

정선미 내장객체 연습문제

2022-02-15

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
```

```
const emailArr = email.split('@')  
for(let i of emailArr){  
  console.log(i)  
}  
  
// 1번 다르게 풀기  
const email = 'jsm760@naver.com';  
const atIndex = email.indexOf('@');  
const emailId = email.substring(0, atIndex);  
const emailAddress = email.substring(atIndex+1);  
console.log(emailId)  
console.log(emailAddress)
```

```
[Running] node "/Users/developer/workspace/studynot  
12-Javascript-내장기능/연습문제/연습문제1.js"  
jsm760  
naver.com  
  
[Done] exited with code=0 in 0.05 seconds
```

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.

```
const ssn = '020517-3*****'
const now_year = new Date();
const year = now_year.getFullYear();

const ssnYear = function(){
  if(gen == 3 || gen == 4){
    return Number(ssn.substring(0,2)) + 2000;
  }else{
    return Number(ssn.substring(0,2)) + 1900;
  }
}

const ssnMon = Number(ssn.substring(2,4));
const ssnDay = Number(ssn.substring(4,6));
const gen = Number(ssn.substring(7,8));
const ssnGen = () => gen == 1 || gen == 3 ? '남자' : '여자';
const ssnAge = year - ssnYear() + 1;

console.log(`${ssnYear()}년 ${ssnMon}월 ${ssnDay}일에 태어난 ${ssnAge}세
${ssnGen()}입니다.`)
```

```
[Running] node "/Users/developer/workspace/studynode.js"
2002년 5월 17일에 태어난 21세 남자입니다.
[Done] exited with code=0 in 0.056 seconds
```

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

```
const str = "수업시간에 배운것은 수업시간에 다 이해하고 넘어가야지 수업시간에 놓치면 따라오기  
힘들다.";  
const word = '수업시간';  
let cut = str.split(word);  
console.log(cut.length - 1);
```

```
[Running] node "/Users/developer/workspace/s  
js"  
3  
[Done] exited with code=0 in 0.056 seconds
```

문제4. (로또번호 생성기 1)

주어진 범위 안에서 랜덤한 숫자를 추출하는 함수는 아래와 같다.

0개의 원소를 갖는 배열 `lotto`를 생성하고 6회의 반복을 수행하는 `for`문을 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 `for`문 안에서 무한반복을 위한 `while`문을 수행해야 합니다.

```
function Random(n1,n2){  
  return parseInt(Math.random() * (n2 - n1 + 1)) + n1  
}  
let lotto = [];  
for(let i = 0; i < 6; i++){  
  let x = Random(1,45);  
  let j = 0;  
  while(j <= i){  
    if(!lotto.includes(x)){  
      lotto.push(x);  
    }  
    j++  
  }  
}
```

```

    }
  }
  console.log(lotto)

```

```

[Running] node "/Users/developer/workspace/stu
js"
[ 23, 20, 1, 22, 41, 21 ]

[Done] exited with code=0 in 0.061 seconds

```

문제5. (로또번호 생성기 2)

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

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

추출된 숫자는 `balls` 배열에서는 삭제되어야 합니다.

```

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

let balls = new Array(45);
for(let i=0;i<balls.length;i++){
  balls[i] = i + 1
};
//console.log(balls)
let lotto = new Array(6);
for(let i=0;i<lotto.length;i++){
  let x = Random(0,balls.length-1);
  if(!lotto.includes(x)){
    lotto[i] = balls[x];
  }else{
    i -= 1
  }
  balls.splice(x,1)
}
console.log(lotto)
console.log(balls)

```

```
[ 1, 22, 33, 34, 31, 16 ]
[
  2, 3, 4, 5, 6, 7, 8, 9, 10, 11,
  12, 13, 14, 15, 17, 18, 19, 20, 21, 23,
  24, 25, 26, 27, 28, 29, 30, 32, 35, 36,
  37, 38, 39, 40, 41, 42, 43, 44, 45
]
```

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 = '';

  for(let i of participant){
    if(!completion.includes(i)){
      answer += i
    }
  }
  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"]));
```

```
[Running] node "/Users/developer/  
js"  
leo  
vinko  
steave  
  
[Done] exited with code=0 in 0.05
```