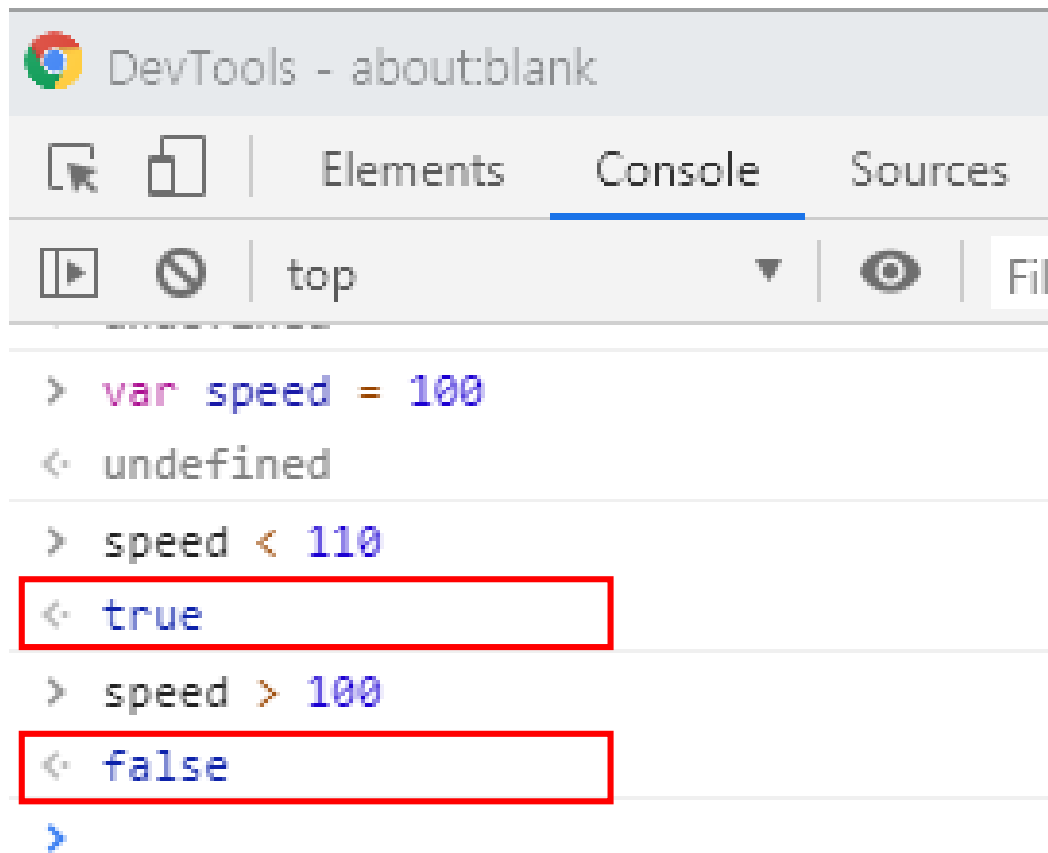
```
//100 < 110
```

```
//true
```

```
speed > 100
```

```
//100 > 100
```

```
//false
```



비교연산자

- 속도 위반 검사하기

```
1  var speed = 80
```

```
2  
```


비교연산자

- 속도 위반 검사하기

```
1  var speed = 80
2  var speedLimit = 100
3  
```

비교연산자

- 속도 위반 검사하기

```
1  var speed = 80
2  var speedLimit = 100
3  var speedViolation = speed >= speedLimit
4  
```

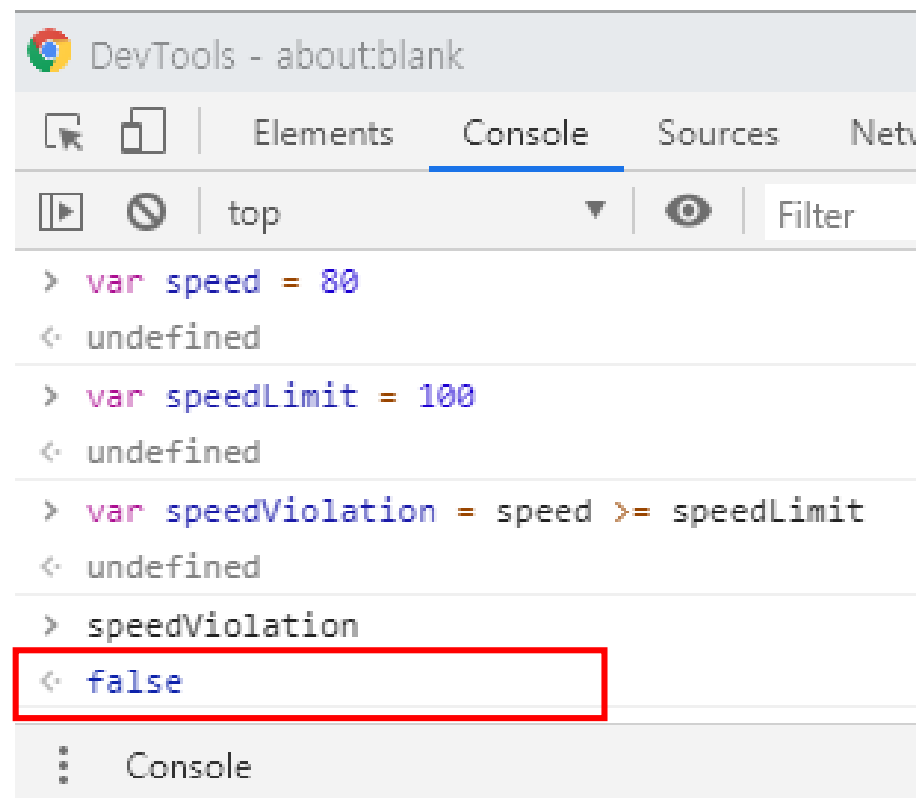
비교연산자

- 실행결과

```
var speedViolation = speed >= speedLimit
```

```
// speedViolation = 80 >= 100
```

```
// speedViolation = false
```



논리연산자 AND

- && (AND)
- 양쪽 피연산자 모두 true 이면 true

```
1  var a = true
2  var b = false
3  var c = true
4  
```

논리연산자 AND

- && (AND)
- 양쪽 피연산자 모두 true 이면 true

```
1  var a = true
2  var b = false
3  var c = true
4  a && b
5  
```

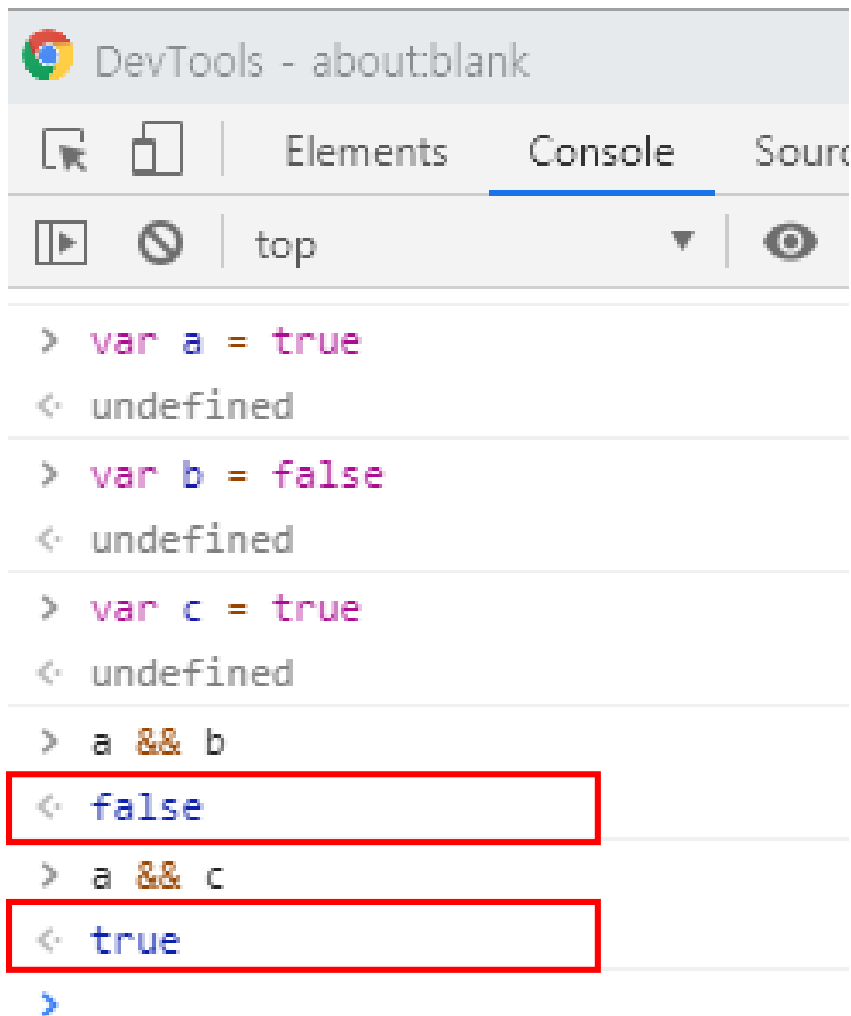
논리연산자 AND

- && (AND)
- 양쪽 피연산자 모두 true 이면 true

```
1  var a = true
2  var b = false
3  var c = true
4  a && b
5  a && c
```

논리연산자 AND

- 실행결과



The screenshot shows the Chrome DevTools Console with the 'Console' tab selected. The breadcrumb navigation shows 'top'. The console contains the following code and output:

```
> var a = true
< undefined

> var b = false
< undefined

> var c = true
< undefined

> a && b
< false

> a && c
< true

>
```

The results `false` and `true` are highlighted with red boxes.

논리연산자 OR

- || (OR)
- 양쪽 피연산자 중 하나가 true 이면 true

```
1  var a = true
2  var b = false
3  var c = true
4  a || b
5  
```

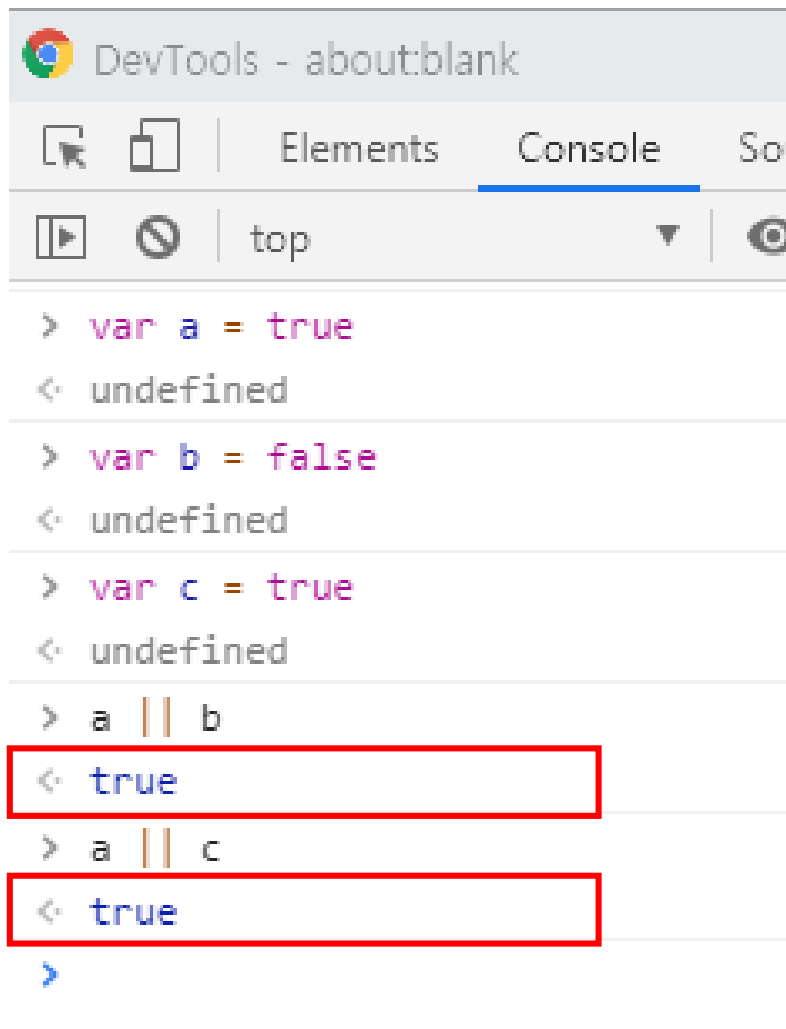

논리연산자 OR

- || (AND)
- 양쪽 피연산자 중 하나가 true 이면 true

```
1  var a = true
2  var b = false
3  var c = true
4  a || b
5  a || c
```

논리연산자 OR

- 실행결과



The screenshot shows the Chrome DevTools Console with the 'Console' tab selected. The breadcrumb shows 'top'. The console contains the following commands and their results:

```
> var a = true
< undefined

> var b = false
< undefined

> var c = true
< undefined

> a || b
< true

> a || c
< true

>
```

The results 'true' for the two OR operations are highlighted with red boxes.

논리연산자

- ! (NOT)
- true 이면 false, false 이면 true

```
1  var a = true
2  var b = false
3  !a
4  
```

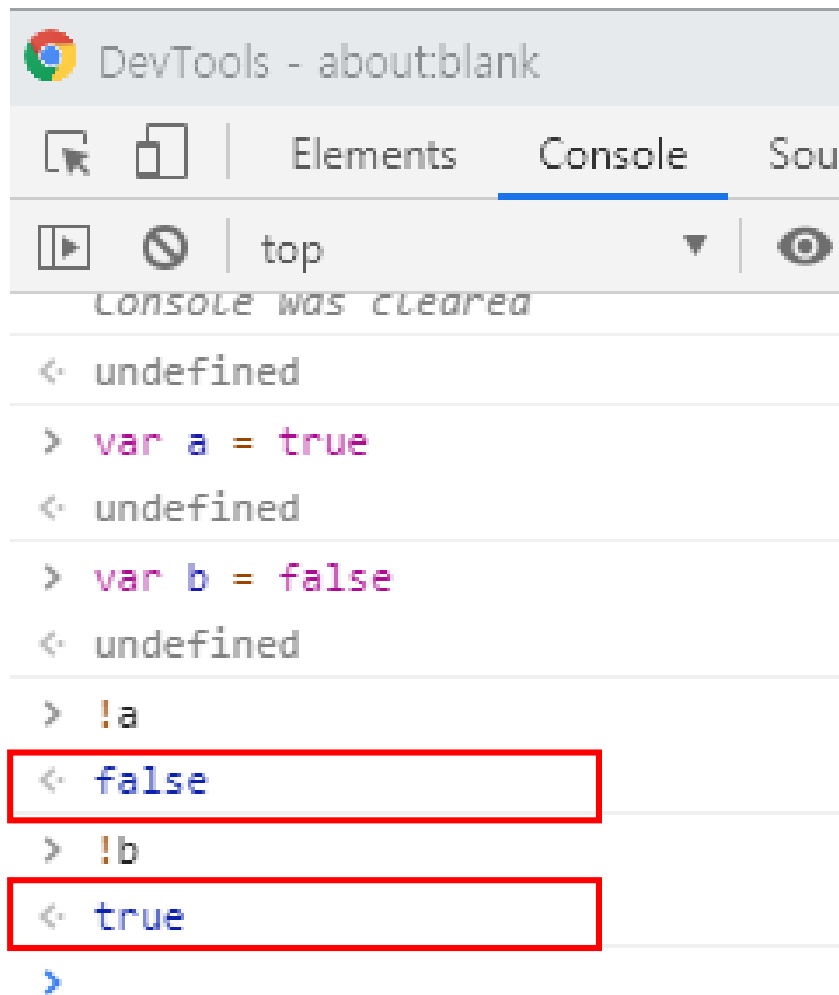
논리연산자 NOT

- ! (NOT)
- true 이면 false, false 이면 true

```
1  var a = true
2  var b = false
3  !a
4  !b
5  
```

논리연산자 NOT

- 실행결과



The screenshot shows the Chrome DevTools Console for an about:blank page. The 'Console' tab is selected. The console output shows the following sequence of commands and results:

```
Console was cleared  
< undefined  
> var a = true  
< undefined  
> var b = false  
< undefined  
> !a  
< false  
> !b  
< true  
>
```

The results `false` and `true` are highlighted with red boxes.

요약

- 자바스크립트 소개
- 변수의 정의
- 변수 사용하기
- 연산자 종류
- 연산자 활용하기

차시 예고

- 7-2 : 자바스크립트 배열 알아보기

- 배열의 정의
- 강의 수강생 목록 예제
- 네비게이션 만들기 예제
- 복불복 게임 만들기 예제

강의를 마치겠습니다
수고하셨습니다

7주차_01 자바스크립트 변수와 연산자