

## 연산자 연습문제

---

### 문제1.

현재 년도에 대한 값을 `year`라는 상수로 생성하고 이 값을 활용하여 자신의 나이를 연산한 후 `age`라는 이름의 지역변수에 할당하고 "나는 OO세 입니다." 라는 형식의 문장으로 이스케이프 문자를 활용하여 출력하시오.

### 문제2.

자신의 나이를 의미하는 상수 `age`를 정의하고 이 값을 활용하여 자신이 태어난 년도 `year`를 전역 변수 형식으로 산출하여 "나는 OOOO년도에 태어났습니다."라는 형식의 문장으로 이스케이프 문자를 활용하여 출력하시오.

### 문제3.

사과를 바구니에 나누어 담으려고 한다. 하나의 바구니는 사과를 10개씩 담을 수 있으며 사과를 담다가 10개 미만으로 남는 경우 하나의 바구니를 추가로 사용해야 한다. 예를 들어 120개의 사과를 나누어 담기 위해서는 12개의 바구니가 필요하지만 121~130개까지 사과가 있다면 13개의 바구니가 필요하고 다시 131개의 사과를 나누어 담기 위해서는 14개의 바구니가 필요하게 된다.

현재 갖고 있는 사과의 수를 의미하는 `numOfApples` 변수에 123이라는 값이 할당되어 있을 경우 필요한 바구니의 수를 구하는 프로그램을 구현하시오.

### 문제4.

어떤 계산기는 입력된 값에서 백의자리 이하를 버리고 결과를 도출한다고 한다. 예를 들어 입력된 값이 457이라면 400이 결과로 출력된다. 이러한 출력결과를 만들 수 있는 코드를 작성하시오.

## 프로그램 흐름제어 연습문제

---

### 문제1

아래의 `switch`문을 `if`문으로 재작성 하시오.

```
const 수학 = "B";

switch (수학) {
  case 'A':
  case 'B':
  case 'C':
    console.log("이 과목을 Pass 했습니다.");
    break;
  default:
    console.log("이 과목을 Pass하지 못했습니다.");
    break;
}
```

## 문제2

한 개의 2진수는 0과 1 두 개의 정보를 표시할 수 있고 두 개의 이진수는 00,01,10,11과 같이 총 네 개의 정보를 표시할 수 있다. 이처럼 이진수가 하나 늘어날 때 마다 2배씩 표현할 수 있는 정보의 수가 늘어날 때 10개의 이진수는 총 몇개의 정보를 표현할 수 있는지를 while문으로 구현하시오.

### 출력예시

```
이진수 1개는 2개의 정보를 표시가능
이진수 2개는 4개의 정보를 표시가능
이진수 3개는 8개의 정보를 표시가능
이진수 4개는 16개의 정보를 표시가능
이진수 5개는 32개의 정보를 표시가능
이진수 6개는 64개의 정보를 표시가능
이진수 7개는 128개의 정보를 표시가능
이진수 8개는 256개의 정보를 표시가능
이진수 9개는 512개의 정보를 표시가능
이진수 10개는 1024개의 정보를 표시가능
```

## 문제3

위 2번 문제를 for문으로 다시 표현하시오.

## 기본문법 활용하기 연습문제

---

### 문제 1.

for문을 사용하여 0부터 10미만의 정수 중에서 홀수만을 큰수부터 출력하시오.

### 문제 2.

while문을 사용하여 0 부터 10 미만의 정수 중에서 홀수만을 큰수부터 출력하시오.

### 문제 3.

1부터 20 미만의 정수 중에서 2 또는 3의 배수인 수의 총합을 구하시오.

### 문제 4.

두 개의 주사위를 던졌을 때, 눈의 합이 6이 되는 모든 경우의 수를 출력하고 경우의 수는 총 몇가지 인지를 아래와 같이 출력하는 코드를 작성하시오.

```
[ 1, 5 ]
[ 2, 4 ]
[ 3, 3 ]
[ 4, 2 ]
[ 5, 1 ]
경우의 수는 5개 입니다.
```

## 문제 5.

for문을 중첩하여 실행하여 아래와 같은 출력 결과를 만드시오.

```
0 1 2 3
1 2 3 4
2 3 4 5
3 4 5 6
```

## 문제 6.

아래와 같은 출력 결과가 나타나도록 중첩 반복문을 for 문 형식으로 구현하시오.

```
1
12
123
1234
12345
123456
1234567
```

## 문제 7.

number라는 변수를 정의하고 1 혹은 2의 값을 임의로 할당하시오. 이 변수에는 1이나 2밖에 저장될 수 없다.

구구단 프로그램을 만들고자 한다.

전체를 출력하는 구구단이 아니라 number가 1일 때는 홀수 단(3, 5, 7, 9), number가 2일 때는 입력하면 짝수 단(2, 4, 6, 8)을 출력하는 프로그램을 완성하시오.

## 배열 연습문제 2

### 문제 12.

자신의 주민번호 한 글자씩 모든 숫자를 원소로 갖는 배열 jumin을 아래와 같이 정의하시오.

```
ssn = [0,1,1,2,1,3,1,0,0,0,1,2,3]
```

정의된 배열을 활용하여 유효한 주민등록번호인지 아닌지를 판별하는 코드를 구현해보자. 판별 방법은 아래와 같다.

- 기본 주민등록코드에는 각 숫자에 대응하는 가중치가 있다. 가중치는 주민등록번호의 순서에 따라 2 3 4 5 6 7 8 9 2 3 4 5 이다.
- 먼저 마지막 숫자는 제외하고, 기본코드의 각 12자리와 가중치를 모두 곱하여 합한다.
- 합한 값을 11로 나눈 나머지 값을 구한다.
- 11에서 그 나머지 값을 뺀 후, 이를 10을 나눈 나머지를 구한다.
- 나머지의 1의 자리 값과 주민등록번호 마지막 자리 값이 맞아야 유효한 주민등록번호이다.

## JSON 연습문제

### 문제1.

다음은 10명의 학생들에 대한 혈액형 데이터이다.

```
['A', 'A', 'A', 'O', 'B', 'B', 'O', 'AB', 'AB', 'O']
```

아래와 같은 JSON을 정의하고, 각 혈액형별 학생수를 아래의 json의 각 key에 대한 value에 저장하시오. (혈액형별 학생 수를 for문을 활용하여 산출해야 합니다.)

```
const result = {"A" : 0, "B" : 0, "AB" : 0, "O" : 0};
```

### 문제2.

다음의 JSON은 어느 학급의 중간고사 성적을 나타낸다.

```
const exam = {  
  "철수": [89, 82, 79, 91],  
  "민영": [91, 95, 94, 89],  
  "남철": [65, 57, 71, 64],  
  "혜진": [82, 76, 81, 83]  
}
```

#### 2-1.

위 데이터에서 학생별 총점과 평균을 구하시오.

#### 출력결과

철수의 총점은 341점 이고 평균은 85.25점 입니다.  
민영의 총점은 369점 이고 평균은 92.25점 입니다.  
남철의 총점은 257점 이고 평균은 64.25점 입니다.  
혜진의 총점은 322점 이고 평균은 80.5점 입니다.

2-2.

위 문제의 점수가 순서대로 국어, 영어, 수학, 과학일 경우 수학에 대한 모든 학생의 총점과 평균을 구하시오.

**출력결과**

모든 학생의 수학 총점은 325점 이고 평균은 81.25점 입니다.

**문제3.**

아래의 데이터는 2021년 01월 25일부터 02월 01일까지의 Covid19 일별 확진자 수를 표현한 자료구조다.

```
covid19 = [  
    {date: '0125', active: 426},  
    {date: '0126', active: 343},  
    {date: '0127', active: 547},  
    {date: '0128', active: 490},  
    {date: '0129', active: 460},  
    {date: '0130', active: 443},  
    {date: '0131', active: 338},  
    {date: '0201', active: 299}  
]
```

3-1.

1월 25일부터 2월 1일까지의 누적 확진자 수와 일 평균 확진자 수를 구하시오.

**출력결과**

누적 확진자 수: 3346  
평균 확진자 수: 418.25

3-2

1월 25일부터 2월 1일까지 중에서 확진자가 가장 많이 나타난 날짜는 언제인가?

**출력결과**

확진자가 가장 많이 나타난 날: 0127

**함수 연습문제**

## 문제1.

아래의 형태로 출력하는 함수를 구현하시오.

```
// max는 출력해야 할 최대 라인 수
function printStar(max) {
    ... 구현하세요 ...
}

printStar(5)
```

### 출력결과

```
*
**
***
****
*****
```

## 문제2.

1번 문제를 응용하여 같은 파라미터를 받았을 때 별을 역으로 출력하는 `printRevStar(max)` 을 구현하시오.

### 출력결과

```
*****
****
***
**
*
```

## 문제3

369게임은 숫자를 순서대로 말하면서 3,6,9가 포함된 횟수만큼 박수를 치는 게임이다.

1부터 파라미터로 전달된 숫자까지 반복하면서 박수를 칠 조건이 충족되면 3,6,9 게임 규칙에 따라 박수를 의미하는 "짝"을 출력하고 그렇지 않은 경우에는 숫자를 출력하고, 박수를 총 몇번 쳤는지를 리턴하는 함수 `myGame(n)`을 작성하시오.

힌트: 문자열은 그 자체가 배열로 인식됩니다.

ex)

```
const str = "Hello";
console.log(str[0]); // --> H
```

```
console.log(str[1]); // --> e
console.log(str[2]); // --> l
```

## 출력예시

myGame(35)를 호출한 경우

```
1
2
짝(3) --> 1번
4
5
짝(6) --> 1번
7
8
짝(9) --> 1번
10
11
12
짝(13) --> 1번
14
15
...
짝(29) --> 1번
짝(30) --> 1번
짝(31) --> 1번
짝(32) --> 1번
짝짝(33) --> 2번
짝(34) --> 1번
짝(35) --> 1번

박수를 총 00번 쳤습니다.
...
```

## 문제4

문제1번을 재귀함수 기법으로 다시 구현해 보세요.

```
// max는 최대 행 수, current는 현재 출력중인 행의 위치
function printStar(max, current=1) {
    ... 구현하세요 ...
}

printStar(5);
```

## 문제5

문제2번을 재귀함수 기법으로 다시 구현해 보세요.

```
// max는 최대 행 수, current는 현재 출력중인 행의 위치
function printRevStar(max, current=1) {
  ... 구현하세요 ...
}

printRevStar(5);
```

## 프로토타입 연습문제

### 문제1.

국어, 영어, 수학 점수를 생성자 파라미터로 입력받아서 합계와 평균을 구하는 클래스 Student를 작성하시오.

이 때 Student 클래스는 합계를 리턴하는 메서드인 `sum()`과 평균을 리턴하는 `avg()`를 제공합니다.

작성된 클래스를 활용하여 아래 표에 대한 학생별 합계 점수와 평균점수를 출력하시오.

클래스는 JSON 형식으로 작성되어야 합니다.

이름	국어	영어	수학
철수	92	81	77
영희	72	95	98
민혁	80	86	84

#### 출력결과

```
철수의 총점은 250점 이고 평균은 83.33333333333333점 입니다.
영희의 총점은 265점 이고 평균은 88.33333333333333점 입니다.
민혁의 총점은 250점 이고 평균은 83.33333333333333점 입니다.
```

### 문제2.

가로(`width`), 세로(`height`)정보를 getter, setter로 관리하는 Rectangle 클래스를 정의하시오.

이 클래스는 생성자의 파라미터가 없으며 둘레의 길이를 구해 리턴하는 `getAround()` 메서드와 넓이를 구해 리턴하는 `getArea()` 메서드를 제공합니다.

클래스는 JSON 형식으로 작성되어야 합니다.

#### 출력결과

가로가 10이고 세로가 5인 경우



둘레의 길이는 30이고 넓이는 50입니다.

## Class 기반 객체지향 연습문제

---

### 문제1.

앞 단원에서 수행한 연습문제 1,2번을 Class 기반의 객체지향으로 재구성하시오.

### 문제2.

다음을 만족하는 Student 클래스를 작성하시오.

1. String형의 학과와 정수형의 학번을 프로퍼티로 선언후 생성자를 통해 주입
2. getter, setter를 정의
3. sayHello() 메서드를 통해 "나는 OOOO학과 OO학번 입니다." 를 출력하는 기능을 구현

### 문제3.

다음을 만족하는 클래스 Account를 작성하시오.

1. 다음의 2 개의 필드를 선언
  - 문자열 owner; (이름)
  - 숫자형 balance; (금액)
2. 위 모든 필드에 대한 getter와 setter의 구현
3. 위 모든 필드를 사용하는 가능한 모든 생성자의 구현
4. 메소드 deposit()의 헤드는 다음과 같으며 인자인 금액을 저축하는 메소드
  - deposit(amount)
5. 메소드 withdraw()의 헤드는 다음과 같으며 인자인 금액을 인출(리턴)하는 메소드
  - withdraw(long amount)
  - 인출 상한 금액은 잔액까지로 하며, 이 경우 이러한 상황을 출력

## 내장객체 연습문제

---

### 문제1.

자신의 이메일 주소를 `email` 이라는 변수에 저장하고 @를 기준으로 아이디와 도메인 부분을 분리하여 출력하시오.

### 출력결과

이메일이 leekh4232@gmail.com 인 경우

```
leekh4232  
gmail.com
```

## 문제2.

임의의 주민번호를 다음과 같이 \*을 포함하여 변수에 저장하시오.

```
ssn = '020517-3*****'
```

또한 현재 년도를 now\_year라는 변수로 저장하시오.

이 값을 사용하여 생년월일, 나이, 성별을 출력하시오.

### 출력결과

```
2002년 5월 17일에 태어난 20세 남자 입니다.
```

## 문제3.

아래의 문장에서 "수업시간"이라는 단어가 총 몇 번 등장하는지 카운트 하는 프로그램을 구현하시오.

```
str = "수업시간에 배운것은 수업시간에 다 이해하고 넘어가야지 수업시간에 놓치면 따라오기  
힘들다."
```

### 출력결과

```
3
```

## 문제4. (로또번호 생성기 1)

주어진 범위 안에서 랜덤한 숫자를 추출하는 함수는 아래와 같다.

```
function random(n1, n2) {  
    return parseInt(Math.random() * (n2 - n1 + 1)) + n1;  
}
```

0개의 원소를 갖는 배열 lotto를 생성하고 6회의 반복을 수행하는 for문을 사용하여 배열의 각 원소를 1~45 사이의 범위를 갖는 임의의 숫자로 채워 넣으시오.

반복이 종료되었을 때 **lotto**의 원소는 6개의 숫자가 채워져 있어야 하고 각 숫자는 중복되지 않아야 합니다.

중복되지 않는 숫자를 생성하기 위해 for문 안에서 무한반복을 위한 while문을 수행해야 합니다.

## 문제5. (로또번호 생성기 2)

1~45사이의 범위의 1씩 증가 하는 원소가 저장되어 있는 배열 **balls**을 생성하고 6개의 빈 칸을 갖는 배열 **lotto**를 생성하시오.

**lotto** 배열을 탐색하는 반복을 수행하면서 **balls** 배열에서 임의의 원소 하나를 추출하여 **lotto** 배열에 채워 넣으시오.

추출된 숫자는 **balls** 배열에서는 삭제되어야 합니다.

## 문제6.

수많은 마라톤 선수들이 마라톤에 참여하였습니다. 단 한 명의 선수를 제외하고는 모든 선수가 마라톤을 완주하였습니다.

마라톤에 참여한 선수들의 이름이 담긴 배열 **participant**와 완주한 선수들의 이름이 담긴 배열 **completion**이 주어질 때, 완주하지 못한 선수의 이름을 **return** 하도록 **solution** 함수를 작성해주세요.

```
function solution(participant, completion) {  
    var answer = '';  
    return answer;  
}
```

### 제한사항

- 마라톤 경기에 참여한 선수의 수는 1명 이상 100,000명 이하입니다.
- completion**의 길이는 **participant**의 길이보다 1 작습니다.
- 참가자의 이름은 1개 이상 20개 이하의 알파벳 소문자로 이루어져 있습니다.
- 참가자 중에는 동명이인이 있을 수 없습니다.

### 입출력 예

participant	completion	return
["leo", "kiki", "eden"]	["eden", "kiki"]	"leo"
["marina", "josipa", "nikola", "vinko", "filipa"]	["josipa", "filipa", "marina", "nikola"]	"vinko"
["mislav", "stanko", "steave", "ana"]	["stanko", "ana", "mislav"]	"steave"

```
// "leo"는 참여자 명단에는 있지만, 완주자 명단에는 없기 때문에 완주하지 못했습니다.  
// 출력결과: "leo"가 출력  
console.log(solution(["leo", "kiki", "eden"],  
                      ["eden", "kiki"]));
```

```
// "vinko"는 참여자 명단에는 있지만, 완주자 명단에는 없기 때문에 완주하지 못했습니다.  
// 출력결과: "vinko"가 출력  
console.log(solution(["marina", "josipa", "nikola", "vinko", "filipa"],  
                      ["josipa", "filipa", "marina", "nikola"]));  
  
// "steave"는 참여자 명단에는 있지만, 완주자 명단에는 없기 때문에 완주하지 못했습니다.  
// 출력결과: "steave"가 출력  
console.log(solution(["mislav", "stanko", "steave", "ana"],  
                      ["stanko", "ana", "mislav"]));
```

### 문제1

다음의 소스코드는 boolean 데이터를 저장하고 있는 배열에 대한 어떤 처리를 보여준다.

실행 결과에서 제시하는 것과 같이 배열에 저장되어 있는 값들을 반전(true는 false로, false는 true로) 변환하는 처리를 완성하시오.

```
var check_list = [true, false, false, true, false];  
console.log("before --> " + check_list);
```

```
console.log("after --> " + check_list);
```

### 실행결과

	Elements	Console	Sources	N
ⓘ	top	▼	Filter	
		before --> true,false,false,true,false		
		after --> false,true,true,false,true		

### 문제2

다음 표는 어떤 학생의 과목별 점수이다.

	HTML	CSS	Javascript
점수	75	82	91

이 학생의 총점과 평균점수를 구하는 프로그램에 대한 아래의 소스코드를 완성하시오.

```
var grade = ; // 성적표 배열  
var sum = 0, avg = 0; // 총점과 평균점수 변수 생성
```

```
// 총점 구하기
```

```
// 평균 구하기
```

```
// arr의 값을 소수점 둘째 자리까지로 제한한다.
```

```
avg = avg.toFixed(2);  
console.log("총점: " + sum + "점, 평균점수: " + avg + "점");
```

### 실행결과

ElementsConsole

top

Filter

총점: 248점, 평균점수: 82.67점

문제3

다음 표는 어떤 학생이 일요일부터 토요일까지의 일주일간 아르바이트를 한 시간이다.

일	월	화	수	목	금	토
7	5	5	5	5	10	7

주말에는 7시간, 평일에는 5시간을 일하기로 했지만, 금요일에 다른 직원의 사정으로 대신 근무를 하게 되어 10시간을 일했다.

이 학생의 시급이 4,500이었지만 목요일부터는 5,200원으로 올랐다고 할 때 일주일간의 총 급여를 구하는 프로그램을 작성하시오. (기본 코드는 아래 제시되는 내용을 사용하세요.)

```
var time = [ 7, 5, 5, 5, 5, 10, 7 ];
var money = 0;
```

```
console.log("1주일간의 전체 급여: " + money + "원");
```


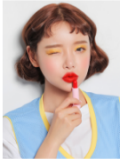


실행결과

top

1주일간의 전체 급여: 213400원

문제4

아래의 화면은 어떤 사람이 쇼핑물의 장바구니에 담은 상품에 대한 내역이다.

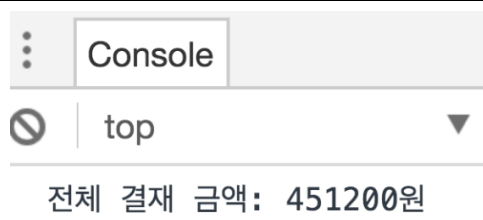
이미지	상품정보	판매가	수량	적립금	배송구분	배송비
	3CE MOOD RECIPE MULTI EYE COLOR PALETTE #PLOT TWIST [옵션: PLOT TWIST/FRE]	38,000원	6	-	기본배송 (해외배송가 능)	[무료]
	LOVE 3CE VELVET LIP STICK #GIRLS GIRLS [옵션: GIRLS GIRLS/FRE]	20,000원	4	-	기본배송 (해외배송가 능)	[무료]
	3CE MOOD RECIPE MATTE LIP COLOR #218 [옵션: 218/FRE]	17,900원	3	-	기본배송 (해외배송가 능)	[무료]
	3CE MOOD RECIPE MATTE LIP COLOR #219 [옵션: 219/FRE]	17,900원	5	-	기본배송 (해외배송가 능)	[무료]

상품의 가격을 원소로 갖는 1차 배열 price와 각 상품의 수량을 원소로 갖는 1차 배열 qty를 다음과 같이 정의하였을 때 이 사람이 총 얼마를 결제해야 하는지 총 결제금액을 구하는 프로그램을 작성하시오.

```
var price = [38000, 20000, 17900, 17900];
var qty = [6, 4, 3, 5];
var money = 0;
```

```
console.log("전체 결제 금액: " + money + "원");
```

#### 실행결과



#### 문제5

문제 4번의 장바구니 내역에서 상품금액(판매가\*수량)이 가장 비싼 항목은 얼마인지 출력하시오.

## 실행결과

⋮

Console

⊘

top

가장 높은 상품금액: 228000원

## 문제6

문제4번의 장바구니 내역에서는 모든 장바구니 상품이 개별 배송이라고 한다. 상품금액(판매가\*수량)이 8만원 이상인 경우 무료로 배송이 된다고 할 때 무료로 배송되는 항목은 모두 몇 개 인지 구하는 프로그램을 구현하시오.

## 실행결과

⋮

Console

⊘

top

무료배송 항목: 3건

## 문제7

다음은 어느 쇼핑몰의 상품 목록 화면이다.

네이버쇼핑 랭킹순

낮은 가격순

높은 가격순

등록일순

상품평 많은순

쇼핑물선택

상품타입(전체)

40개씩 보기

리스트

그리드

<p>[NIKE] 나이키 에어 베이퍼맥스 CS (AH904...)</p> <p>209,000원</p> <p>🔍광고 핫타공식물</p> <p>10% 쿠폰, 무료배송</p>	<p>나이키 에어 베이퍼 맥스 플라이니트 8...</p> <p>109,000원</p> <p>🔍광고 문트레이드</p> <p>N Pay 포인트 3,270원</p> <p>리뷰 702</p>	<p>나이키 에어맥스 제로 에센셜 / 블랙</p> <p>119,000원</p> <p>🔍광고 panypany</p> <p>N Pay 포인트 4,570원</p> <p>리뷰 45</p>	<p>나이키 에어 볼텍스 903 896-003</p> <p>109,000원</p> <p>🔍광고 나이키공식홈...</p> <p>무료배송&amp;무료반품</p>	<p>[국내출고]나이키 에어 맥스 2017 6중</p> <p>94,000원</p> <p>🔍광고 인포인스토</p> <p>N Pay 포인트 2,820원</p> <p>리뷰 230</p>
---	--	--	---	--

아래의 소스코드는 위의 상품 목록에서 상품의 가격을 원소로 하는 배열 money를 정의하였을 때 “낮은 가격순” 버튼이 눌러졌을 때 상품의 가격을 재정렬하기 위한 코드에 대한 일부이다. 빈 칸을 채워넣어 완성하시오.



```

var price = [209000,109000,119000,109000,94000];
console.log("상품가격 --> " + price);

for (var i=0; i<[ ]; i++) {
  for (var j=[ ]; j<price.length; j++) {
    if (price[i] > price[j]) {
      var tmp = price[i];
      price[i] = price[j];
      price[j] = tmp;
    }
  }
}

console.log("낮은가격순 --> " + price);

```

#### 실행결과

⋮ Console

top ▼ Filter

상품가격 --> 209000,109000,119000,109000,94000

낮은가격순 --> 94000,109000,109000,119000,209000

#### 문제8

아래의 코드는 배열의 원소를 반대로 배치하는 소스코드의 일부이다.  
빈 칸을 완성하시오. (번호가 같은 박스는 같은 코드가 들어갑니다.)

```

var arr = [5, 3, 2, 8, 9];
console.log('before --> ' + arr);

for (var i = 0; i < parseInt([ ] 1); i++) {
  var tmp = arr[i];
  arr[i] = arr([ ] 2);
  arr([ ] 2) = tmp;
}

console.log('after --> ' + arr);

```

#### 실행결과

⋮

Console

⊘

top

▼

before --> 5,3,2,8,9

after --> 9,8,2,3,5

문제9

다음 표는 어느 학급의 성적표이다.

	HTML	CSS	Javascript
둘리	78	89	96
도우너	62	77	67
또치	54	90	80
희동	100	99	98

학생별 총점과 평균점수를 구하기 위한 소스코드를 아래와 같이 작성중이다. 빈 칸을 완성하시오.  
단, 평균점수의 경우 toFixed() 함수를 사용하여 소수점 둘째 자리까지만 출력하시오.

```

/** 학생이름 배열 */
var student = ['둘리', '도우너', '또치', '희동'];

/** 성적표 배열 */
var grade = [
    [ 78, 89, 96 ],
    [ 62, 77, 67 ],
    [ 54, 90, 80 ],
    [ 100, 99, 98 ],
];

// 총점과 평균점수를 저장할 변수
var sum = 0, avg = 0;

/** 총점과 평균 구하기 */

```

실행결과

⋮

Console

⊘

top

▼

Filter


둘리 총점: 263점, 평균: 87.67점

도우너 총점: 206점, 평균: 68.67점

또치 총점: 224점, 평균: 74.67점

희동 총점: 297점, 평균: 99.00점

문제10			
<p>위의 문제에서 반 평균을 출력하시오. 반 평균은 학생의 평균점수 총 합/학생수로 구합니다.</p> <p>[주의] toFixed() 함수로 가공된 값은 문자열 형태이기 때문에 숫자 연산이 불가능합니다. 이를 감안하여 반 점수를 구하세요.</p>			
실행결과			
⊘	top	▼	Filter
둘리 총점: 263점, 평균: 87.67점			
도우너 총점: 206점, 평균: 68.67점			
또치 총점: 224점, 평균: 74.67점			
희동 총점: 297점, 평균: 99.00점			
반평균=82.5점			

문제11			
아래는 어느 게임 유저의 아이템 인벤토리이다.			
			

첫 줄에 있는 아이템을 판매한다고 할 때 이 유저가 벌어들이는 골드(G)의 총액은 얼마인가?  
(판매를 위해 선택한 아이템의 상단에는 1개당 가격이 표시되고, 아이템을 판매할 때는 원래 가격의 90%만 받을 수 있다고 한다.)

- 가격과 아이템 수량의 정보를 2차 배열로 구성하는 소스코드를 완성하세요.  
(“단가-수량”을 한 세트로 하는 정보들을 배열로 구성하면 2차 배열 형태가 됩니다.)

#### 실행결과



## 배열 연습문제 2

### 문제 12.

자신의 주민번호 한 글자씩 모든 숫자를 원소로 갖는 배열 jumin을 아래와 같이 정의하시오.

```
ssn = [0,1,1,2,1,3,1,0,0,0,1,2,3]
```

정의된 배열을 활용하여 유효한 주민등록번호인지 아닌지를 판별하는 코드를 구현해보자. 판별 방법은 아래와 같다.

- 기본 주민등록코드에는 각 숫자에 대응하는 가중치가 있다. 가중치는 주민등록번호의 순서에 따라 2 3 4 5 6 7 8 9 2 3 4 5 이다.
- 먼저 마지막 숫자는 제외하고, 기본코드의 각 12자리와 가중치를 모두 곱하여 합한다.
- 합한 값을 11로 나눈 나머지 값을 구한다.
- 11에서 그 나머지 값을 뺀 후, 이를 10을 나눈 나머지를 구한다.
- 나머지의 1의 자리 값과 주민등록번호 마지막 자리 값이 맞아야 유효한 주민등록번호이다.

# JSON 연습문제

## 문제1.

다음은 10명의 학생들에 대한 혈액형 데이터이다.

```
['A', 'A', 'A', 'O', 'B', 'B', 'O', 'AB', 'AB', 'O']
```

아래와 같은 JSON을 정의하고, 각 혈액형별 학생수를 아래의 json의 각 key에 대한 value에 저장하시오. (혈액형별 학생 수를 for문을 활용하여 산출해야 합니다.)

```
const result = {"A" : 0, "B" : 0, "AB" : 0, "O" : 0};
```

## 문제2.

다음의 JSON은 어느 학급의 중간고사 성적을 나타낸다.

```
const exam = {  
  "철수": [89, 82, 79, 91],  
  "민영": [91, 95, 94, 89],  
  "남철": [65, 57, 71, 64],  
  "혜진": [82, 76, 81, 83]  
}
```

### 2-1.

위 데이터에서 학생별 총점과 평균을 구하시오.

#### 출력결과

```
철수의 총점은 341점 이고 평균은 85.25점 입니다.  
민영의 총점은 369점 이고 평균은 92.25점 입니다.  
남철의 총점은 257점 이고 평균은 64.25점 입니다.  
혜진의 총점은 322점 이고 평균은 80.5점 입니다.
```

### 2-2.

위 문제의 점수가 순서대로 국어, 영어, 수학, 과학일 경우 수학에 대한 모든 학생의 총점과 평균을 구하시오.

#### 출력결과

모든 학생의 수학 총점은 325점 이고 평균은 81.25점 입니다.

### 문제3.

아래의 데이터는 2021년 01월 25일부터 02월 01일까지의 Covid19 일별 확진자 수를 표현한 자료구조다.

```
covid19 = [  
    {date: '0125', active: 426},  
    {date: '0126', active: 343},  
    {date: '0127', active: 547},  
    {date: '0128', active: 490},  
    {date: '0129', active: 460},  
    {date: '0130', active: 443},  
    {date: '0131', active: 338},  
    {date: '0201', active: 299}  
]
```

#### 3-1.

1월 25일부터 2월 1일까지의 누적 확진자 수와 일 평균 확진자 수를 구하시오.

#### 출력결과

누적 확진자 수: 3346  
평균 확진자 수: 418.25

#### 3-2

1월 25일부터 2월 1일까지 중에서 확진자가 가장 많이 나타난 날짜는 언제인가?

#### 출력결과

확진자가 가장 많이 나타난 날: 0127

## 함수 연습문제

### 문제1.

아래의 형태로 출력하는 함수를 구현하시오.

```
// max는 출력해야 할 최대 라인 수
function printStar(max) {
  ... 구현하세요 ...
}

printStar(5)
```

### 출력결과

```
*
**
***
****
*****
```

## 문제2.

1번 문제를 응용하여 같은 파라미터를 받았을 때 별을 역으로 출력하는 `printRevStar(max)` 을 구현하시오.

### 출력결과

```
*****
****
***
**
*
```

## 문제3

369게임은 숫자를 순서대로 말하면서 3,6,9가 포함된 횟수만큼 박수를 치는 게임이다.

1부터 파라미터로 전달된 숫자까지 반복하면서 박수를 칠 조건이 충족되면 3,6,9 게임 규칙에 따라 박수를 의미하는 "짝"을 출력하고 그렇지 않은 경우에는 숫자를 출력하고, 박수를 총 몇번 쳤는지를 리턴하는 함수 `myGame(n)`을 작성하시오.

힌트: 문자열은 그 자체가 배열로 인식됩니다.

ex)

```
const str = "Hello";
console.log(str[0]); // --> H
console.log(str[1]); // --> e
console.log(str[2]); // --> l
```



## 출력예시

myGame(35)를 호출한 경우

```

1
2
짝(3) --> 1번
4
5
짝(6) --> 1번
7
8
짝(9) --> 1번
10
11
12
짝(13) --> 1번
14
15
...
짝(29) --> 1번
짝(30) --> 1번
짝(31) --> 1번
짝(32) --> 1번
짝짝(33) --> 2번
짝(34) --> 1번
짝(35) --> 1번

박수를 총 00번 쳤습니다.
...
```

## 문제4

문제1번을 재귀함수 기법으로 다시 구현해 보세요.

```

// max는 최대 행 수, current는 현재 출력중인 행의 위치
function printStar(max, current=1) {
    ... 구현하세요 ...
}

printStar(5);
```

## 문제5

문제2번을 재귀함수 기법으로 다시 구현해 보세요.

```
// max는 최대 행 수, current는 현재 출력중인 행의 위치
function printRevStar(max, current=1) {
    ... 구현하세요 ...
}

printRevStar(5);
```

## 프로토타입 연습문제

### 문제1.

국어, 영어, 수학 점수를 생성자 파라미터로 입력받아서 합계와 평균을 구하는 클래스 Student를 작성하시오.

이 때 Student 클래스는 합계를 리턴하는 메서드인 `sum()`과 평균을 리턴하는 `avg()`를 제공합니다.

작성된 클래스를 활용하여 아래 표에 대한 학생별 합계 점수와 평균점수를 출력하시오.

클래스는 JSON 형식으로 작성되어야 합니다.

이름	국어	영어	수학
철수	92	81	77
영희	72	95	98
민혁	80	86	84

### 출력결과

```
철수의 총점은 250점 이고 평균은 83.33333333333333점 입니다.
영희의 총점은 265점 이고 평균은 88.33333333333333점 입니다.
민혁의 총점은 250점 이고 평균은 83.33333333333333점 입니다.
```

### 문제2.

가로(`width`), 세로(`height`)정보를 getter, setter로 관리하는 Rectangle 클래스를 정의하시오.

이 클래스는 생성자의 파라미터가 없으며 둘레의 길이를 구해 리턴하는 `getAround()` 메서드와 넓이를 구해 리턴하는 `gerArea()` 메서드를 제공합니다.

클래스는 JSON 형식으로 작성되어야 합니다.

### 출력결과

가로가 10이고 세로가 5인 경우

```
둘레의 길이는 30이고 넓이는 50입니다.
```

## Class 기반 객체지향 연습문제

---

### 문제1.

앞 단원에서 수행한 연습문제 1,2번을 Class 기반의 객체지향으로 재구성하시오.

### 문제2.

다음을 만족하는 Student 클래스를 작성하시오.

1. String형의 학과와 정수형의 학번을 프로퍼티로 선언후 생성자를 통해 주입
2. getter, setter를 정의
3. sayHello() 메서드를 통해 "나는 OOOO학과 OO학번 입니다." 를 출력하는 기능을 구현

### 문제3.

다음을 만족하는 클래스 Account를 작성하시오.

1. 다음의 2 개의 필드를 선언
  - 문자열 owner; (이름)
  - 숫자형 balance; (금액)
2. 위 모든 필드에 대한 getter와 setter의 구현
3. 위 모든 필드를 사용하는 가능한 모든 생성자의 구현
4. 메소드 deposit()의 헤드는 다음과 같으며 인자인 금액을 저축하는 메소드
  - deposit(amount)
5. 메소드 withdraw()의 헤드는 다음과 같으며 인자인 금액을 인출(리턴)하는 메소드
  - withdraw(long amount)
  - 인출 상한 금액은 잔액까지로 하며, 이 경우 이러한 상황을 출력

## 내장객체 연습문제

---

### 문제1.

자신의 이메일 주소를 `email` 이라는 변수에 저장하고 @를 기준으로 아이디와 도메인 부분을 분리하여 출력하시오.

#### 출력결과

이메일이 leekh4232@gmail.com 인 경우

```
leekh4232  
gmail.com
```

## 문제2.

임의의 주민번호를 다음과 같이 \*을 포함하여 변수에 저장하시오.

```
ssn = '020517-3*****'
```

또한 현재 년도를 now\_year라는 변수로 저장하시오.

이 값을 사용하여 생년월일, 나이, 성별을 출력하시오.

### 출력결과

```
2002년 5월 17일에 태어난 20세 남자 입니다.
```

## 문제3.

아래의 문장에서 "수업시간"이라는 단어가 총 몇 번 등장하는지 카운트 하는 프로그램을 구현하시오.

```
str = "수업시간에 배운것은 수업시간에 다 이해하고 넘어가야지 수업시간에 놓치면 따라오기  
힘들다."
```

### 출력결과

```
3
```

## 문제4. (로또번호 생성기 1)

주어진 범위 안에서 랜덤한 숫자를 추출하는 함수는 아래와 같다.

```
function random(n1, n2) {  
    return parseInt(Math.random() * (n2 - n1 + 1)) + n1;  
}
```

0개의 원소를 갖는 배열 lotto를 생성하고 6회의 반복을 수행하는 for문을 사용하여 배열의 각 원소를 1~45 사이의 범위를 갖는 임의의 숫자로 채워 넣으시오.

반복이 종료되었을 때 **lotto**의 원소는 6개의 숫자가 채워져 있어야 하고 각 숫자는 중복되지 않아야 합니다.

중복되지 않는 숫자를 생성하기 위해 for문 안에서 무한반복을 위한 while문을 수행해야 합니다.

## 문제5. (로또번호 생성기 2)

1~45사이의 범위의 1씩 증가 하는 원소가 저장되어 있는 배열 **balls**을 생성하고 6개의 빈 칸을 갖는 배열 **lotto**를 생성하시오.

**lotto** 배열을 탐색하는 반복을 수행하면서 **balls** 배열에서 임의의 원소 하나를 추출하여 **lotto** 배열에 채워 넣으시오.

추출된 숫자는 **balls** 배열에서는 삭제되어야 합니다.

## 문제6.

수많은 마라톤 선수들이 마라톤에 참여하였습니다. 단 한 명의 선수를 제외하고는 모든 선수가 마라톤을 완주하였습니다.

마라톤에 참여한 선수들의 이름이 담긴 배열 **participant**와 완주한 선수들의 이름이 담긴 배열 **completion**이 주어질 때, 완주하지 못한 선수의 이름을 **return** 하도록 **solution** 함수를 작성해주세요.

```
function solution(participant, completion) {
    var answer = '';
    return answer;
}
```

### 제한사항

- 마라톤 경기에 참여한 선수의 수는 1명 이상 100,000명 이하입니다.
- **completion**의 길이는 **participant**의 길이보다 1 작습니다.
- 참가자의 이름은 1개 이상 20개 이하의 알파벳 소문자로 이루어져 있습니다.
- 참가자 중에는 동명이인이 있을 수 없습니다.

### 입출력 예

participant	completion	return
["leo", "kiki", "eden"]	["eden", "kiki"]	"leo"
["marina", "josipa", "nikola", "vinko", "filipa"]	["josipa", "filipa", "marina", "nikola"]	"vinko"
["mislav", "stanko", "steave", "ana"]	["stanko", "ana", "mislav"]	"steave"

```
// "leo"는 참여자 명단에는 있지만, 완주자 명단에는 없기 때문에 완주하지 못했습니다.
// 출력결과: "leo"가 출력
console.log(solution(["leo", "kiki", "eden"],
                      ["eden", "kiki"]));
```

```
// "vinko"는 참여자 명단에는 있지만, 완주자 명단에는 없기 때문에 완주하지 못했습니다.  
// 출력결과: "vinko"가 출력  
console.log(solution(["marina", "josipa", "nikola", "vinko", "filipa"],  
                      ["josipa", "filipa", "marina", "nikola"]));  
  
// "steave"는 참여자 명단에는 있지만, 완주자 명단에는 없기 때문에 완주하지 못했습니다.  
// 출력결과: "steave"가 출력  
console.log(solution(["mislav", "stanko", "steave", "ana"],  
                      ["stanko", "ana", "mislav"]));
```