

JS

2장

자바스크립트 코어 객체와 배열





1

객체란?

객체 개념

3

□ 현실 세계는 객체들의 집합

- ▣ 사람, 책상, 자동차, TV 등
- ▣ 객체는 자신만의 고유한 **속성** 존재
 - 자동차: <색상:오렌지, 배기량:3000CC, 제조사:한성, 번호:서울1-1>
 - 사람: <이름:김선린, 나이:17, 성별:남, 주소:서울>
 - 은행계좌: <소유자:황기태, 계좌번호:111, 잔액:35000원>



자동차 객체(car)



은행 계좌(account)

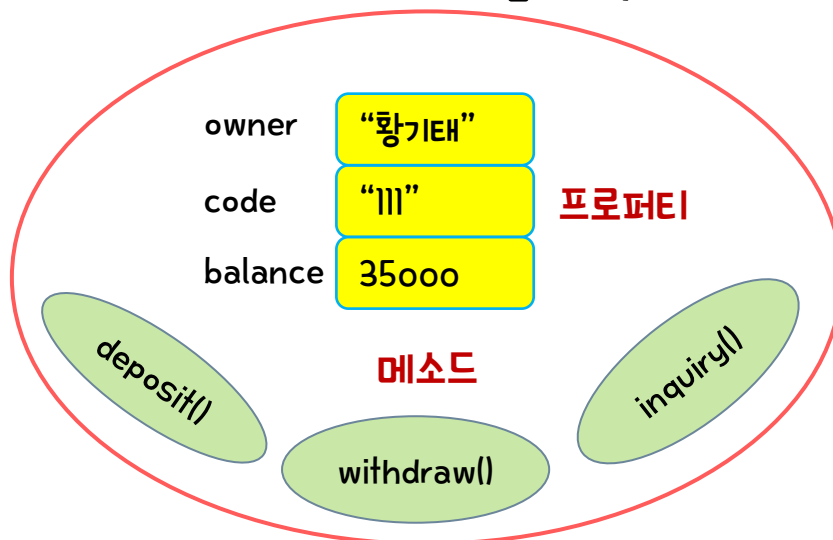
자바스크립트 객체

4

□ 자바스크립트 객체 구성

▣ 여러 개의 **프로퍼티**(property)와 **메소드**(method)로 구성

- 프로퍼티 : 객체의 고유한 속성(변수)
- 메소드(method) : 함수. 객체 내부의 프로퍼티 값을 조작하거나 연산을 수행하고 결과 리턴



자바스크립트 객체 account

```
var account = {  
  owner    : "황기태",  
  code     : "111",  
  balance  : 35000,  
  deposit  : function() { ... },  
  withdraw : function() { ... },  
  inquiry  : function() { ... }  
};
```

account 객체를 만드는 자바스크립트 코드

자바스크립트 객체 종류

5

- 자바스크립트는 객체 기반 언어
 - ▣ 자바스크립트는 객체 지향 언어 아님! (→캡슐화, 상속, 다형성 등의 특징을 완벽히 갖추지 못함)
- 브라우저가 제공하는 자바스크립트 객체

1. 코어 객체 (내장 객체)

- 자바스크립트 언어가 실행되는 어디서나 사용 가능한 기본 객체
- 자바스크립트 엔진에 내장되어 있음
- 배열(Array), 날짜(Date), 문자(String), 수학(Math) 타입 등

2. 문서 객체 (DOM(Document Object Model))

- HTML 문서에 작성된 각 HTML 태그들을 객체화한 것들
- HTML 문서의 내용과 모양을 제어하기 위한 목적

3. 브라우저 객체 (BOM(Browser Object Model))

- 자바스크립트로 브라우저를 제어하기 위해 제공되는 객체
- 브라우저에 계층 구조로 내장되어 있는 객체
- window, screen, location, history, navigator 등



2

코어 객체 (내장 객체)

코어 객체 (내장 객체)

7

□ 코어 객체 종류

- ▣ `Array, Date, String, Math` 등

□ 코어 객체 생성

- ▣ `new` 키워드 이용

```
var today = new Date();    // 시간 정보를 다루는 Date 타입의 객체 생성  
var msg = new String("Hello"); // "Hello" 문자열을 담은 String 타입의 객체 생성
```

- 이때 `today, msg` 는 생성된 객체를 가리키는 **레퍼런스 변수** (대부분 그냥 객체라고 부름)

- ▣ 객체가 생성되면 객체 내부에 프로퍼티와 메소드들 존재

□ 객체 접근

- ▣ 객체와 멤버 사이에 **점(.)** 연산자 이용

```
obj.프로퍼티 = 값;           // 객체 obj의 프로퍼티 값 변경  
변수 = obj.프로퍼티;        // 객체 obj의 프로퍼티 값 알아내기  
obj.메소드(매개변수 값들);   // 객체 obj의 메소드 호출
```

예제) 자바스크립트 객체 생성 및 활용

8

<script>

// Date 객체 생성

객체 생성

var today = new Date();

// Date 객체의 toLocaleString() 메소드 호출

document.write("현재 시간 : " + today.toLocaleString() + "
");

메소드 호출

// String 객체 생성

var myStr = new String("자바스크립트 공부하기");

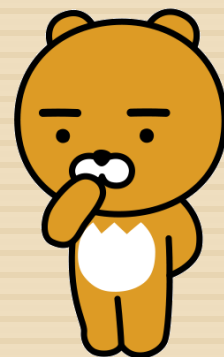
document.write("myStr의 내용 : " + myStr + "
");

document.write("myStr의 길이 : " + myStr.length + "
");

프로퍼티 읽기

</script>

현재 시간 : 2018. 9. 2. 오후 2:55:18
myStr의 내용 : 자바스크립트 공부하기
myStr의 길이 : 11



2-1

코어 객체 (내장 객체) - Array

배열 (Array)

lo

□ 배열

- 여러 개의 원소들을 연속적으로 저장
- 전체를 하나의 단위로 다루는 데이터 구조

□ 배열 생성 예시

**** 배열의 크기를 미리 지정하지 않음**

//c와 자바와의 차이점. 연속적인 공간이 아니라, 산재해 있어
마음대로 늘이거나 줄일 수 있음. 쓰기는 편하나 효율의 차이 존재

```
var cities = ["Seoul", "New York", "Paris"];
```

cities	"Seoul"	cities[0]
	"New York"	cities[1]
	"Paris"	cities[2]

```
var n = [4, 5, -2, 28, 33];
```

n	4	5	-2	28	33
	n[0]	n[1]	n[2]	n[3]	n[4]

- 0부터 시작하는 **인덱스**를 이용하여 배열의 각 원소 접근

```
var name = cities[0];  
cities[1] = "Gainesville";
```

```
// name은 "Seoul"  
// "New York" 자리에 "Gainesville" 저장
```

배열 만드는 방법-1. []로 배열 만들기

11

□ 배열 만드는 2가지 방법

- []로 배열 만들기
- **Array 객체**로 배열 만들기

□ []로 배열 만들기

- [] 안에는 원소들의 초기 값 나열

```
var week = ["월", "화", "수", "목", "금", "토", "일"];  
var plots = [-20, -5, 0, 15, 20];
```

□ 배열 크기

- 배열의 크기는 고정되지 않고, 원소 추가 시 늘어남

```
plots[6] = 33; // plots 배열에 6번째 원소 추가. 배열 크기는 6이 됨  
plots[7] = 22; // plots 배열에 7번째 원소 추가. 배열 크기는 7이 됨  
plots[10] = 11; // plots 배열에 11번째 원소 추가. 앞에 비워진 부분은 empty
```

문제) []로 배열 만들기

12

- []로 정수 5개(20, 5, 8, 15, 20)를 저장할 배열을 만들기
- 원소의 값만큼 '*'를 출력하기

[]로 배열 만들기

```
*****20
*****5
*****8
*****15
*****20
```

배열 만드는 방법-2. Array 객체로 배열 만들기

13

□ 초기 값을 가진 배열 생성 (괄호!!)

```
var week = new Array("월", "화", "수", "목", "금", "토", "일");
```

□ 초기화되지 않은 배열 생성

▣ 일정 크기의 배열 생성 후 나중에 원소 값 저장

```
var week = new Array(7);    // 7개의 원소를 가진 배열 생성  
  
week[0] = "월";  
week[1] = "화";  
  
...  
week[6] = "일";
```

□ 빈 배열 생성

▣ 원소 개수를 예상할 수 없는 경우

* 반드시 인덱스 순서대로 하나씩 추가해야 함

```
var week = new Array();    // 빈 배열 생성
```

```
week[0] = "월";           // 배열 week 크기 자동으로 1  
week[1] = "화";           // 배열 week 크기 자동으로 2
```

배열의 원소 개수, **length** 프로퍼티

14

□ 배열의 크기 : Array 객체의 length 프로퍼티

```
var plots = [-20, -5, 0, 15, 20];  
var week = new Array("월", "화", "수", "목", "금", "토", "일");  
var m = plots.length;    // m은 5  
var n = week.length;    // n은 7
```

□ length 프로퍼티는 사용자가 임의로 값 변경 가능

- length 프로퍼티는 Array 객체에 의해 자동 관리
- 사용자가 임의로 값 변경 가능
 - 배열의 크기를 줄이거나 늘일 수 있음

■ 예

```
plots.length = 10; // plots의 크기는 5에서 10으로 늘어남  
plots.length = 2; // plots의 크기는 2로 줄어 들어,  
                  처음 2개의 원소 외에는 모두 삭제 됨
```

문제) Array 객체로 배열 만들기

15

- Array 객체로 배열 생성 후 15.1, 15.4, 16.1, 17.5, 19.2, 21.4 저장
- 배열의 크기만큼 for문 돌려서 총합 구한 후 평균 구하기

Array 객체로 배열 만들기

평균 온도는 17.45

자바스크립트 배열의 특징

16

- 배열은 Array 객체
 - ▣ []로 생성해도 Array 객체로 다루어짐 → Array 객체 메소드 활용 가능
- 배열에 여러 타입의 데이터(정수, 실수, 문자열, 논리값, 객체, 등) 섞어 저장 가능

```
var any = new Array(5);    // 5개의 원소를 가진 배열 생성
any[0] = 0;                // 정수 저장
any[1] = 5.5;              // 실수 저장
any[2] = "문자열";         // 문자열 저장
any[3] = new Date();       // Date 객체 저장
```


Array 객체의 메소드

17

종류	설명
join(연결 문자)	배열 객체의 데이터를 연결 문자 기준으로 1개의 문자형 데이터로 리턴. 배열을 서버로 보낼 때 유용. 현재 배열은 변하지 않음.
reverse()	배열 객체의 데이터 순서를 거꾸로 바꾼 후 반환..
sort()	배열 객체의 데이터를 오름차순으로 정렬.
slice ([index] [,index2])	배열 객체의 데이터 중 원하는 인덱스 구간만큼 잘라서 배열 객체로 반환. index1 생략시 0부터, index2생략시 끝원소까지 포함. 현재 배열은 변하지 않음.
splice(시작인덱스, 삭제할 데이터개수[, 추가할데이터])	배열 객체의 지정 데이터를 삭제하고 그 구간에 새 데이터를 삽입
concat(arr)	현재 배열에 배열 arr의 원소들을 덧붙여 만든 새 배열 리턴. 현재 배열을 변하지 않음

Array 객체의 메소드

18

종류	설명
pop()	배열에 저장된 데이터 중 마지막 인덱스에 저장된 데이터를 삭제 후 반환
push(new data)	배열 객체의 마지막 인덱스에 새 데이터를 삽입
shift()	배열 객체에 저장된 데이터 중 첫번째 인덱스에 저장된 데이터를 삭제 후 반환
unShift(new data)	배열 객체의 첫번째 인덱스에 새 데이터를 삽입

문제) Array 객체의 메소드 활용

19

- 배열 arr_1의 원소: “사당”, “잠실”, “강남”, “방배”
- 배열 arr_2의 원소 : “숙대입구”, “동역공”, “혜화”
- join()
- concat()
- slice()
- sort()
- reverse()
- splice()
- pop()
- shift()
- push()
- unshift()