

Object 오브젝트

자바스크립트의 오브젝트

자바스크립트 오브젝트

- 오브젝트 구분
 - 빌트인 오브젝트 Built-in Object
 - 네이티브 오브젝트 Native Object
 - 호스트 오브젝트 Host Object


네이티브 오브젝트

- 빌트인 오브젝트
 - 사전에 만들어 놓은 오브젝트
 - 빌트인 **Number** 오브젝트,
빌트인 **String** 오브젝트
- 네이티브 오브젝트
 - **JS** 스펙에서 정의한 오브젝트
 - 빌트인 오브젝트 포함, 여기에
 - **JS** 코드를 실행할 때 만드는 오브젝트
 - 예: **Argument** 오브젝트

호스트 오브젝트

- 빌트인, 네이티브 오브젝트를 제외한 오브젝트
 - 예: window, DOM 오브젝트 [코드 1](#)
- JS는 호스트 환경에서
 - 브라우저의 모든 요소 기술을
 - 연결하고 융합하며 이를 제어

오브젝트와 인스턴스

- 강좌에서는
 - 오브젝트, 인스턴스를 구분 

오브젝트와 인스턴스

- 강좌에서는
 - 오브젝트, 인스턴스를 구분 ++
- 강좌에서 오브젝트는
 - `new` 연산자를 사용하지 않고
 - 빌트인 오브젝트로 만든 오브젝트를 지칭
- `var abc = new Object();`
`var obj = {};`
 - `new` 연산자를 사용한 `abc`는 인스턴스
 - 사용하지 않은 `obj`는 오브젝트
- 오브젝트를 번역하면 객체이지만
뉘앙스 차이 있으며, 단계적으로 설명합니다.

빌트인 **Object** 프로퍼티(ES3)

프로퍼티 리스트(ES3)

이름	개요
new Object()	파라미터 데이터 타입의 인스턴스 생성
Object()	Object 인스턴스 생성
Object.prototype	
constructor	생성자
valueOf()	프리미티브 값 반환
hasOwnProperty()	프로퍼티 소유 여부 반환
propertyIsEnumerable()	프로퍼티 열거 여부 반환
isPrototypeOf()	prototype의 존재 여부 반환
toString()	문자열로 변환
toLocaleString()	지역화 문자열로 변환

Object 인스턴스 생성

new Object()

구분	데이터(값)
파라미터	값opt
반환	생성한 인스턴스

- 인스턴스를 생성하여 반환
- 파라미터의 데이터 타입에 따라
 - 생성할 인스턴스 결정 [코드 1](#) [코드 2](#)
- 파라미터 값이 undefined, null이면
 - 빈 Object 인스턴스 반환 [코드 3](#)

Object()

구분	데이터(값)
파라미터	값opt
반환	생성한 인스턴스

- Object 인스턴스 생성
 - 파라미터는 {name: value} 형태 [코드 1](#)

Object 생성 방법

- `var abc = {};`
 - `var abc = Object();`와 같음
 - 즉, `var abc = {}`를 실행하면
Object 인스턴스가 생성됨 [코드 1](#) [코드 2](#)
- `{}` 표기를
 - 오브젝트 리터럴 Literal이라고 부름

valueOf()

구분	데이터(값)
data	Object 인스턴스, 숫자
파라미터	사용하지 않음
반환	프리티미티브 값

- data 위치에 작성한
Object 인스턴스의 프리미티브 값 반환 [코드 1](#)

빌트인 오브젝트 구조

빌트인 오브젝트 구조

- 오브젝트 이름(Object, String, Number...)
- 오브젝트.prototype
 - 인스턴스 생성 가능 여부 기준
 - 프로퍼티를 연결하는 오브젝트
- 오브젝트.prototype.constructor
 - 오브젝트의 생성자
- 오브젝트.prototype.method
 - 메소드 이름과 함수 작성
- [오브젝트 구조](#)

함수와 메소드

함수와 메소드 연결

- 함수
 - 오브젝트에 연결
 - `Object.create()`
- 메소드
 - 오브젝트의 `prototype`에 연결
 - `Object.prototype.toString()`

15.2.3.5 Object.create (O [, Properties])

The create function creates a new object with a specified prototype.

15.2.4.2 Object.prototype.toString ()

When the toString method is called, the following steps are taken:

함수, 메소드 호출

- 함수 호출 방법
 - `Object.create();` [코드 1](#)
- 메소드 호출 방법
 - `Object.prototype.toString();`
 - 또는 인스턴스를 생성하여 호출 [코드 2](#)
- 함수와 메소드를 구분해야 하는 이유
 - JS 코드 작성 방법이 다르기 때문
 - 함수는 파라미터에 값을 작성하고
메소드는 메소드 앞에 값을 작성 [코드 3](#)

메소드와 메서드

- 메서드(method): [méθəd]
- 국립국어원 표준국어대사전
 - 메소드는 검색되고 메서드는 검색되지 않음
 - 프로그램과 관련지어 설명하고 있음
 - 강좌에서는 메소드로 표기

프로퍼티 처리 메소드

hasOwnProperty()

구분	데이터(값)
object	기준 인스턴스
파라미터	프로퍼티 이름
반환	true, false

- 인스턴스에 파라미터 이름이
 - 존재하면 **true** 반환
 - 존재하지 않으면 **false** 반환 [코드 1](#) [코드 2](#)
- 자신이 만든 것이 아니라
 - 상속받은 프로퍼티이면 **false** 반환 [코드 3](#)

propertyIsEnumerable()

구분	데이터(값)
object	인스턴스, 오브젝트
파라미터	프로퍼티 이름
반환	true, false

- 오브젝트에서 프로퍼티를
 - 열거할 수 있으면 **true** 반환 [코드 1](#)
 - 열거할 수 없으면 **false** 반환 [코드 2](#)

Object와 prototype

빌트인 **Object** 특징

- 인스턴스를 만들 수 있는
모든 빌트인 오브젝트의 `__proto__`에
`Object.prototype`의 6개 메소드가 설정됨
- 따라서 빌트인 오브젝트로 만든
인스턴스에도 설정됨
- `Object.prototype`

isPrototypeOf()

구분	데이터(값)
object	검색할 오브젝트.prototype
파라미터	검색 대상 오브젝트
반환	true, false

- 파라미터에 작성한 오브젝트에
 - **object** 위치에 작성한 **prototype**이
 - 존재하면 **true** 반환
 - 존재하지 않으면 **false** 반환 [코드 1](#)

toString()

구분	데이터(값)
object	Object 인스턴스
파라미터	사용 불가
반환	변환한 값

- 인스턴스 타입을 문자열로 표시 [코드 1](#)
- 오브젝트에 toString()이 있으면
 - toString()이 호출되고
 - 없으면 Object의 toString()이 호출됨 [코드 2](#)

toLocaleString()

구분	데이터(값)
data	변환 대상
파라미터	사용하지 않음
반환	변환한 값

- 지역화 문자 변환 메소드 대체 호출
 - Array, Number, Date 오브젝트의
 - toLocaleString() 메소드가
먼저 호출됩니다. [코드 1](#)