

6_JS(ES6)_PART_12

1. 일급 함수 (first class function)

자바스크립트 함수의 특징

- (1) 자바스크립트는 함수를 일급 시민(first class citizen) 으로 다룸
- (2) 이 말은 곧, 함수를 단순한 값으로 본다는 뜻
- (3) 함수는 객체 (object) 타입 중 하나

<https://www.geeksforgeeks.org/what-is-first-class-citizen-in-javascript/>

: 프로그래밍 언어가 함수를 값으로 취급하고, 인수로 전달하고, 다른 함수에서 함수를 반환하는 기능이 있는 경우 프로그래밍 언어에는 일급 함수가 있으며 함수는 반환하는 기능이 있는 경우 프로그래밍 언어에는 일급 함수가 있으며 함수는 해당 프로그래밍 언어에서 일급 시민이라고 한다.

■ 다음과 같은 문법이 가능하다

```
//함수 표현식, 화살표 함수
const addFunction = (a, b) => a + b;

const counter = {
  value: 10,
  //오브젝트 메서드에 함수 표현식
  increment: function() {
    this.value++
  }
}

//함수를 값처럼 전달
button.addEventListener('click', () => console.log('click'));
```

2. 고차함수(higher order function, HOC)

함수를 받거나, 함수를 리턴하는 함수

자바스크립트가 일급 함수이기 때문에 가능하다.

```
//함수를 받는 addEventListener 는 HOC, () => console.log('click') 이 함수는 콜백함수
button.addEventListener('click', () => console.log('click'));

//함수를 리턴하는 함수 counter는 HOC
function count() {
  let cnt = 0;
  return function() {
    cnt++;
  }
}
```

3. 클로저

모든 함수는 그 함수가 만들어지는 시점의 실행 컨텍스트를 기억하며, (해당 실행 컨텍스트가 없더라도) 그 변수 환경에 접근할 수 있다.

```
const passengerBooking = function() {  
  let passengerCount = 0;  
  
  return function() {  
    passengerCount++;  
    console.log('passengerCount', passengerCount);  
  }  
}  
  
const booker = passengerBooking();
```

```
booker();  
booker();  
booker();  
//passengerCount1  
//passengerCount2  
//passengerCount3
```

: booker 가 만들어지는 시점을 기억한다.

클로저는 함수가 만들어지는 시점의 변수들 간 연결을 끊어지지 않게 해주는 장치이다.

ex) 함수 - 사람 | 클로저 - 가방 | 변수 - 공책, 연필

-
- 자바스크립트 함수는 일급 함수 특성을 가진다 따라서 고차 함수로 사용하는 것도 가능하다
 - 클로저는 함수가 생성되는 시점의 EC 를 기억하여 변수를 연결하는 장치