

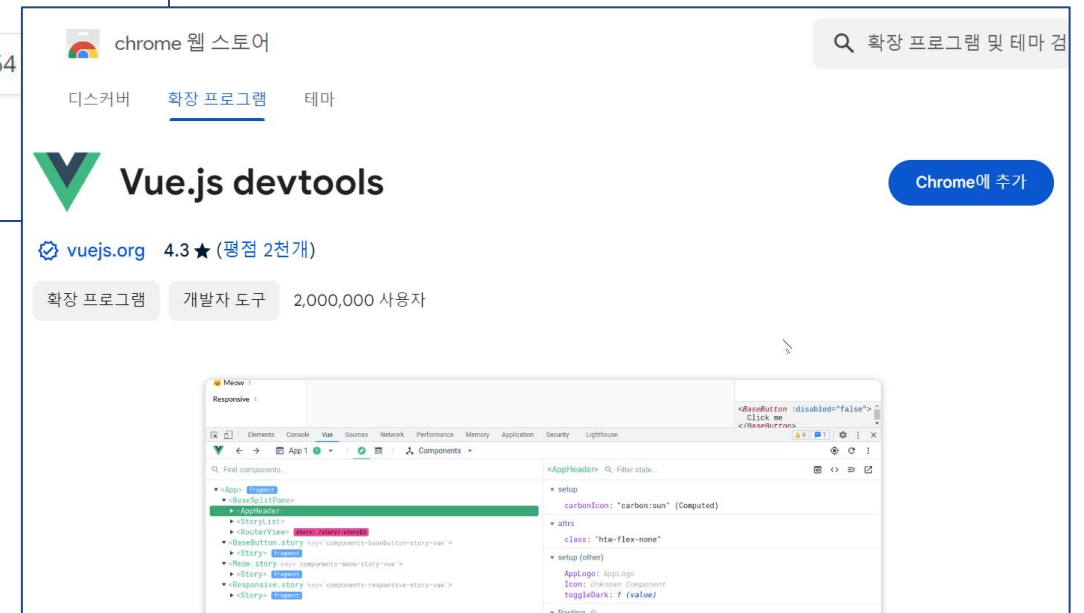
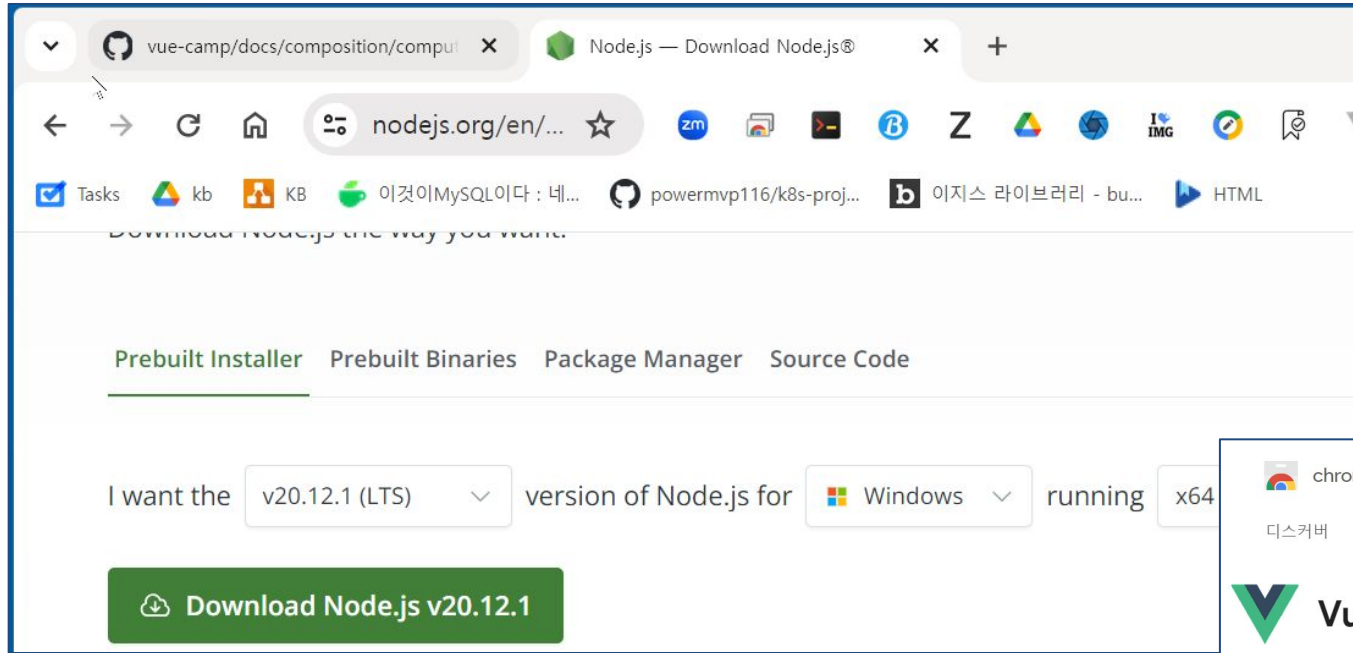
2024년 상반기 K-디지털 트레이닝

# ES6

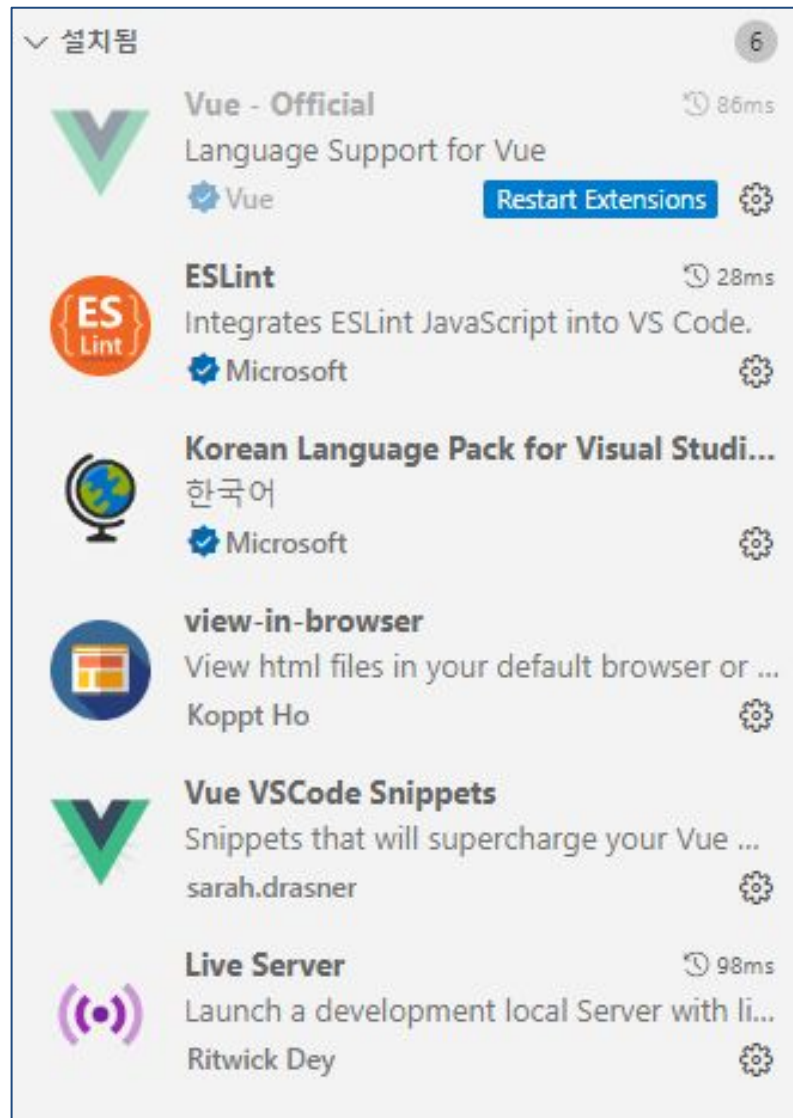
---

[KB] IT's Your Life

- nodeJS와 vueJS 크롬 확장 플러그인 설치하시오.



- vsCode에 플러그인 설치하시오.



- 프로젝트 생성 후, 브라우저에서 다음을 테스트 하시오.
- 프로젝트 명: **start-app**
- 파일 명: **hello-vue.html**

The screenshot displays a web development environment with three main components:

- Code Editor (Left):** Shows the source code for `hello-vue.html`. The code includes a DOCTYPE declaration, HTML5 boilerplate, a Vue.js script, and a Vue application instance. The application has a data property `message` set to "Hello Vue3".
 

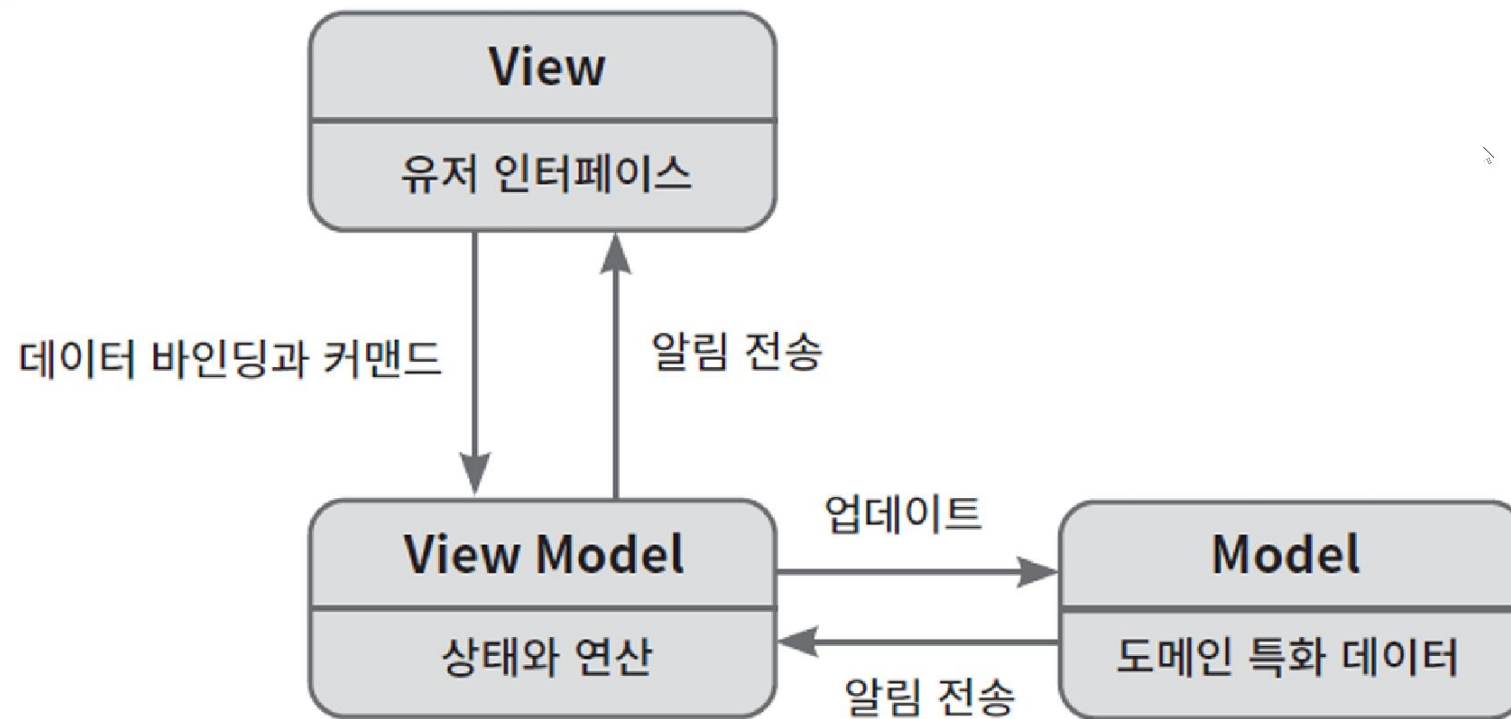
```

1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1">
6   <title>Document</title>
7 </head>
8 <body>
9   <div id="app">
10     <h2>{{ message }}</h2>
11   </div>
12   <script src="https://unpkg.com/vue"></script>
13   <script>
14     var model = {
15       message : "Hello Vue3"
16     };
17
18     var vm = Vue.createApp({
19       name : "App",
20       data() {
21         return model;
22       }
23     }).mount('#app')
24   </script>
25 </body>
26 </html>
      
```
- Browser (Top Right):** Displays the rendered output "Hello Vue3". The Vue.js component inspector shows the component is `<App>` version 3.4.21, with a render time of 18.4 ms. The data property is visible with the value `message: "Hello Vue3"`.
- Context Menu (Bottom Center):** A right-click menu is open over the code editor, showing various actions and their keyboard shortcuts. The "Open with Live Server" option is highlighted.
 

정의로 이동	F12
참조로 이동	Shift+F12
피킹	
모든 참조 찾기	Shift+<Alt>+F12
기호 이름 바꾸기	F2
모든 항목 변경	Ctrl+F2
문서 서식	Shift+<Alt>+F
리팩터링...	Ctrl+Shift+R
소스 작업...	
잘라내기	Ctrl+X
복사	Ctrl+C
붙여넣기	Ctrl+V
<b>Open with Live Server</b>	<b>&lt;Alt&gt;+L &lt;Alt&gt;+O</b>
Stop Live Server	<Alt>+L <Alt>+C
View In Browser	
View In Other Browsers	
명령 팔레트...	Ctrl+Shift+P
- Settings Panel (Bottom Right):** The "More settings" menu is open, showing options like "Devtools plugins", "Documentation", "Report a bug", and "What's new".

- 다음을 간단히 설명하시오.

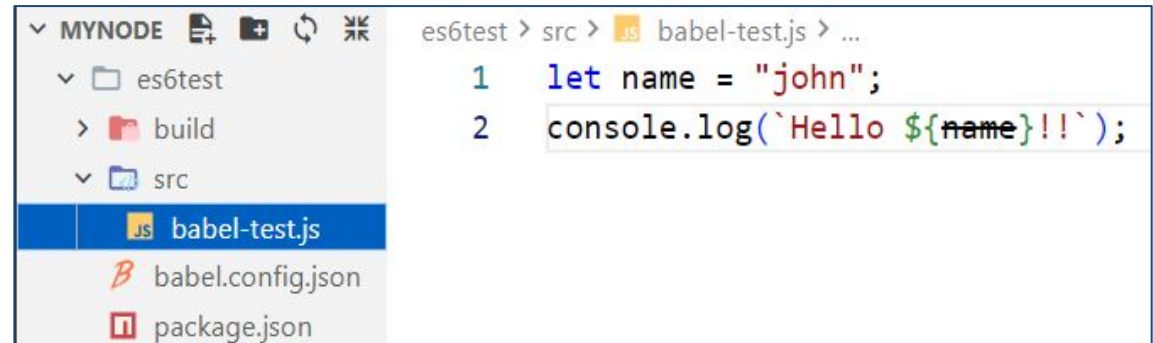
## MVVM



- 다음 es6test 프로젝트를 생성하고, babel을 설정하고 테스트하시오.

```
1  {
2    "name": "test",
3    "version": "1.0.0",
4    "description": "",
5    "main": "index.js",
6    "scripts": {
7      "test": "echo \"Error: no test specified\" && exit 1",
8      "build": "babel src -d build"
9    },
10   "author": "",
11   "license": "ISC",
12   "devDependencies": {
13     "@babel/cli": "^7.24.5",
14     "@babel/core": "^7.24.5",
15     "@babel/preset-env": "^7.24.5",
16     "core-js": "^3.37.0"
17   }
18 }
19
```

- npm init
- npm install --save-dev @babel/core @babel/cli @babel/preset-env core-js
- babel.config.json
- src/test.js
- package.json에  
"build": "babel src -d build"
- npm run build 또는 npx babel src -d build
- node ../build/test.js



```
es6test > src > babel-test.js > ...
1  let name = "john";
2  console.log(`Hello ${name}!!`);
```

- 다음 문제를 해결할 개선된 코드를 제시하시오.

```
console.log(A1);  
var A1 = "hello";
```

브라우저에서 이러한  
자바스크립트 코드가 가능한가?  
왜 그런가?

```
var A1 = 100;  
console.log(A1);  
var A1 = "hello";  
console.log(A1);
```

가능한 코드인가? 왜  
그런가?



## Q4- 가변 파라미터

- 다음과 같이 가변적인 입력값을 허용할 수 있는 함수를 정의하여, 프로젝트를 빌드하고 다음과 같이 출력되도록 실행하시오.

```
addContact("홍길동", "010-222-3331")
addContact("이몽룡", "010-222-3331", "02-3422-9900", "서울시")
addContact("이몽룡", "010-222-3331", "02-3422-9900", "서울시", "hero@email.com", "감자", "고구마")
```

```
administrator@MacBook-Pro es6test % node build/basic-q4-param.js
name >> 홍길동
home >> 없음
address >> 없음
email >> 없음
favoritesFoods >>
-----
name >> 이몽룡
home >> 02-3422-9900
address >> 서울시
email >> 없음
favoritesFoods >>
-----
name >> 이몽룡
home >> 02-3422-9900
address >> 서울시
email >> hero@email.com
favoritesFoods >> 감자,고구마
```

- 프로젝트 초기화  
`npm init`
- 소스코드 작성  
`src/test.js`
- 소스코드 실행  
`node src/test.js`



## Q5 - 구조 분할 할당(다음 페이지에 이어)

- addContact3()함수를 정의하시오.
- 입력하지 않으면 email은 “이메일없음”, age는 0으로 설정

```

name>> 지수
phone>> 010-222-3333
email>> hera
next_age>> 101
-----
name2>> 지수
tel>> SKT 010-222-3333
contact_email>> hera@email.com
next_age>> 101

```

함수내 출력

리턴값 구조 분할 할당  
후 출력

```
let {name2, tel, contact_email, next_age} = addContact3("지수", "010-222-3333", "hera", 100);
```

## Q5 - 구조 분할 할당(다음 페이지에 이어)

- addContact2()함수를 정의하시오.
- email을 입력하지 않았으면, "이메일 없음"으로 처리
- age를 입력하지 않았으면, 0으로 처리

```
name>> 지 수
phone>> 010-222-3333
email>> hera
next_age>> 1
-----
name2>> 지 수
tel>> SKT 010-222-3333
contact_email>> hera@email.com
next_age>> 1
```

함수내 출력

리턴값 구조 분할 할당  
후 출력

```
let {name2, tel, contact_email, next_age} = addContact2({
  name : "지수",
  phone : "010-222-3333",
  email : "hera"
});
```

## Q6 - 화살표 함수

- 다음을 화살표 함수로 변경하고, 각각 호출하시오.

1

```
const test1 = function(a,b) {  
  return a+b;  
}
```

2

```
function test2(params) {  
  console.log(params + "님 환영합니다.!");  
}
```

3

```
function test3() {  
  return "ok";  
}
```

300

hong님 환영합니다.!

OK

## Q7 - 화살표 함수

- 다음의 **this**는 함수가 포함관계에 있을 때 객체가 전달되지 않는다. 해결책 2가지를 제시하고 프로그래밍 하여 값을 비교해보시오.

```
var obj = { result:0 };
obj.add = function(x,y) {
  function inner() {
    this.result = x+y;
  }
  inner();
}
obj.add(3,4)

console.log(obj)           // { result: 0 }
console.log(result)        // 7
```

- 다음 변수 값들을 객체 리터럴로 표현하여 객체의 각 값을 화면에 출력하시오.

```
var name = "홍길동";  
var age = 20;  
var email = "gdhong@test.com";
```

```
{ name: '홍길동', age: 20, email: 'sdhong@test.com' }  
name>> 홍길동  
age>> 20  
email>> sdhong@test.com
```



## Q8 - 객체 리터럴

- 쇼핑물의 물건을 객체 리터럴로 표현하시오.
- 예시 )
  - 속성 : 물건이름, 원가격, 할인가격, 1개당가격, 도착여부, 평점, 적립금
  - 메서드 : 익명 함수 형태(할인을 하다), 일반 함수 형태(물건 속성값을 프린트 하다.)

할인을 하는 중입니다 .

name>> 에너자이저

total\_price>> 5000

discount\_price>> 4700

one\_price>> 2350

arrive\_date>> 내일(수)도착보장

rate>> 50

stack\_money>> 188



에너자이저 2입

6% 5,000원

**4,700원**

(1개입당 2,350원)

로켓배송

내일(수) 도착 보장

★★★★★ (50)

최대 188원 적립



- 다음과 같이 출력되도록 완성하시오.

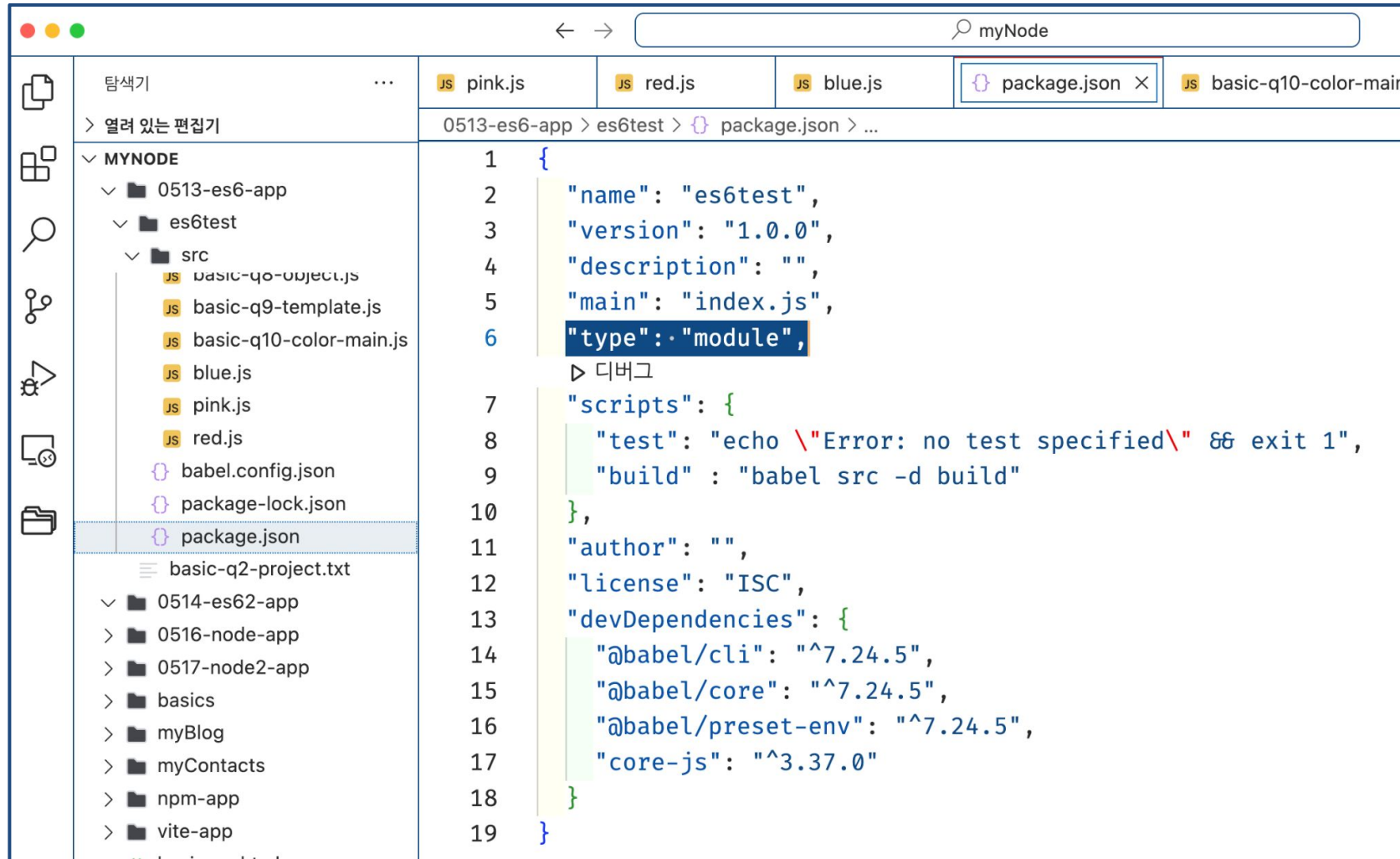
```
const d1 = new Date();  
let name = "홍길동";  
let product = "갤럭시s7";  
let price = 199000;
```

홍길동님에게 2024년 5월 5일에 연락했다.  
갤럭시 s7의 가격은 199,000원입니다.

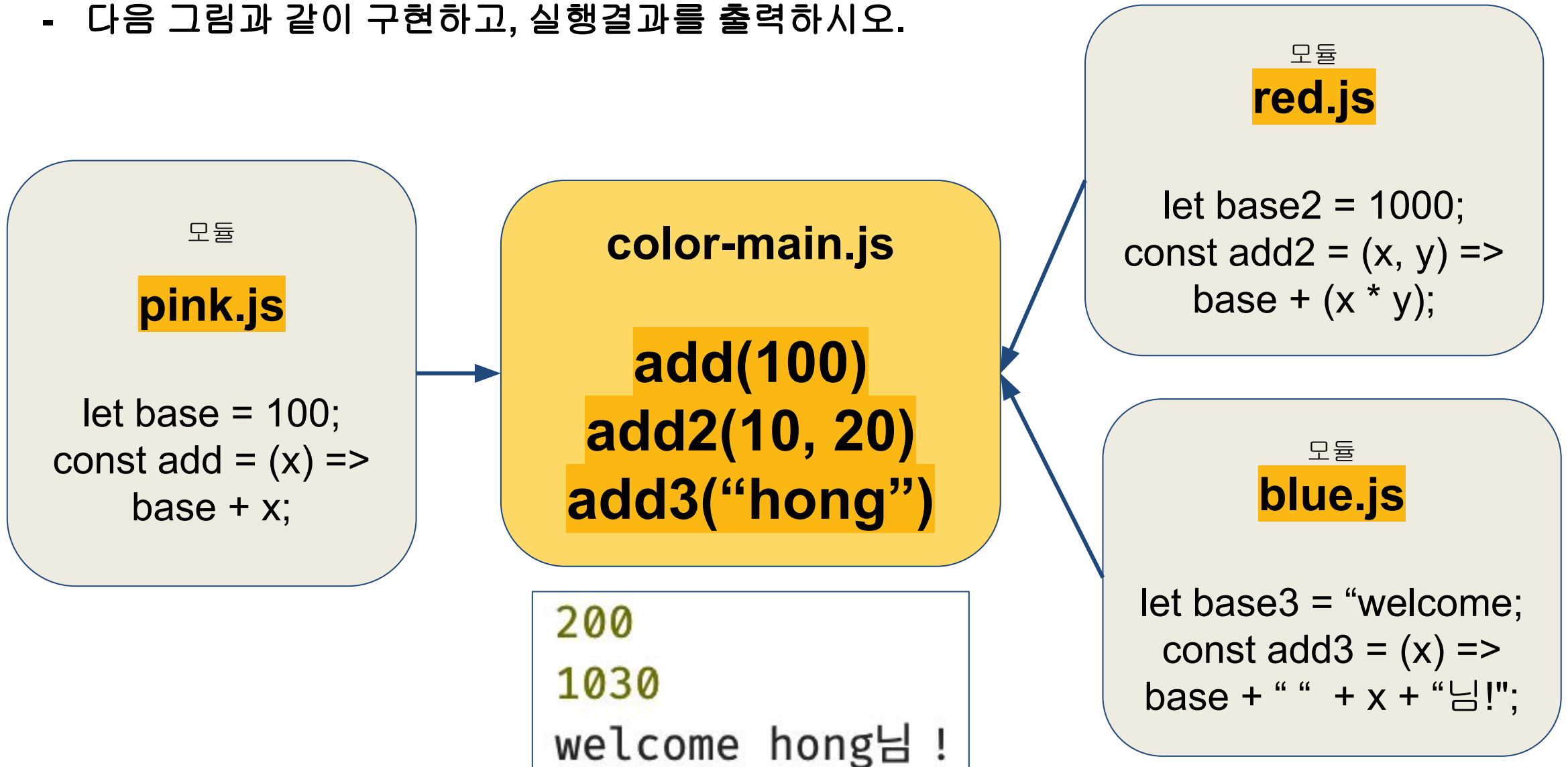
toLocaleString()함수 이용



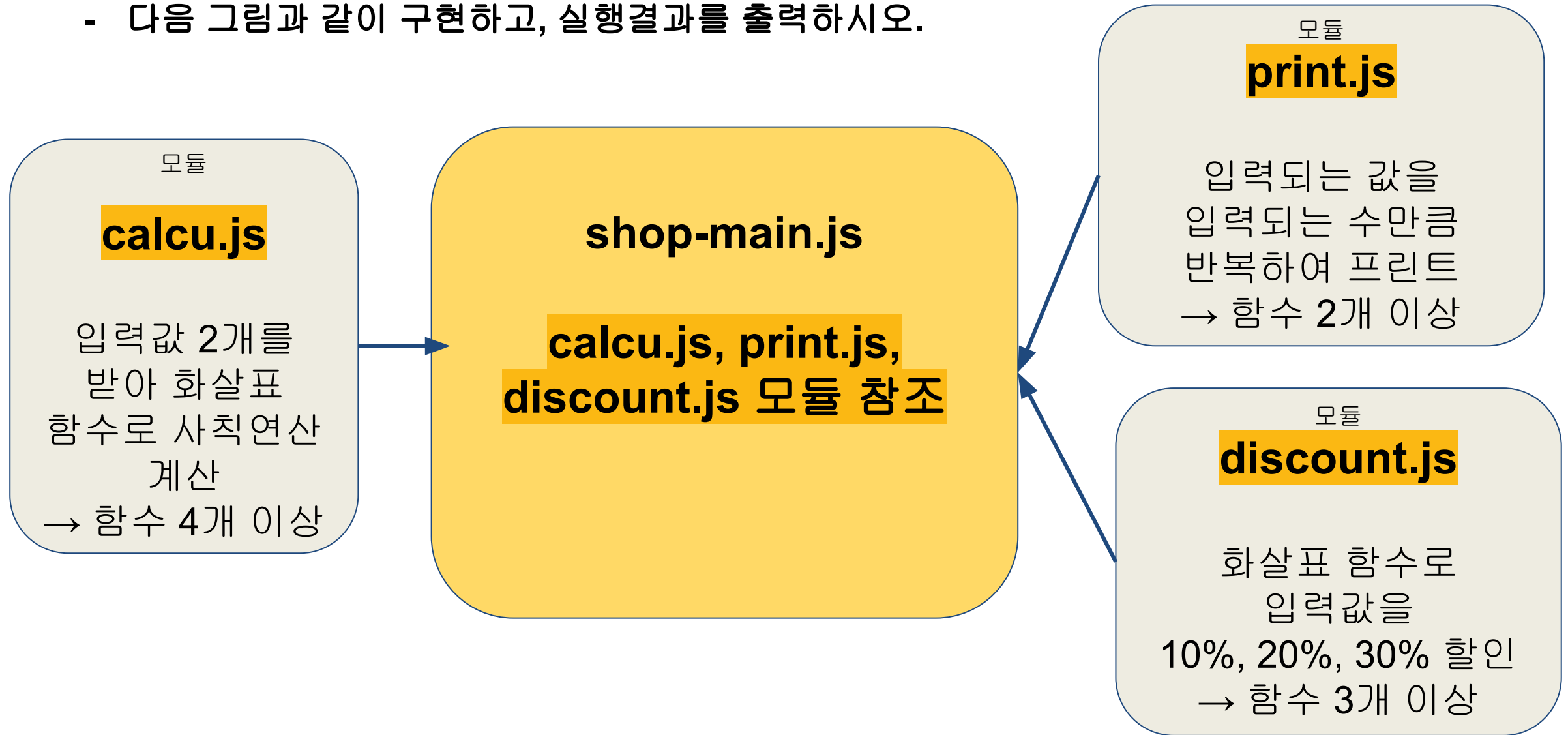
- import문으로 모듈을 사용하고자 하는 경우에는 다음과 같이 설정해준다.



- 다음 그림과 같이 구현하고, 실행결과를 출력하시오.



- 다음 그림과 같이 구현하고, 실행결과를 출력하시오.



덧셈 :  $10 + 5 = 15$

뺄셈 :  $10 - 5 = 5$

곱셈 :  $10 * 5 = 50$

나눗셈 :  $10 / 5 = 2$

Hello, world!

Hello, world!

Hello, world!

Original Price: ₩10,000

Price after 10% discount: ₩9,000

Price after 20% discount: ₩8,000

Price after 30% discount: ₩7,000

- 주문 종류: 라떼, 아이스티, 밀크티
- 주문 가격: 라떼(5000원), 아이스티(6000원), 밀크티(5500원)
- 라떼 2개, 아이스티 4개, 밀크티 3개 주문한 경우
  - 참조한 모듈을 사용하여 계산 후, 다음 페이지와 같이 출력
- 총 계산 금액
  - 15만원 이상이면 30%할인 결제
  - 10만원 이상이면 20%할인 결제
  - 5만원 이상이면 10%할인 결제

총 주문 금액 : ₩50,500

할인 후 결제 금액 : ₩45,450

## - 주문 내용:

- 주문 종류	가격	수량	주문가격
- 라떼	5000원	2	10000원
- 아이스티	6000원	4	24000원
- 밀크티	5500원	3	16500원

- -----

- 전체 주문 종류 : 라떼, 아이스티, 밀크티
- 전체 주문 수량 : 9개
- 전체 주문 금액 : 50500원

- -----

- 할인 금액 : 5050원
- 총 결제 금액 : 45450원

수고하셨습니다!

