

첫 복습 정리

- 2주차에 배운 내용: 변수, 문, 연산자, 제어문, 데이터 타입, 객체, 클래스, 문자열, 배열, 타입변환, 함수
- 3주차에 배운 내용: 구조분해할당, 돔, 스타일 조작, 이벤트, this, 실행 컨텍스트, 스코프, 클로저

1. 데이터 타입: String, Number, Function, Array, Object, Boolean, undefined(데이터 없는 것), null(빈 데이터를 넣은 것)

- 데이터 타입 변환하기 문자를 숫자로: `parseInt('문자열인 숫자')` 숫자를 문자열로: `toString(몇진수?)` 기본은 10진수.

2. 배열 메서드

1. 삭제와 삽입

- `.push(~)`: 뒤에 데이터 삽입
- `.unshift(~)`: 앞에 데이터 삽입
- `.pop()`: 뒤의 데이터 제거
- `.shift()`: 앞의 데이터 제거
- 지정해서 제거하려면?
- `splice()`: 원본 배열을 변경하고 **바꾼 것을 리턴**
- `slice()`: 원본 변화 X 지정한 범위의 **배열 얇은 복사 리턴**

2. 정렬

- `.sort()`: 원본 배열을 변경. **정렬한 배열을 리턴**
- `.reverse()`: 원본 배열의 순서를 반전.

3. 검사

- `.filter()`: 조건이 참인 값만 모아서 **배열로 리턴**
- `.every()`: 모든 값이 맞으면 **true**, 아니면 **false** 리턴

4. 반복

- `.forEach(인자, 함수)`: 연산을 통해 배열을 변경
- `.map(인자, 함수)`: **새로운 배열**을 리턴
- `.reduce()`: 누적 **연산 결과 하나**를 리턴

3. 함수

- 화살표 함수 표현: `const add = (x) => {return a * 2}` `const add = x => a * 2 {}` 생략 `=>` 리턴을 생략해도 된다.

4. 객체

- 키와 밸류로 구성된 프로퍼티를 가지고 있음
- `.키값` or `.[키값]`을 통해 밸류에 접근 가능
- 프로퍼티 생성: `person.age = 20` 객체가 소유하지 않은 프로퍼티 키에 값을 할당
- `for-in`과 `for-of`의 차이: 객체는 `for-in` 반복문을 사용한다. 키를 순회하기 위한 문법으로 프로퍼티의 순서(원래 없었음)가 보장되지 않는다. 이를 극복하기 위해 배열을 위한 `for-of`가 추가되었다.

5. DOM(Document Object Model: 문서 객체 모델)

- HTML 요소는 HTML DOM을 통해 접근 가능

- Document 메소드: HTML 요소 선택, HTML 요소 생성, HTML 이벤트 핸들러 추가, HTML 객체의 선택 등
- HTML 요소의 선택: `.getElementBy~` 시리즈(5에 유의할 것) `.querySelector()`
- HTML 요소 생성: `document.createElement()` `document.write()`
- HTML 요소 내용 변경: `.innerHTML` or `.innerText` : 태그 안의 값 or 텍스트 변경
- css 스타일 접근 (요소가 선택된 경우): `.style.바꿀스타일 = '값'` `.getAttribute('href')` 특정 태그의 속성값 가져옴 `.className` 클래스 이름을 확인하거나 `.className = '바꿀값'`으로 덮어씌움 `.classList` 클래스 값을 배열의 형태로 가져옴. `.classList.add()`, `.classList.remove()`
- 부모 혹은 형제에 접근 `.parentElement` `.firstElementChild`, `.lastElementChild` `.previousElementSibling`, `.nextElementSibling`
- `e.preventDefault()`