

Object 오브젝트(ES5)

ES5 Object 특징

- ES5 Object에 함수가 추가됨
 - 메소드는 하나도 없음
- 빌트인 Object의 모든 메소드는
 - 대부분의 빌트인 오브젝트에 첨부됨
 - 빌트인으로 오브젝트를 생성하므로 연결이 많이 발생
- 함수는 첨부되지 않으므로
 - 연결 부하를 줄일 수 있음

ES5 Object 함수

이름	개요
defineProperty()	프로퍼티 추가, 프로퍼티 속성 변경
defineProperties()	다수의 프로퍼티 추가, 속성 변경
getPrototypeOf()	prototype에 연결된 프로퍼티 반환
getOwnPropertyNames()	프로퍼티 이름을 배열로 반환
keys()	열거 가능 프로퍼티 이름 반환
getOwnPropertyDescriptor()	디스크립터 속성 반환
preventExtensions()	프로퍼티 추가 금지 설정
isExtensible()	프로퍼티 추가 금지 여부 반환
seal()	프로퍼티 추가, 삭제 금지 설정
isSealed()	프로퍼티 추가, 삭제 금지 여부 반환
freeze()	프로퍼티 추가, 삭제/변경 금지 설정
isFrozen()	프로퍼티 추가, 삭제/변경 금지 여부 반환

프로퍼티 디스크립터

이름	개요
value	[[Value]], 설정할 값
writable	[[Writable]], 값 변경 가능 여부
get	[[Get]], 값 반환 프로퍼티 함수
set	[[Set]], 값 설정 프로퍼티 함수
enumerable	[[Enumerable]], 프로퍼티 열거 가능 여부
configurable	[[Configurable]], 프로퍼티 삭제 가능 여부

Object에 프로퍼티 추가

defineProperty()

구분	데이터(값)
object	Object 오브젝트
파라미터	대상 오브젝트
	프로퍼티 이름
	속성
반환	대상 오브젝트

- 대상 오브젝트에 프로퍼티 추가 또는 프로퍼티 속성 변경
- 프로퍼티마다 상태를 갖고 있음
 - 상태란? 변경/삭제/열거 가능 여부
 - 상태가 가능일 때만 처리할 수 있음
 - 프로퍼티를 추가할 때 상태 결정 [코드 1](#)

defineProperties()

구분	데이터(값)
object	Object 오브젝트
파라미터	대상 오브젝트
	프로퍼티 이름과 속성
반환	대상 오브젝트

- 다수의 프로퍼티를 추가하거나 속성 변경
 - 함수 기능은 `defineProperty()`와 같음 [코드 1](#)

프로퍼티 디스크립터

프로퍼티 디스크립터

타입	속성 이름	속성 값	디폴트 값	개요
데이터	value	JS 지원 데이터 타입	undefined	프로퍼티 값으로 사용
	writable	true, false	false	false: value 값 변경 불가
액세스	get	Function Object, undefined	undefined	프로퍼티 함수
	set	Function Object, undefined	undefined	프로퍼티 함수
공용	enumerable	true, false	false	false: for-in으로 열거 불가
	configurable	true, false	false	false: 프로퍼티 삭제 불가

- 프로퍼티 디스크립터 Descriptor
 - 프로퍼티의 속성 이름과 속성 값을 정의
- 디스크립터 타입 분류
 - 데이터 프로퍼티 디스크립터
 - 액세스 Access 프로퍼티 디스크립터
 - 공용 프로퍼티 디스크립터
 - 디스크립터 타입에 속한 속성만 같이 사용할 수 있음

디스크립터 타입 인식 기준

타입	속성 이름
데이터	value
	writable
악세스	get
	set
공용	enumerable
	configurable

- 먼저 **value** 또는 **writable** 작성 체크
- 작성되어 있으면
 - 데이터 프로퍼티 디스크립터 타입으로 인식
- 작성되어 있지 않으면
 - 악세스 프로퍼티 디스크립터 타입으로 인식
- 데이터와 악세스 프로퍼티 디스크립터를 함께 작성할 수 없으므로 구분 가능

value 속성

- 프로퍼티 값을
 - {value: "JS북"} 형태로 작성
 - for~in에서 "JS북"이 프로퍼티 값이 됨 [코드 1](#)
- value 속성을
get/set 속성과 같이 작성 불가 [코드 2](#)

writable 속성

- 프로퍼티 값 변경 가능, 불가
- **writable: true**
 - 프로퍼티 변경 가능 [코드 1](#)
- **writable: false**
 - 디폴트 값: **false**
 - 프로퍼티 변경 불가
에러가 발생하지 않지만,
값이 변경되지 않음 [코드 2](#)

enumerable 속성

- `for~in`으로 열거 가능 여부
- `enumerable: true`
 - 프로퍼티 열거 가능 [코드 1](#)
- `enumerable: false`
 - 디폴트 값: `false`
 - 프로퍼티 열거 불가 [코드 2](#)

configurable 속성

- 프로퍼티 삭제 가능, 불가
- configurable: true
 - 프로퍼티 삭제 가능
 - value 이외 속성 변경 가능 [코드 1](#)
- configurable: false
 - 디폴트 값: false
 - 프로퍼티 삭제 불가
 - value 이외 속성 변경 불가 [코드 2](#)

getter, setter

get 속성

- getter
 - OOP 용어
- `var result = obj.book;` 코드를 만나면 [코드 1](#)
 - `obj.book`의 `get` 함수가 호출되며
 - `get` 함수에서 "JS책"을 반환
 - 반환된 "JS책"을 `result` 변수에 할당
- `obj.book.get()`처럼 함수로 호출하면 에러 발생

set 속성

- setter
 - OOP 용어
- `obj.book = "JS책";` 코드를 만나면 [코드 1](#)
 - `obj.book`의 `set` 함수를 호출하면서
 - "JS책"을 파라미터 값으로 넘겨 줌
 - `data`의 `title` 프로퍼티에 "JS책"을 설정
- `obj.book;` 코드를 만나면
 - `obj.book`의 `get` 함수가 호출되며
 - `get` 함수에서 `data.title` 값을 반환
 - `setter`에서 설정한 "JS책"이 반환됨

프로퍼티 추출

getPrototypeOf()

구분	데이터(값)
object	Object 오브젝트
파라미터	대상 인스턴스
반환	프로퍼티

- 파라미터의 `prototype`에
연결된 프로퍼티 반환 [코드 1](#)
- 참고
 - `setPrototypeOf()`가
ES5 스펙에 없고 ES6에 있음

getOwnPropertyNames()

구분	데이터(값)
object	Object 오브젝트
파라미터	대상 오브젝트
반환	[프로퍼티 이름]

- 오브젝트의 프로퍼티 이름을 배열로 반환
- 열거 가능 여부를 체크하지 않음
- 자신이 만든 프로퍼티가 대상
 - 다른 오브젝트에서 받은 프로퍼티는 제외 [코드 1](#)

keys()

구분	데이터(값)
object	Object 오브젝트
파라미터	대상 오브젝트
반환	프로퍼티 이름

- 열거 가능 프로퍼티 이름 반환
 - (enumerable: true) [코드 1](#)

프로퍼티 디스크립터 함수

getOwnPropertyDescriptor()

구분	데이터(값)
object	Object 오브젝트
파라미터	대상 오브젝트
	프로퍼티 이름
반환	디스크립터

- 프로퍼티 디스크립터의 속성 이름, 값 반환
 - 다른 오브젝트에서 받은 프로퍼티는 제외 [코드 1](#)

preventExtensions()

구분	데이터(값)
object	Object 오브젝트
파라미터	대상 오브젝트
반환	대상 오브젝트

- 오브젝트에 프로퍼티 추가 금지 설정
- 프로퍼티 삭제, 변경은 가능
- 추가 금지를 설정한 후에는
추가 가능으로 변경 불가 [코드 1](#)

isExtensible()

구분	데이터(값)
object	Object 오브젝트
파라미터	대상 오브젝트
반환	true/false

- 오브젝트에 프로퍼티 추가 금지 여부 반환
 - **true**: 추가 가능, **false**: 추가 불가 [코드 1](#)

seal()

구분	데이터(값)
object	Object 오브젝트
파라미터	대상 오브젝트
반환	대상 오브젝트

- 오브젝트에 프로퍼티 추가, 삭제 금지 설정
- 추가 금지는 오브젝트 단위로 설정하고
삭제 금지는 프로퍼티 단위로 설정 [코드 1](#)
- 추가 금지를 하더라도 변경은 가능

isSealed()

구분	데이터(값)
object	Object 오브젝트
파라미터	대상 오브젝트
반환	true/false

- 오브젝트에 프로퍼티
추가, 삭제 금지 여부 반환
 - true: 불가, false: 가능 [코드 1](#)

freeze()

구분	데이터(값)
object	Object 오브젝트
파라미터	대상 오브젝트
반환	대상 오브젝트

- 오브젝트에 프로퍼티
추가, 삭제, 변경 금지 설정 [코드 1](#)

isFrozen()

구분	데이터(값)
object	Object 오브젝트
파라미터	대상 오브젝트
반환	true/false

- 오브젝트에 프로퍼티
추가, 삭제, 변경 금지 여부 반환
 - true: 불가, false: 가능 [코드 1](#)