**JavaScript 클래스**

**<<01. 생성자 함수(prototype)>>**

**\* 객체 변수가 언제나 바뀔 수 있다는 걸 인지하고 직접 객체명.변수명이 아닌 this.변수명 O 으로 하는 게 좋음**

**\* 생성자 함수**

**\* 리터럴 방식:** const heropy= {} 방법으로 객체 선언하는 것

**\* heropy === 인스턴스**

**\* prototype:**

- 메모리에 한 번만 만들어지는 함수를 참고하는 것

**\* 클래스:** 프로토타입을 사용해서 new라는 키워드와 함께 생성자 함수로 인스턴스를 만들어내는 개념

\* prototype 으로 만들어진 메소드가 있음

**\* 생성자 함수가 일반적으로 일반 함수와 구분이 안 됨**

- (암묵적) new와 함께 사용하는 생성자 함수를 camelcase가 아닌 파스칼케이스로 첫번째 단어를 대문자로 사용하도록 함

**<<02. this>>**

**\* this:**

-일반(Normal) 함수는 호출 위치에 따라 this 정의

- 화살표(Arrow) 함수는 자신이 선언된 함수 범위에서 this 정의

**\* prototype에서의 this**

**\* 새로운 예제**

**<<03. ES6 Classes>>**

**\* Class를 통해 이전 생성자 함수 예제 바꾸기**

**<<04. 상속(확장)>>**

**\* super:** 해당 함수가 곧 상속 받은 객체가 됨. 그래서 해당 함수에 해당되는 로직을 실행할 수 있게 됨. 상속할클래스를 의미함.

- 부모 클래스의 내용이 자식 클래스에도 비효율적으로 중복 존재하고 있을 때 그 내용을 치우고 대신 들어가는 키워드

**\* class Bicycle extends Vehicle:** //클래스 상속받을클래스명 확장(상속) 상속할클래스명