3. 함수형 프로그래밍 (실습1)

Prof. Seunghyun Park (sp@hansung.ac.kr)

Mobility and Cybersecurity Lab. https://sp-mcslab.github.io

Division of Computer Engineering

한성대학교 HANSUNG UNIVERSITY

학습 목표: 3. 함수형 프로그래밍

- 코드 분석 준비
- 함수를 매개변수로 활용
- 함수의 결과로 함수를 반환
- 불변성: 원본 객체의 값 수정여부 확인
- 순수함수

- 함수형 프로그래밍
  - 데이터 변환
    - Array.join()
    - Array.filter(callback)
    - Array.map(callback)
    - Array.reduce(callback)
    - Array.indexOf()
    - Object.keys(object)
  - 고차 함수

실습8: 함수를 매개변수로 활용 분석

```
/* ch03/ex/ex08-1.html from ch03-01-05-functional.html */
```

- `const insideFn = logger => logger("함수를 다른 함수에 매개변수로 전달")`
- `insideFn(message => console.log(message))`

함수를 다른 함수에 매개변수로 전달

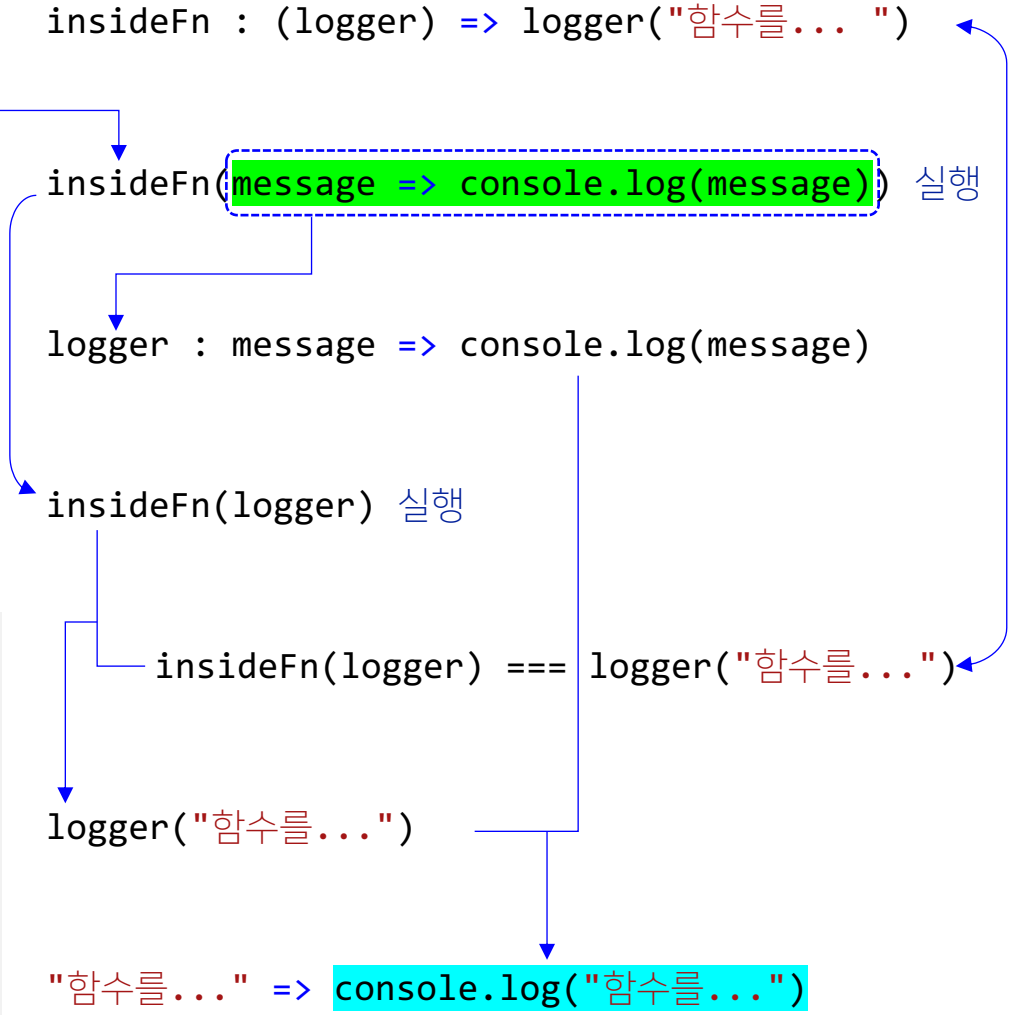
1. 코드의 주요 지점에 break point를 찍고,

2. 실행 중 객체의 값 확인

```
Watch
> insideFn
> logger
> message
```

비교:
ch03/01/05-2.html

```
/* ch03/ex/ex08-2.html */
const insideFn = (logger) => {
  logger("함수를 다른 함수에 매개변수로 전달");
};
const temp = (msg) => {
  console.log(msg);
};

insideFn(temp);
```



실습9: 함수의 결과로 함수를 반환

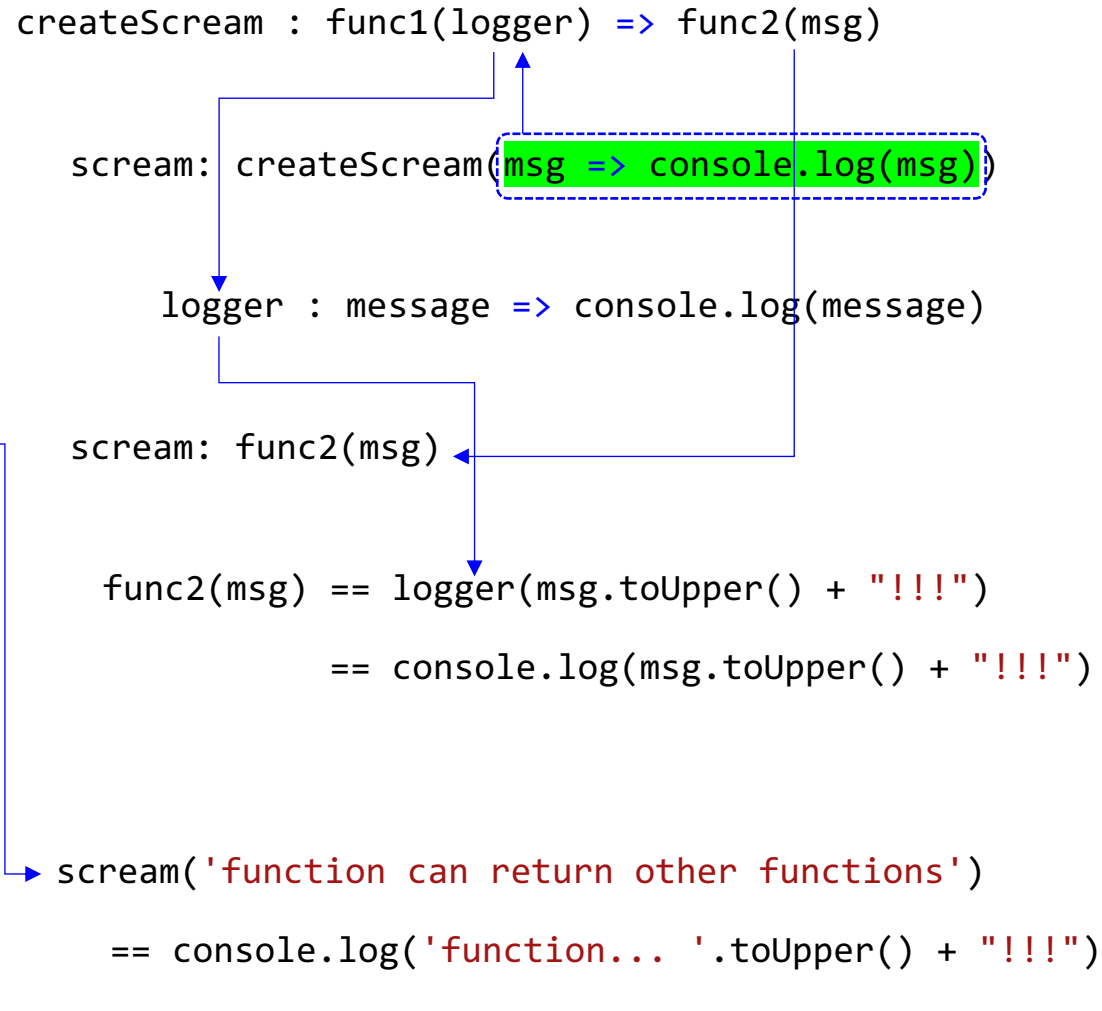
```
/* ch03/ex/ex09.html from ch03-01-06-functional.html */

const createScream = function(logger) {
    return function(message) {
        logger(message.toUpperCase() + "!!!")
    }
}

const scream = createScream(message => console.log(message))
scream('function can return other functions')
```

```
FUNCTION CAN RETURN OTHER FUNCTIONS!!!
```

1. 코드의 주요 지점에 break point를 찍고,

2. 실행 중 객체의 값 확인

```
Watch
> createScream
> logger
> message
> scream
```



실습10: 원본 객체의 값 수정여부 확인

```
/* ch03/ex/ex10-1.html from ch03/04/01.html */

let color_lawn = {
  title: "잔디",
  color: "#00FF00",
  rating: 0
}

// 매개변수로 전달받은 객체의 속성을 변경하여 반환
function rateColor(obj, rating) {
  obj.rating = rating
  return obj
}

console.log(color_lawn.rating)

console.log(rateColor(color_lawn, 5).rating)
console.log(color_lawn.rating)
```

실습11: 순수함수

```
/* ch03/ex/ex11-1.html from ch03/05/01.html */
let frederick = {
  name: "Frederick",
  canRead: false,
  canWrite: false
}

// 순수하지 않은 함수
// 인자 없음, return 문 없음, 원본 객체를 변화시킴
function selfEducate() {
  frederick.canRead = true;
  frederick.canWrite = true;
}

console.log(frederick);

selfEducate();
console.log(frederick);
```

실습 12~15. 데이터 변환

- 순수 함수를 사용한 데이터 처리 → 원본의 복제본을 생성하여 처리하고, 결과를 반환

- `Array.prototype.join()`
  - 배열의 모든 요소를 연결해 하나의 문자열로 만들어 반환

- `Array.prototype.filter(callback)`
  - 주어진 함수의 조건을 통과하는 모든 요소를 모아 새로운 배열로 반환

- `Array.prototype.map(callback)`
  - 배열 내 모든 요소에 대해 주어진 함수를 호출한 결과를 새로운 배열로 반환

- `Array.prototype.reduce(callback, initValue)`
  - 배열 내 모든 요소에 대해 callback을 실행하고 하나의 결과 값을 반환

- `Array.prototype.indexOf()`
  - 배열 내 지정된 요소를 찾을 수 있는 첫 번째 인덱스 반환 (단, 존재하지 않으면 -1 반환)

- `Object.keys(object)`
  - 주어진 객체의 속성 이름을 열거하는 배열을 반환

실습12: 데이터 변환 – Array.join()

- **문제: data의 요소를 (,)으로 연결하여 문자열로 출력**

  (단, 원래의 문자열을 변경하지 않음)

```
/* ch03/ex/ex12.html */
const address = [
    "Division of Computer Engineering",
    "Hansung University",
    "116",
    "Samseongyo-ro",
    "16-gil",
    "Seongbuk-gu",
    "Seoul",
    "02876",
    "South Korea"
];

console.log(address);
console.log(                    );
console.log(address);
```

- `Array.prototype.join()`
  - 배열의 모든 요소를 연결해 하나의 문자열로 만들어 반환

```
(9) ['Division of Computer Engineering', 'Hansung
University', '116', 'Samseongyo-ro', '16-gil',
'Seongbuk-gu', 'Seoul', '02876', 'South Korea']
Division of Computer Engineering, Hansung University,
116, Samseongyo-ro, 16-gil, Seongbuk-gu, Seoul, 02876,
South Korea
(9) ['Division of Computer Engineering', 'Hansung
University', '116', 'Samseongyo-ro', '16-gil',
'Seongbuk-gu', 'Seoul', '02876', 'South Korea']
```

```
 0: "Division of Computer Engineering"
 1: "Hansung University"
 2: "116"
 ...
 8: "South Korea"
 length: 9
[[Prototype]]: Array(0)
```

실습13-1: 데이터 변환 – Array.filter()

- 문제: data의 요소 중 첫 글자가 'S'인 요소만 출력

```
 0: "Division of Computer Engineering"
 1: "Hansung University"
 2: "116"
 3: "Samseongyo-ro"
 4: "16-gil"
 5: "Seongbuk-gu"
 6: "Seoul"
 7: "02876"
 8: "South Korea"
 length: 9
[[Prototype]]: Array(0)
```

- Array.prototype.filter(*callback*)
  - 주어진 함수의 조건을 통과하는 모든 요소를 모아 새로운 배열로 반환

```
/* ch03/ex/ex13-1.html */

const address = [ ... ];

console.log(address);
console.log(                                    );
console.log(address);
```

```
(9) ['Division of Computer Engineering', 'Hansung University', '116', 'Samseongyo-ro', '16-gil', 'Seongbuk-gu', 'Seoul', '02876', 'South Korea']
(4) ['Samseongyo-ro', 'Seongbuk-gu', 'Seoul', 'South Korea']
(9) ['Division of Computer Engineering', 'Hansung University', '116', 'Samseongyo-ro', '16-gil', 'Seongbuk-gu', 'Seoul', '02876', 'South Korea']
```

실습13-2: 데이터 변환 – Array.filter()

- 문제: data의 요소 중
  첫 글자가 'S'가 아닌 요소만 출력하되,
  주어진 기능을 수행하는 함수로 구현

- `Array.prototype.filter(callback)`
  - 주어진 함수의 조건을 통과하는 모든 요소를 모아 새로운 배열로 반환

```
/* ch03/ex/ex13-2.html */

const address = [ ... ];

console.log(address);

const newFilter =
                                                    ;
console.log(newFilter(address, 'S'));

console.log(address)
```

```
(9) ['Division of Computer Engineering', 'Hansung University', '116', 'Samseongyo-ro', '16-gil', 'Seongbuk-gu', 'Seoul', '02876', 'South Korea']
(5) ['Division of Computer Engineering', 'Hansung University', '116', '16-gil', '02876']
(9) ['Division of Computer Engineering', 'Hansung University', '116', 'Samseongyo-ro', '16-gil', 'Seongbuk-gu', 'Seoul', '02876', 'South Korea']
```